

Cloud Container Engine

API Reference

Issue	01
Date	2025-08-26



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2025. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 Before You Start..... 1

2 API Overview..... 5

3 Calling APIs..... 24

3.1 Making an API Request..... 24

3.2 Authentication..... 28

3.3 Response..... 30

4 APIs..... 32

4.1 API URL..... 32

4.2 Cluster Management..... 33

4.2.1 Creating a Cluster..... 33

4.2.2 Reading a Specified Cluster..... 150

4.2.3 Listing Clusters in a Specified Project..... 202

4.2.4 Updating a Specified Cluster..... 257

4.2.5 Deleting a Cluster..... 325

4.2.6 Hibernating a Cluster..... 384

4.2.7 Waking Up a Cluster..... 389

4.2.8 Obtaining a Cluster Certificate..... 394

4.2.9 Revoking a Cluster Certificate of a User..... 407

4.2.10 Modifying Cluster Specifications..... 413

4.2.11 Querying a Job..... 426

4.2.12 Binding/Unbinding Public API Server Address..... 437

4.2.13 Obtaining Cluster Access Address..... 449

4.2.14 Obtaining a Cluster's Logging Configurations..... 459

4.2.15 Configuring Cluster Logs..... 466

4.2.16 Obtaining the Partition List..... 477

4.2.17 Creating a Partition..... 485

4.2.18 Obtaining Partition Details..... 499

4.2.19 Updating a Partition..... 506

4.2.20 Obtaining the Global OBS Access Secret of a Cluster..... 520

4.2.21 Updating the Global OBS Access Secret of a Cluster..... 524

4.2.22 Obtaining the Global OBS Access Secret of a Project..... 527

4.2.23 Updating the Global OBS Access Secret of a Project..... 530

4.2.24 Obtaining Available Cluster Flavors.....	533
4.2.25 Obtaining the Feature Gate Status.....	539
4.2.26 Obtaining the AZ List.....	541
4.2.27 Obtaining Cluster Quotas.....	545
4.3 Node Management.....	549
4.3.1 Creating a Node.....	549
4.3.2 Reading a Specified Node.....	687
4.3.3 Listing All Nodes in a Cluster.....	748
4.3.4 Updating a Specified Node.....	809
4.3.5 Deleting a Node.....	871
4.3.6 Enabling Scale-In Protection for a Node.....	933
4.3.7 Disabling Scale-In Protection for a Node.....	936
4.3.8 Synchronizing Nodes.....	940
4.3.9 Synchronizing Nodes in Batches.....	946
4.3.10 Accepting a Node.....	951
4.3.11 Managing a Node in a Customized Node Pool.....	993
4.3.12 Resetting a Node.....	1001
4.3.13 Removing a Node.....	1046
4.3.14 Migrating a Node.....	1060
4.3.15 Migrating a Node to a Custom Node Pool.....	1090
4.4 Node Pool Management.....	1095
4.4.1 Creating a Node Pool.....	1095
4.4.2 Reading a Specified Node Pool.....	1247
4.4.3 Listing All Node Pools in a Specified Cluster.....	1327
4.4.4 Updating a Specified Node Pool.....	1408
4.4.5 Deleting a Node Pool.....	1552
4.4.6 Scaling a Node Pool.....	1632
4.4.7 Synchronizing Node Configurations in a Node Pool.....	1649
4.5 Storage Management.....	1666
4.5.1 Creating a PVC (to be discarded).....	1666
4.5.2 Deleting a PVC (to be discarded).....	1678
4.6 Add-on Management.....	1685
4.6.1 Installing an Add-on Instance.....	1685
4.6.2 Listing Add-on Templates.....	1700
4.6.3 Updating an Add-on Instance.....	1710
4.6.4 Rolling Back an Add-on Instance.....	1725
4.6.5 Deleting an Add-on Instance.....	1737
4.6.6 Querying an Add-on Instance.....	1749
4.6.7 Listing Add-on Instances.....	1758
4.7 Cluster Upgrade.....	1768
4.7.1 Upgrading a Cluster.....	1768
4.7.2 Obtaining Cluster Upgrade Task Details.....	1782

4.7.3 Retrying a Cluster Upgrade Task.....	1787
4.7.4 Suspending a Cluster Upgrade Task (Deprecated).....	1791
4.7.5 Continuing to Execute a Cluster Upgrade Task (Deprecated).....	1795
4.7.6 Obtaining a List of Cluster Upgrade Task Details.....	1799
4.7.7 Performing a Pre-upgrade Check.....	1805
4.7.8 Obtaining Details About a Pre-upgrade Check Task of a Cluster.....	1815
4.7.9 Obtaining a List of Pre-upgrade Check Tasks of a Cluster.....	1824
4.7.10 Performing a Post-upgrade Check.....	1834
4.7.11 Backing Up a Cluster.....	1838
4.7.12 Obtaining a List of Cluster Backup Task Details.....	1840
4.7.13 Obtaining the Cluster Upgrade Information.....	1846
4.7.14 Obtaining a Cluster Upgrade Path.....	1851
4.7.15 Obtaining the Configuration of Cluster Upgrade Feature Gates.....	1856
4.7.16 Enabling the Cluster Upgrade Process Booting Task.....	1860
4.7.17 Obtaining a List of Upgrade Workflows.....	1872
4.7.18 Obtaining Details About a Specified Cluster Upgrade Task.....	1882
4.7.19 Updating the Status of a Specified Cluster Upgrade Booting Task.....	1892
4.8 Quota Management.....	1903
4.8.1 Querying Resource Quotas.....	1903
4.9 API Versions.....	1908
4.9.1 Obtaining API Versions.....	1909
4.10 Tag Management.....	1913
4.10.1 Adding Resource Tags to a Specified Cluster in Batches.....	1913
4.10.2 Deleting Resource Tags of a Specified Cluster in Batches.....	1921
4.10.3 Obtaining Resource Tags.....	1927
4.10.4 Obtaining Custom Tags.....	1932
4.10.5 Obtaining Node Labels.....	1935
4.11 Configuration Management.....	1939
4.11.1 Obtaining the List of Parameters That Can Be Configured for a Node Pool.....	1939
4.11.2 Obtaining the List of Parameters That Can Be Configured for a Cluster.....	1946
4.11.3 Obtaining the List of Configurable Parameters Supported by a Cluster.....	1952
4.11.4 Obtaining the Parameters That Can Be Configured for a Node Pool.....	1957
4.11.5 Changing the Values of Configuration Parameters of a Node Pool.....	1965
4.12 Chart Management.....	1977
4.12.1 Uploading a Chart.....	1977
4.12.2 Obtaining a Chart List.....	1982
4.12.3 Obtaining a Release List.....	1987
4.12.4 Updating a Chart.....	1994
4.12.5 Creating a Release.....	1999
4.12.6 Deleting a Chart.....	2007
4.12.7 Updating a Release.....	2012
4.12.8 Obtaining a Chart.....	2021

4.12.9 Deleting a Release.....	2026
4.12.10 Downloading a Chart.....	2032
4.12.11 Obtaining a Release.....	2037
4.12.12 Obtaining Chart Values.....	2043
4.12.13 Obtaining Historical Records of a Release.....	2048
4.12.14 Obtaining the Quota of a User Chart.....	2054
4.13 Permissions Management.....	2059
4.13.1 Creating an Access Policy.....	2059
4.13.2 Obtaining the Access Policy List.....	2068
4.13.3 Obtaining Access Policy Details.....	2074
4.13.4 Updating an Access Policy.....	2080
4.13.5 Deleting an Access Policy.....	2089
4.14 Add-on Instance Parameters.....	2090
4.14.1 CoreDNS.....	2090
4.14.2 CCE Container Storage (Everest).....	2095
4.14.3 CCE Node Problem Detector.....	2099
4.14.4 Kubernetes Dashboard.....	2103
4.14.5 CCE Cluster Autoscaler.....	2105
4.14.6 NGINX Ingress Controller.....	2110
4.14.7 Kubernetes Metrics Server.....	2115
4.14.8 CCE Advanced HPA.....	2118
4.14.9 CCE Cloud Bursting Engine for CCI.....	2122
4.14.10 CCE AI Suite (NVIDIA GPU).....	2125
4.14.11 CCE AI Suite (Ascend NPU).....	2129
4.14.12 Volcano Scheduler.....	2132
4.14.13 CCE Secrets Manager for DEW.....	2140
4.14.14 CCE Network Metrics Exporter.....	2142
4.14.15 NodeLocal DNSCache.....	2144
4.14.16 Cloud Native Cluster Monitoring.....	2148
4.14.17 Cloud Native Log Collection.....	2151
5 Kubernetes APIs.....	2157
6 Out-of-Date APIs.....	2162
6.1 Obtaining Cluster Certificates (Deprecated).....	2162
6.2 Creating a PV (Deprecated).....	2166
6.3 Deleting a PV (Deprecated).....	2178
7 Permissions and Supported Actions.....	2186
8 Appendix.....	2193
8.1 Status Code.....	2193
8.2 Error Codes.....	2197
8.3 Obtaining a Project ID.....	2204
8.4 Obtaining an Account ID.....	2205

8.5 Specifying Add-ons to Be Installed During Cluster Creation..... 2206

8.6 How to Obtain Parameters in the API URI.....2208

8.7 Creating a VPC and Subnet..... 2211

8.8 Creating a Key Pair..... 2212

8.9 Node Flavor Description..... 2212

8.10 Adding a Salt in the password Field When Creating a Node.....2215

8.11 Maximum Number of Pods That Can Be Created on a Node.....2217

8.12 Node OS.....2221

8.13 Space Allocation of a Data Disk.....2237

8.14 Attaching Disks to a Node..... 2245

8.15 Generating API Parameters on the Console..... 2255

1 Before You Start

Cloud Container Engine (CCE) is a container service that allows you to run containers efficiently in the cloud. CCE provides highly scalable, high-performance, enterprise-class Kubernetes clusters and supports Docker containers. With CCE, you can easily deploy, manage, and scale containerized applications in the cloud.

This document describes how to use APIs for performing operations on CCE, such as creating or deleting CCE resources, modifying resource specifications, or adding NICs. For details about all supported operations, see [API Overview](#).

If you plan to access CCE resources through an API, ensure that you are familiar with CCE concepts. For details, see [Service Overview](#).

CCE supports both Kubernetes-native APIs and proprietary APIs. With these APIs, you can use all functions of CCE.

- CCE has opened APIs through API gateways to support operations on cloud service infrastructures (for example, creating a node). Operations on cluster resources (such as [creating a workload](#)) are also supported.
- Kubernetes-native APIs: You can perform operations on cluster resources (such as [creating a workload](#)) using the Kubernetes-native API server. However, operations on cloud service infrastructures (such as creating a node) are not supported.

For details about Kubernetes-native API versions, see <https://kubernetes.io/docs/concepts/overview/kubernetes-api/>.

NOTE

- The Kubernetes-native APIs called in the current version do not support HTTP persistent connections.
- The Kubernetes-native APIs in the current version include Beta APIs, whose version names include **beta**, for example, **v1beta1**. This type of APIs varies according to Kubernetes-native APIs. Therefore, you are advised to use this type of APIs in unimportant scenarios, for example, short-term test clusters.

CCE supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see [3 Calling APIs](#).

Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary with services and regions. An endpoint can be obtained from [Regions and Endpoints](#).

You need to select an endpoint based on your service requirements.

- The URL format for cluster, node, node pool, add-on, and quota management is **https://Endpoint/uri**. *uri* indicates the resource path, that is, the API access path.
- The URL format for Kubernetes APIs, storage management, and add-on management is **https://{clusterid}.Endpoint/uri**. In the URL, *{clusterid}* indicates the cluster ID, and *uri* indicates the resource path, that is, the path for API access.

NOTE

- The format of the URL called by the add-on management APIs is **https://{clusterid}.Endpoint/uri**. However, *{clusterid}* is used only for the domain name and is not verified or used by the APIs. Set *{clusterid}* in the query or body. For details about *{clusterid}*, see the add-on management sections.
- *{clusterid}* is required for Kubernetes APIs and storage management, which indicates the cluster that needs to be accessed by calling the API.

Table 1-1 URL parameters

Parameter	Description
{clusterid}	Cluster ID. After a cluster is created, call the API for obtaining a cluster in a specified project to obtain the cluster ID.
Endpoint	Entry (URL) for a web service. Endpoints vary depending on services and regions.
uri	Access path of an API for performing an operation. Obtain the path from the URI of an API. For example, the resource-path of the API used to obtain a user token is v3/auth/tokens .

Notes and Constraints

- CCE imposes a quota on the number and capacity of resources that a user can access. By default, you can create a maximum of five clusters in each region and a cluster can have a maximum of 50 nodes. To create more clusters or add more nodes, [submit a service ticket](#) to increase the quota. For more details about quotas, see [Quotas](#).
- For more constraints, see API description.

Concepts

- Account

An account is created upon successful registration. The account has full access permissions for all of its cloud services and resources. It can be used to reset

user passwords and grant user permissions. The account is a payment entity, which should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.

- User

An IAM user is created using an account to use cloud services. Each IAM user has their own identity credentials (password and access keys).

The account name, username, and password will be required for API authentication.

- Region

Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified into universal regions and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides specific services for specific tenants.

For details, see [Region and AZ](#).

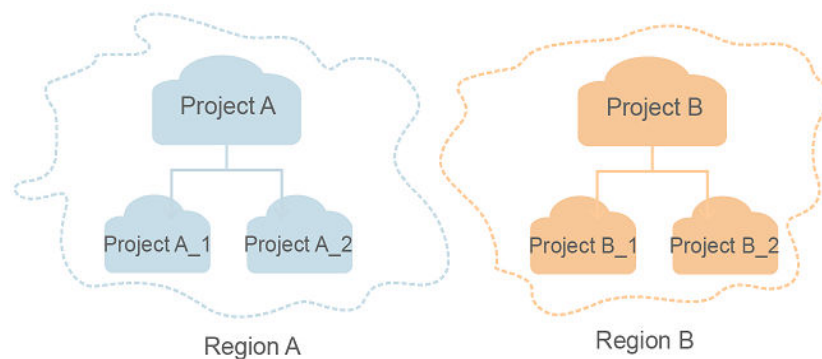
- AZ

An AZ comprises of one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

- Project

A project corresponds to a region. Default projects are defined to group and physically isolate resources (including compute, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources under their accounts in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolation model



- Enterprise project

Enterprise projects group and manage resources across regions. Resources in different enterprise projects are logically isolated. An enterprise project can

contain resources of multiple regions, and resources can be added to or removed from enterprise projects.

For details about enterprise projects and about how to obtain enterprise project IDs, see [Enterprise Management User Guide](#).

2 API Overview

APIs provided by CCE are classified into two types: proprietary APIs and Kubernetes-native APIs. By using these two types of APIs, you can use all functions provided by CCE, including creating clusters and nodes, using the Kubernetes-native APIs to create workloads, and using the proprietary CCE APIs to monitor application data.

Type	Subtype	Description
Proprietary CCE APIs	Cluster-related APIs	Manage clusters, including creating and deleting clusters. You can use APIs in this category to create clusters and obtain information about created clusters.
	Node-related APIs	Manage nodes, including creating and deleting nodes. You can use these APIs in this category to add nodes to clusters and obtain information about created nodes.
	Node Pool-related APIs	Manage node pools, including creating and deleting node pools. You can use APIs in this category to create node pools and obtain information about created node pools.
	Add-on-related APIs	Manage add-ons, including obtaining add-on templates and creating, updating, deleting, and getting add-on instances.
	Quota-related API	Obtain quotas of CCE resources.
Kubernetes-native APIs	N/A	For details about how to call Kubernetes-native APIs, see Kubernetes APIs .

Cluster-related APIs

Table 2-1 Cluster-related APIs

API	Description
Creating a Cluster	Create an empty cluster, which has only master nodes but do not have worker nodes.
Reading a Specified Cluster	Obtain details about a specified cluster.
Listing Clusters in a Specified Project	Obtain details about all clusters in a specified project.
Updating a Specified Cluster	Update information about a specified cluster.
Deleting a Cluster	Delete a specified cluster.
Hibernating a Cluster	Hibernate a specified cluster.
Waking Up a Cluster	Wake up a hibernated cluster.
Obtaining Cluster Certificates	Obtain certificates of a specified cluster.
Obtaining Job Information	Obtain the progress of a job with a specified job ID returned after a job request is issued.

Node-related APIs

Table 2-2 Node-related APIs

API	Description
Creating a Node	Create a node in a specified cluster.
Obtaining a Specified Node	Obtain details about a node with a specified node ID.
Listing All Nodes in a Cluster	Obtain details about all nodes in a cluster with a specified cluster ID.
Updating a Specified Node	Update information about a specified node.
Deleting a Node	Delete a specified node
Accepting a Node	Accept a node into a specified cluster.
Resetting a Node	Reset a node in a specified cluster.

API	Description
Removing a Node	Remove a node from a specified cluster.
Migrating a Node	Migrate a node from a specified cluster to another cluster.

Node Pool-related APIs

Table 2-3 Node pool-related APIs

API	Description
Creating a Node Pool	Create a node pool in a specified cluster.
Reading a Specified Node Pool	Obtain details about a node with a specified node ID.
Listing All Node Pools in a Specified Cluster	Obtain details about all node pools in a cluster with a specified cluster ID.
Updating a Specified Node Pool	Update information about a specified node pool.
Deleting a Node Pool	Delete a specified node pool.

Add-on-related APIs

Table 2-4 Add-on-related APIs

API	Description
Installing an Add-on Instance	Install an add-on by using the add-on template.
Listing Add-on Templates	Query add-on information.
Updating an Add-on Instance	Update an add-on instance.
Deleting an Add-on Instance	Delete an add-on instance.
Reading an Add-on Instance	Obtain details about an add-on instance.
Listing Add-on Instances	List all add-on instances in the cluster.

Quota-related API

Table 2-5 Quota-related API

API	Description
Querying Resource Quotas	Query resource quotas.

Kubernetes APIs

API	Function	URI
Node	Reading a specified node	GET /api/v1/nodes/{name}
	Listing all nodes	GET /api/v1/nodes
	Updating a specified node	PATCH /api/v1/nodes/{name}
Namespace	Creating a namespace	POST /api/v1/namespaces
	Deleting a namespace	DELETE /api/v1/namespaces/{name}
	Querying a specified namespace	GET /api/v1/namespaces/{name}
	Replacing a specified namespace	PUT /api/v1/namespaces/{name}
	Replacing the status of a specified namespace	PUT /api/v1/namespaces/{name}/status
	Replacing the finalize values of a specified namespace	PUT /api/v1/namespaces/{name}/finalize
	Listing namespaces	GET /api/v1/namespaces
	Updating a specified namespace	PATCH /api/v1/namespaces/{name}
Resource quotas	Querying resource quotas	GET /api/v1/resourcequotas
	Creating a resource quota	POST /api/v1/namespaces/{namespace}/resourcequotas
	Updating a resource quota	PUT /api/v1/namespaces/{namespace}/resourcequotas/{name}
	Deleting a resource quota	DELETE /api/v1/namespaces/{namespace}/resourcequotas/{name}

API	Function	URI
Pod	Creating a pod	POST /api/v1/namespaces/{namespace}/pods
	Deleting a pod	DELETE /api/v1/namespaces/{namespace}/pods/{name}
	Deleting all pods	DELETE /api/v1/namespaces/{namespace}/pods
	Reading a specified pod	GET /api/v1/namespaces/{namespace}/pods/{name}
	Replacing a specified pod	PUT /api/v1/namespaces/{namespace}/pods/{name}
	Replacing the status of a specified pod	PUT /api/v1/namespaces/{namespace}/pods/{name}/status
	Listing all pods in a specified namespace	GET /api/v1/namespaces/{namespace}/pods
	Listing pods	GET /api/v1/pods
	Updating a specified pod	PATCH /api/v1/namespaces/{namespace}/pods/{name}
Deploy ment	Creating a Deployment	POST /apis/apps/v1/namespaces/{namespace}/deployments
	Rolling back a Deployment	PATCH /apis/apps/v1/namespaces/{namespace}/deployments/{name} (for clusters of v1.17 or later) POST /apis/apps/v1beta1/namespaces/{namespace}/deployments/{name}/rollback (for clusters of v1.15 and earlier) POST /apis/extensions/v1beta1/namespaces/{namespace}/deployments/{name}/rollback (for clusters of v1.15 and earlier)
	Deleting a Deployment	DELETE /apis/apps/v1/namespaces/{namespace}/deployments/{name}
	Deleting all Deployments	DELETE /apis/apps/v1/namespaces/{namespace}/deployments
	Reading a specified Deployment	GET /apis/apps/v1/namespaces/{namespace}/deployments/{name}
	Reading the status of a specified Deployment	GET /apis/apps/v1/namespaces/{namespace}/deployments/{name}/status

API	Function	URI
	Reading the scaling operation of a specified Deployment	GET /apis/apps/v1/namespaces/{namespace}/deployments/{name}/scale
	Replacing a specified Deployment	PUT /apis/apps/v1/namespaces/{namespace}/deployments/{name}
	Replacing the status of a specified Deployment	PUT /apis/apps/v1/namespaces/{namespace}/deployments/{name}/status
	Replacing the scaling operation of a specified Deployment	PUT /apis/apps/v1/namespaces/{namespace}/deployments/{name}/scale
	Listing Deployments in a specified namespace	GET /apis/apps/v1/namespaces/{namespace}/deployments
	Listing all Deployments	GET /apis/apps/v1/deployments
	Updating a specified Deployment	PATCH /apis/apps/v1/namespaces/{namespace}/deployments/{name}
	Updating the status of a specified Deployment	PATCH /apis/apps/v1/namespaces/{namespace}/deployments/{name}/status
	Updating the scaling operation of a specified Deployment	PATCH /apis/apps/v1/namespaces/{namespace}/deployments/{name}/scale
StatefulSet	Creating a StatefulSet	POST /apis/apps/v1/namespaces/{namespace}/statefulsets
	Deleting a specified StatefulSet	DELETE /apis/apps/v1/namespaces/{namespace}/statefulsets/{name}
	Deleting all StatefulSets	DELETE /apis/apps/v1/namespaces/{namespace}/statefulsets
	Reading a specified StatefulSet	GET /apis/apps/v1/namespaces/{namespace}/statefulsets/{name}
	Reading the status of a specified StatefulSet	GET /apis/apps/v1/namespaces/{namespace}/statefulsets/{name}/status
	Replacing a specified StatefulSet	PUT /apis/apps/v1/namespaces/{namespace}/statefulsets/{name}
	Replacing the status of a specified StatefulSet	PUT /apis/apps/v1/namespaces/{namespace}/statefulsets/{name}/status
	Listing StatefulSets in a specified namespace	GET /apis/apps/v1/namespaces/{namespace}/statefulsets
	Listing all StatefulSets	GET /apis/apps/v1/statefulsets

API	Function	URI
	Updating a specified StatefulSet	PATCH /apis/apps/v1/namespaces/{namespace}/statefulsets/{name}
	Updating the status of a specified StatefulSet	PATCH /apis/apps/v1/namespaces/{namespace}/statefulsets/{name}/status
Daemon Set	Creating a DaemonSet	POST /apis/apps/v1/namespaces/{namespace}/daemonsets
	Deleting a specified DaemonSet	DELETE /apis/apps/v1/namespaces/{namespace}/daemonsets/{name}
	Deleting all DaemonSets	DELETE /apis/apps/v1/namespaces/{namespace}/daemonsets
	Reading a specified DaemonSet	GET /apis/apps/v1/namespaces/{namespace}/daemonsets/{name}
	Reading the status of a specified DaemonSet	GET /apis/apps/v1/namespaces/{namespace}/daemonsets/{name}/status
	Updating a specified DaemonSet	PATCH /apis/apps/v1/namespaces/{namespace}/daemonsets/{name}
	Updating the status of a specified DaemonSet	PATCH /apis/apps/v1/namespaces/{namespace}/daemonsets/{name}/status
	Listing all DaemonSets	GET /apis/apps/v1/daemonsets
	Listing DaemonSets in a specified namespace	GET /apis/apps/v1/namespaces/{namespace}/daemonsets
	Replacing a specified DaemonSet	PUT /apis/apps/v1/namespaces/{namespace}/daemonsets/{name}
	Replacing the status of a specified DaemonSet	PUT /apis/apps/v1/namespaces/{namespace}/daemonsets/{name}/status
Job	Creating a job	POST /apis/batch/v1/namespaces/{namespace}/jobs
	Deleting a job	DELETE /apis/batch/v1/namespaces/{namespace}/jobs/{name}
	Deleting all jobs	DELETE /apis/batch/v1/namespaces/{namespace}/jobs
	Reading a specified job	GET /apis/batch/v1/namespaces/{namespace}/jobs/{name}
	Reading the status of a specified job	GET /apis/batch/v1/namespaces/{namespace}/jobs/{name}/status
	Replacing a specified job	PUT /apis/batch/v1/namespaces/{namespace}/jobs/{name}

API	Function	URI
	Replacing the status of a specified job	PUT /apis/batch/v1/namespaces/{namespace}/jobs/{name}/status
	Listing jobs in a specified namespace	GET /apis/batch/v1/namespaces/{namespace}/jobs
	Listing all jobs	GET /apis/batch/v1/jobs
	Updating the status of a specified job	PATCH /apis/batch/v1/namespaces/{namespace}/jobs/{name}/status
	Updating a specified job	PATCH /apis/batch/v1/namespaces/{namespace}/jobs/{name}
CronJob	Creating a CronJob	POST /apis/batch/v1/namespaces/{namespace}/cronjobs (only for clusters of v1.25 or later) POST /apis/batch/v1beta1/namespaces/{namespace}/cronjobs (only for clusters of v1.23 or earlier)
	Deleting a CronJob	DELETE /apis/batch/v1/namespaces/{namespace}/cronjobs/{name} (only to clusters of v1.25 or later) DELETE /apis/batch/v1beta1/namespaces/{namespace}/cronjobs/{name} (only for clusters of v1.23 or earlier)
	Deleting all CronJobs	DELETE /apis/batch/v1/namespaces/{namespace}/cronjobs (only for clusters of v1.25 or later) DELETE /apis/batch/v1beta1/namespaces/{namespace}/cronjobs (only for clusters of v1.23 or earlier)
	Reading a specified CronJob	GET /apis/batch/v1/namespaces/{namespace}/cronjobs/{name} (only for clusters of v1.25 or later) GET /apis/batch/v1beta1/namespaces/{namespace}/cronjobs/{name} (only for clusters of v1.23 or earlier)
	Reading the status of a specified CronJob	GET /apis/batch/v1/namespaces/{namespace}/cronjobs/{name}/status (only for clusters of v1.25 or later) GET /apis/batch/v1beta1/namespaces/{namespace}/cronjobs/{name}/status (only for clusters of v1.23 or earlier)

API	Function	URI
	Replacing a specified CronJob	PUT /apis/batch/v1/namespaces/{namespace}/cronjobs/{name} (only for clusters of v1.25 or later) PUT /apis/batch/v1beta1/namespaces/{namespace}/cronjobs/{name} (only for clusters of v1.23 or earlier)
	Replacing the status of a specified CronJob	PUT /apis/batch/v1/namespaces/{namespace}/cronjobs/{name}/status (only for clusters of v1.25 or later) PUT /apis/batch/v1beta1/namespaces/{namespace}/cronjobs/{name}/status (only for clusters of v1.23 or earlier)
	Listing CronJobs under a specified namespace	GET /apis/batch/v1/namespaces/{namespace}/cronjobs (only for clusters of v1.25 or later) GET /apis/batch/v1beta1/namespaces/{namespace}/cronjobs (only for clusters of v1.23 or earlier)
	Listing all CronJobs	GET /apis/batch/v1/cronjobs (only for clusters of v1.25 or later) GET /apis/batch/v1beta1/cronjobs (only for clusters of v1.23 or earlier)
	Updating the status of a specified CronJob	PATCH /apis/batch/v1/namespaces/{namespace}/cronjobs/{name}/status (only for clusters of v1.25 or later) PATCH /apis/batch/v1beta1/namespaces/{namespace}/cronjobs/{name}/status (only for clusters of v1.23 or earlier)
	Updating a specified CronJob	PATCH /apis/batch/v1/namespaces/{namespace}/cronjobs/{name} (only for clusters of v1.25 or later) PATCH /apis/batch/v1beta1/namespaces/{namespace}/cronjobs/{name} (only for clusters of v1.23 or earlier)
ReplicaSet	Listing ReplicaSets	GET /apis/apps/v1/namespaces/{namespace}/replicasets
	Reading a specified ReplicaSet	GET /apis/apps/v1/namespaces/{namespace}/replicasets/{name}
	Listing all ReplicaSets	GET /apis/apps/v1/replicasets
ReplicationController	Creating a ReplicationController	POST /api/v1/namespaces/{namespace}/replicationcontrollers

API	Function	URI
	Deleting a ReplicationController	DELETE /api/v1/namespaces/{namespace}/replicationcontrollers/{name}
	Deleting all ReplicationControllers	DELETE /api/v1/namespaces/{namespace}/replicationcontrollers
	Reading a ReplicationController under a specified namespace	GET /api/v1/namespaces/{namespace}/replicationcontrollers/{name}
	Replacing a ReplicationController under a specified namespace	PUT /api/v1/namespaces/{namespace}/replicationcontrollers/{name}
	Replacing the status of a ReplicationController under a specified namespace	PUT /api/v1/namespaces/{namespace}/replicationcontrollers/{name}/status
	Listing ReplicationControllers in a specified namespace	GET /api/v1/namespaces/{namespace}/replicationcontrollers
	Listing ReplicationControllers	GET /api/v1/replicationcontrollers
	Updating a specified ReplicationController	PATCH /api/v1/namespaces/{namespace}/replicationcontrollers/{name}
Endpoints	Creating an endpoint	POST /api/v1/namespaces/{namespace}/endpoints
	Deleting an endpoint	DELETE /api/v1/namespaces/{namespace}/endpoints/{name}
	Deleting all endpoints	DELETE /api/v1/namespaces/{namespace}/endpoints
	Querying a specified endpoint	GET /api/v1/namespaces/{namespace}/endpoints/{name}
	Replacing a specified endpoint	PUT /api/v1/namespaces/{namespace}/endpoints/{name}
	Listing endpoints	GET /api/v1/endpoints
	Listing endpoints in a specified namespace	GET /api/v1/namespaces/{namespace}/endpoints
	Updating a specified endpoint	PATCH /api/v1/namespaces/{namespace}/endpoints/{name}

API	Function	URI
Service	Creating a Service	POST /api/v1/namespaces/{namespace}/services
	Deleting a specified Service	DELETE /api/v1/namespaces/{namespace}/services/{name}
	Obtaining a specified Service	GET /api/v1/namespaces/{namespace}/services/{name}
	Replacing a specified Service	PUT /api/v1/namespaces/{namespace}/services/{name}
	Listing Services in a specified namespace	GET /api/v1/namespaces/{namespace}/services
	Listing Services	GET /api/v1/services
	Updating a specified Service	PATCH /api/v1/namespaces/{namespace}/services/{name}
Ingress	Creating an ingress	POST /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses (for clusters of v1.21 and later) POST /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses (for clusters from v1.15 to v1.21) POST /apis/extensions/v1beta1/namespaces/{namespace}/ingresses (for clusters earlier than v1.15)
	Updating a specified ingress	PATCH /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses/{name} (for clusters of v1.21 and later) PATCH /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters from v1.15 to v1.21) PATCH /apis/extensions/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters earlier than v1.15)
	Replacing a specified ingress	PUT /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses/{name} (for clusters of v1.21 and later) PUT /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters from v1.15 to v1.21) PUT /apis/extensions/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters earlier than v1.15)

API	Function	URI
	Deleting an ingress	DELETE /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses/{name} (for clusters of v1.21 and later) DELETE /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters from v1.15 to v1.21) DELETE /apis/extensions/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters earlier than v1.15)
	Deleting all ingresses	DELETE /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses (for clusters of v1.21 and later) DELETE /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses (for clusters from v1.15 to v1.21) DELETE /apis/extensions/v1beta1/namespaces/{namespace}/ingresses (for clusters earlier than v1.15)
	Obtaining a specified ingress	GET /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses/{name} (for clusters of v1.21 and later) GET /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters from v1.15 to v1.21) GET /apis/extensions/v1beta1/namespaces/{namespace}/ingresses/{name} (for clusters earlier than v1.15)
	Listing ingresses in a specified namespace	GET /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses (for clusters of v1.21 and later) GET /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses (for clusters from v1.15 to v1.21) GET /apis/extensions/v1beta1/namespaces/{namespace}/ingresses (for clusters earlier than v1.15)
	Listing ingresses	GET /apis/networking.k8s.io/v1/ingresses (for clusters of v1.21 and later) GET /apis/networking.k8s.io/v1beta1/ingresses (for clusters from v1.15 to v1.21) GET /apis/extensions/v1beta1/ingresses (for clusters earlier than v1.15)

API	Function	URI
	Obtaining the status of an ingress in a specified namespace	GET /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses/{name}/status (for clusters of v1.21 and later) GET /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name}/status (for clusters from v1.15 to v1.21) GET /apis/extensions/v1beta1/namespaces/{namespace}/ingresses/{name}/status (for clusters earlier than v1.15)
	Replacing the status of an ingress in a specified namespace	PUT /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses/{name}/status (for clusters of v1.21 and later) PUT /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name}/status (for clusters from v1.15 to v1.21) PUT /apis/extensions/v1beta1/namespaces/{namespace}/ingresses/{name}/status (for clusters earlier than v1.15)
	Updating the status of an ingress in a specified namespace	PATCH /apis/networking.k8s.io/v1/namespaces/{namespace}/ingresses/{name}/status (for clusters of v1.21 and later) PATCH /apis/networking.k8s.io/v1beta1/namespaces/{namespace}/ingresses/{name}/status (for clusters from v1.15 to v1.21) PATCH /apis/extensions/v1beta1/namespaces/{namespace}/ingresses/{name}/status (for clusters earlier than v1.15)
Network Policy	Creating a network policy	POST /apis/networking.k8s.io/v1/namespaces/{namespace}/networkpolicies
	Updating a specified network policy	PATCH /apis/networking.k8s.io/v1/namespaces/{namespace}/networkpolicies/{name}
	Replacing a specified network policy	PUT /apis/networking.k8s.io/v1/namespaces/{namespace}/networkpolicies/{name}

API	Function	URI
	Deleting a specified network policy	DELETE /apis/networking.k8s.io/v1/namespaces/{namespace}/networkpolicies/{name}
	Deleting network policies in batches	DELETE /apis/networking.k8s.io/v1/namespaces/{namespace}/networkpolicies
	Reading a specified network policy	GET /apis/networking.k8s.io/v1/namespaces/{namespace}/networkpolicies/{name}
	Listing network policies in a specified namespace	GET /apis/networking.k8s.io/v1/namespaces/{namespace}/networkpolicies
	Listing all network policies	GET /apis/networking.k8s.io/v1/networkpolicies
PersistentVolume	Creating a PersistentVolume	POST /api/v1/persistentvolumes
	Deleting a specified PersistentVolume	DELETE /api/v1/persistentvolumes/{name}
	Deleting all PersistentVolumes	DELETE /api/v1/persistentvolumes
	Reading a specified PersistentVolume	GET /api/v1/persistentvolumes/{name}
	Replacing a specified PersistentVolume	PUT /api/v1/persistentvolumes/{name}
	Replacing the status of a specified PersistentVolume	PUT /api/v1/persistentvolumes/{name}/status
	Listing all PersistentVolumes	GET /api/v1/persistentvolumes
	Updating a specified PersistentVolume	PATCH /api/v1/persistentvolumes/{name}
PersistentVolumeClaim	Creating a PersistentVolumeClaim	POST /api/v1/namespaces/{namespace}/persistentvolumeclaims
	Deleting a specified PersistentVolumeClaim	DELETE /api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}
	Deleting all PersistentVolumeClaims	DELETE /api/v1/namespaces/{namespace}/persistentvolumeclaims
	Reading a specified PersistentVolumeClaim	GET /api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}

API	Function	URI
	Replacing a specified PersistentVolumeClaim	PUT /api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}
	Replacing the status of a specified PersistentVolumeClaim	PUT /api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}/status
	Listing PersistentVolumeClaims in a specified namespace	GET /api/v1/namespaces/{namespace}/persistentvolumeclaims
	Listing all PersistentVolumeClaims	GET /api/v1/persistentvolumeclaims
	Updating a specified PersistentVolumeClaim	PATCH /api/v1/namespaces/{namespace}/persistentvolumeclaims/{name}
ConfigMap	Creating a ConfigMap	POST /api/v1/namespaces/{namespace}/configmaps
	Deleting a ConfigMap	DELETE /api/v1/namespaces/{namespace}/configmaps/{name}
	Deleting all ConfigMaps	DELETE /api/v1/namespaces/{namespace}/configmaps
	Reading a specified ConfigMap	GET /api/v1/namespaces/{namespace}/configmaps/{name}
	Replacing a specified ConfigMap	PUT /api/v1/namespaces/{namespace}/configmaps/{name}
	Listing ConfigMaps in a specified namespace	GET /api/v1/namespaces/{namespace}/configmaps
	Listing all ConfigMaps	GET /api/v1/configmaps
	Updating a specified ConfigMap	PATCH /api/v1/namespaces/{namespace}/configmaps/{name}
Secret	Creating a secret	POST /api/v1/namespaces/{namespace}/secrets
	Deleting a secret	DELETE /api/v1/namespaces/{namespace}/secrets/{name}
	Deleting all secrets in a specified namespace	DELETE /api/v1/namespaces/{namespace}/secrets
	Reading a specified secret	GET /api/v1/namespaces/{namespace}/secrets/{name}
	Replacing a specified secret	PUT /api/v1/namespaces/{namespace}/secrets/{name}

API	Function	URI
	Listing the secrets in a namespace	GET /api/v1/namespaces/{namespace}/secrets
	Listing secrets in a cluster	GET /api/v1/secrets
RBAC/ ClusterRole	Creating a ClusterRole	POST /apis/rbac.authorization.k8s.io/v1/clusterroles
	Updating a specified ClusterRole	PATCH /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}
	Replacing a specified ClusterRole	PUT /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}
	Deleting a specified ClusterRole	DELETE /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}
	Deleting ClusterRoles in batches	DELETE /apis/rbac.authorization.k8s.io/v1/clusterroles
	Reading a specified ClusterRole	GET /apis/rbac.authorization.k8s.io/v1/clusterroles/{name}
	Listing ClusterRoles	GET /apis/rbac.authorization.k8s.io/v1/clusterroles
RBAC/ ClusterRoleBinding	Creating a ClusterRoleBinding	POST /apis/rbac.authorization.k8s.io/v1/clusterrolebindings
	Updating a specified ClusterRoleBinding	PATCH /apis/rbac.authorization.k8s.io/v1/clusterrolebindings/{name}
	Replacing a specified ClusterRoleBinding	PUT /apis/rbac.authorization.k8s.io/v1/clusterrolebindings/{name}
	Deleting a specified ClusterRoleBinding	DELETE /apis/rbac.authorization.k8s.io/v1/clusterrolebindings/{name}
	Deleting ClusterRoleBindings in batches	DELETE /apis/rbac.authorization.k8s.io/v1/clusterrolebindings
	Reading a specified ClusterRoleBinding	GET /apis/rbac.authorization.k8s.io/v1/clusterrolebindings/{name}
	Listing ClusterRoleBindings	GET /apis/rbac.authorization.k8s.io/v1/clusterrolebindings
RBAC/ Role	Creating a Role	POST /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/roles
	Updating a specified Role	PATCH /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/roles/{name}

API	Function	URI
	Replacing a specified Role	PUT /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/roles/{name}
	Deleting a specified Role	DELETE /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/roles/{name}
	Deleting Roles in batches	DELETE /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/roles
	Reading a specified Role	GET /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/roles/{name}
	Listing Roles in a specified namespace	GET /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/roles
	Listing all Roles	GET /apis/rbac.authorization.k8s.io/v1/roles
RBAC/ RoleBin ding	Creating a RoleBinding	POST /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/rolebindings
	Updating a specified RoleBinding	PATCH /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/rolebindings/{name}
	Replacing a specified RoleBinding	PUT /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/rolebindings/{name}
	Deleting a specified RoleBinding	DELETE /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/rolebindings/{name}
	Deleting RoleBindings in batches	DELETE /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/rolebindings
	Reading a specified RoleBinding	GET /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/rolebindings/{name}
	Listing RoleBindings in a specified namespace	GET /apis/rbac.authorization.k8s.io/v1/namespaces/{namespace}/rolebindings
	Listing all RoleBindings	GET /apis/rbac.authorization.k8s.io/v1/rolebindings
API groups	Listing APIVersions	GET /api
	Listing APIGroups	GET /apis
	Listing APIResources of GroupVersion apiregistration.k8s.io/ v1beta1	GET /apis/apiregistration.k8s.io/v1beta1

API	Function	URI
	Listing APIResources of GroupVersion extensions/v1beta1	GET /apis/extensions/v1beta1
	Listing APIResources of GroupVersion apps/v1&apps/v1beta1	GET /apis/apps/v1 (for clusters later than v1.15) GET /apis/apps/v1beta1 (for clusters of v1.15 and earlier)
	Listing APIResources of GroupVersion authentication.k8s.io/v1	GET /apis/authentication.k8s.io/v1
	Listing APIResources of GroupVersion authentication.k8s.io/v1beta1	GET /apis/authentication.k8s.io/v1beta1
	Listing APIResources of GroupVersion authorization.k8s.io/v1	GET /apis/authorization.k8s.io/v1
	Listing APIResources of GroupVersion authorization.k8s.io/v1beta1	GET /apis/authorization.k8s.io/v1beta1
	Listing APIResources of GroupVersion autoscaling/v1	GET /apis/autoscaling/v1
	Listing APIResources of GroupVersion batch/v1	GET /apis/batch/v1
	Listing APIResources of GroupVersion certificates.k8s.io/v1beta1	GET /apis/certificates.k8s.io/v1beta1
	Listing APIResources of GroupVersion networking.k8s.io/v1	GET /apis/networking.k8s.io/v1
	Listing APIResources of GroupVersion policy/v1beta1	GET /apis/policy/v1beta1
	Listing APIResources of GroupVersion rbac.authorization.k8s.io/v1beta1	GET /apis/rbac.authorization.k8s.io/v1beta1

API	Function	URI
	Listing APIResources of GroupVersion storage.k8s.io/v1	GET /apis/storage.k8s.io/v1
	Listing APIResources of GroupVersion storage.k8s.io/v1beta1	GET /apis/storage.k8s.io/v1beta1
	Listing APIResources of GroupVersion apiextensions.k8s.io/v1beta1	GET /apis/apiextensions.k8s.io/v1beta1
	Listing APIResources of GroupVersion v1	GET /api/v1
Event	Reading events	GET /api/v1/events
	Listing events in a specified namespace	GET /api/v1/namespaces/{namespace}/events

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for [obtaining a user token](#) as an example to demonstrate how to call an API. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

{URI-scheme}://{Endpoint}/{resource-path}?{query-string}

Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be transmitted separately.

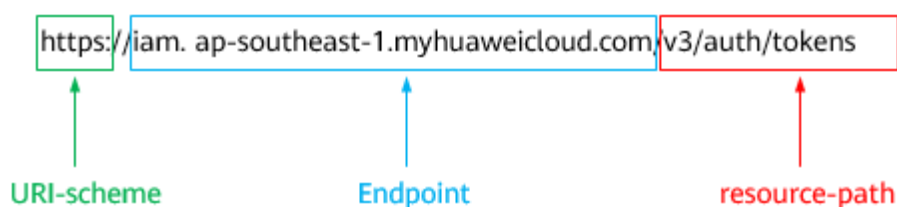
Table 3-1 URI parameters

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from Regions and Endpoints . For example, the endpoint of IAM in region CN-Hong Kong is iam.ap-southeast-1.myhuaweicloud.com .
resource-path	Access path of an API for performing an operation. Obtain the path from the URI of an API. For example, the resource-path of the API used to obtain a user token is /v3/auth/tokens .
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of <i>Parameter name=Parameter value</i> . For example, ?limit=10 indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in the **CN-Hong Kong** region, obtain the endpoint of IAM (**iam.ap-southeast-1.myhuaweicloud.com**) for this region and the **resource-path (/v3/auth/tokens)** in the URI of the API used to **obtain a user token**. Then, construct the URI as follows:

```
https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```

Figure 3-1 Example URI



NOTE

To simplify the URI display in this document, each API is provided only with a **resource-path** and a request method. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

Request Methods

The HTTP protocol defines the following request methods that can be used to send a request to the server.

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Requests the server to return the response header.
PATCH	Requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

For example, in the case of the API used to **obtain a user token through password authentication**, the request method is **POST**. The request is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```

Request Header

You can also add additional header fields to a request, such as the fields required by a specified URI or HTTP method. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

Table 3-3 lists common request header fields.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of <i>Hostname:Port number</i> . If the port number is not specified, the default port is used. The default port number for https is 443 .	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	Specifies the type (or format) of the message body. The default value application/json is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Specifies the length of the request body. The unit is byte.	No	3495

Parameter	Description	Mandatory	Example Value
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in Obtaining a Project ID .	No This field is mandatory for requests that use AK/SK authentication in the Dedicated Cloud (DeC) scenario or multi-project scenario.	e9993fc787d94b6c886cb aa340f9c0f4
X-Auth-Token	Specifies the user token. It is a response to the API for obtaining a user token through password authentication . (This is the only API that does not require authentication). After the request is processed, the value of X-Subject-Token in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvcNAQc-Co...ggg1BBIINPXsidG9rZ

 NOTE

In addition to supporting authentication using tokens, APIs support authentication using AK/SK, which uses SDKs to sign a request. During the signature, the **Authorization** (signature authentication) and **X-Sdk-Date** (time when a request is sent) headers are automatically added in the request.

For more details, see "Authentication Using AK/SK" in [Authentication](#).

The API used to [obtain a user token through password authentication](#) does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

(Optional) Request Body

This part is optional. The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers content except the request header.

The request body varies between APIs. Some APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

In the case of the API used to [obtain a user token through password authentication](#), the request parameters and description can be obtained from the API request. The following provides an example request with a body included. Replace *username*, *domainname*, ******* (login password), and *xxxxxxxxxxxxxxxxxxxx* (project name) with the actual values. You can obtain a project name from [Regions and Endpoints](#).

NOTE

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see [Obtaining a User Token Through Password Authentication](#).

POST <https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens>

Content-Type: application/json

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxxxxxxxxxxxxxx"
      }
    }
  }
}
```

If all data required for the API request is available, you can send the request to call the API through [curl](#), [Postman](#), or coding. In the response to the API used to obtain a user token, **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

3.2 Authentication

Requests for calling an API can be authenticated using either of the following methods:

- Token authentication: Requests are authenticated using tokens.

- AK/SK authentication: Requests are encrypted using an AK/SK. AK/SK-based authentication is recommended because it is more secure than token-based authentication.

Token-based Authentication

NOTE

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API used to obtain a user token.

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API.

You can obtain a token by calling the API for [obtaining a user token through password authentication](#). When you call the API, set **auth.scope** in the request body to **project**.

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****#",
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxx"
      }
    }
  }
}
```

After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when calling other APIs. For example, if the token is **ABCDEFGH....**, **X-Auth-Token: ABCDEFGH....** can be added to a request as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Auth-Token: ABCDEFGH....
```

AK/SK Authentication

NOTE

- AK/SK authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token authentication is recommended.
- The AK/SK can either be permanent or temporary. If it is temporary, the **X-Security-Token** field must be included in the request header. The value is the security token of the temporary AK/SK.

In AK/SK authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.
- SK: secret access key, which is used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

In AK/SK authentication, you can use an AK/SK to sign requests based on the signature algorithm or using the signing SDK. For details about how to sign requests and use the signing SDK, see [API Request Signing Guide](#).

 NOTE

The signing SDK is only used for signing requests and is different from the SDKs provided by services.

3.3 Response

Status Code

After sending a request, you will receive a response, including a status code, response header, and response body.

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see [Status Code](#).

For example, if status code **201** is returned for calling the API used to [obtain a user token through password authentication](#), the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-Type**.

[Figure 3-2](#) shows the response header fields for the API used to [obtain a user token through password authentication](#). The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-2 Header fields of the response to the request for obtaining a user token

```
connection → keep-alive
content-type → application/json
date → Tue, 12 Feb 2019 06:52:13 GMT
server → Web Server
strict-transport-security → max-age=31536000; includeSubdomains;
transfer-encoding → chunked
via → proxy A
x-content-type-options → nosniff
x-download-options → noopen
x-frame-options → SAMEORIGIN
x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5
x-subject-token
→ [REDACTED]
+ [REDACTED]
+ [REDACTED]
x-xss-protection → 1; mode=block;
```

Response Body

The body of a response is often returned in structured format as specified in the **Content-Type** header field. The response body transfers content except the response header.

The following is part of the response body for the API used to **obtain a user token through password authentication**.

```
{
  "token": {
    "expires_at": "2019-02-13T06:52:13.855000Z",
    "methods": [
      "password"
    ],
    "catalog": [
      {
        "endpoints": [
          {
            "region_id": "ap-southeast-1",
            .....

```

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{
  "error_msg": "The format of message is error",
  "error_code": "AS.0001"
}
```

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

4 APIs

4.1 API URL

- The URL format for cluster, node, node pool, add-on, and quota management is **https://Endpoint/uri**. *uri* indicates the resource path, that is, the API access path.
- The URL format for Kubernetes APIs, storage management, and add-on management is **https://{clusterid}.Endpoint/uri**. In the URL, *{clusterid}* indicates the cluster ID, and *uri* indicates the resource path, that is, the path for API access.

NOTE

- The format of the URL called by the add-on management APIs is **https://{clusterid}.Endpoint/uri**. However, *{clusterid}* is used only for the domain name and is not verified or used by the APIs. Set *{clusterid}* in the query or body. For details about *{clusterid}*, see the add-on management sections.
- *{clusterid}* is required for Kubernetes APIs and storage management, which indicates the cluster that needs to be accessed by calling the API.

Table 4-1 URL parameters

Parameter	Description
{clusterid}	Cluster ID. After a cluster is created, call the API for obtaining a cluster in a specified project to obtain the cluster ID.
Endpoint	URL that is the entry point for a web service. You can obtain it from Endpoints .
uri	Access path of an API for performing an operation. Obtain the path from the URI of an API. For example, the resource-path of the API used to obtain a user token is v3/auth/tokens .

4.2 Cluster Management

4.2.1 Creating a Cluster

Function

This API is used to create an empty cluster, which has only master nodes but no worker nodes. After creating a cluster by calling this API, you can add nodes by [creating nodes](#).

NOTE

- The URL for cluster management is in the format of **https://Endpoint/uri**, in which *uri* indicates the resource path, that is, the path for API access.
- By default, ICAgent is not installed when you call this API to create a cluster. If you need to install ICAgent, add "cluster.install.addons.external/install":{"addonTemplateName":"icagent"}," to **annotations** in the request body. ICAgent will be automatically installed during cluster creation. ICAgent is an O&M data collection agent used by Application Performance Management (APM). It runs on each server to collect data from probes in real time. ICAgent is the prerequisite for achieving application O&M. If ICAgent is not installed, the application O&M functions cannot be used.

Constraints

Before calling the CCE API to create a cluster, ensure that the following conditions are met:

- A VPC is available. Otherwise, the cluster cannot be created. If a VPC is available, you do not need to create a new one. A VPC provides an isolated, configurable, manageable virtual network environment for CCE clusters. If no VPC is available, you need to create one first. For details, see [Creating a VPC](#).
- Before creating a cluster, you should properly plan the container and Service CIDR blocks. After a cluster using a container tunnel network is created, you cannot modify the CIDR blocks. After a cluster using a VPC network or Cloud Native 2.0 network is created, you can only add CIDR blocks or subnet CIDR blocks to the cluster but cannot modify the existing CIDR blocks or subnet CIDR blocks. To modify the existing CIDR blocks, you need to create a new cluster. Exercise caution when performing this operation.
- An agency has been correctly created and is not deleted. If the agency verification fails, the cluster fails to be created. You can log in to the CCE console to see if there is any agency. If no agency is created, the system prompts you to create one. If an agency has been created, no message is displayed.
- By default, an account can create up to five clusters in each region. If you need to create more clusters, you can submit an application to increase the quota. For details, see [How Do I Increase My Quota?](#)

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters

Table 4-2 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Request Parameters

Table 4-3 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-4 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	Details: API type Constraints: The value cannot be changed. Options: • Cluster • cluster Default value: N/A
apiVersion	Yes	String	Details: API version Constraints: The value cannot be changed. Options: • v3 Default value: N/A

Parameter	Mandatory	Type	Description
metadata	Yes	ClusterMetadata object	Details: Basic information about a cluster. Metadata is a collection of attributes. Constraints: None
spec	Yes	CreateClusterRequestSpec object	Details: Detailed description about a cluster. CCE creates or updates objects by defining or updating spec . Constraints: None

Table 4-5 ClusterMetadata

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Cluster name Constraints: None Options: Enter 4 to 128 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A

Parameter	Mandatory	Type	Description
uid	No	String	Details: Cluster ID, which identifies a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. When you create a yearly/monthly cluster, no cluster ID will be returned in the response body. Options: N/A Default value: N/A
alias	No	String	Details: Alias of a cluster name displayed on the CCE console. The name can be changed. A cluster alias must be unique. Constraints: In the request body for creating or updating a cluster, if the cluster alias is not specified or set to null, the cluster name will be used as the cluster alias. In the response body for creating a cluster or other response bodies, if the cluster alias is not configured, no value will be returned. Options: Enter 4 to 128 characters starting with a letter and not starting or ending with a hyphen (-). Only digits, letters, and hyphens (-) are allowed. Default value: N/A

Parameter	Mandatory	Type	Description
annotations	No	Map<String,String>	Details: Cluster annotations in the format of key-value pairs <pre>"annotations": { "key1" : "value1", "key2" : "value2" }</pre> Constraints: This field is not stored in the database and is only used to specify the add-ons to be installed in the cluster. Options: N/A Default value: N/A NOTE - Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. - You can install ICAgent during cluster creation by adding the key-value pair "cluster.install.addons.external/install": "[{"addonTemplateName": "icagent"}]".
labels	No	Map<String,String>	Details: Cluster labels, in the format of key-value pairs Constraints: The value of this field is automatically generated by the system and is used by the frontend to identify the features supported by the cluster during the upgrade. Custom values are invalid. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
creationTimes tamp	No	String	Details: Time when a cluster was created Constraints: The value is automatically recorded during cluster creation and cannot be specified. Options: N/A Default value: N/A
updateTimest amp	No	String	Details: Time when a cluster was updated Constraints: The value is automatically recorded during cluster updates and cannot be specified. Options: N/A Default value: N/A
timezone	No	String	Details: Time zone of a cluster. The time zone must be included in IANA Time Zone Database , for example, Asia/Shanghai . Constraints: None Options: N/A Default value: N/A

Table 4-6 CreateClusterRequestSpec

Parameter	Mandatory	Type	Description
category	No	String	Details: Cluster type Constraints: None Options: • CCE: CCE cluster. • Turbo: CCE Turbo cluster. CCE clusters support hybrid deployment of VMs and BMSs, and heterogeneous nodes such as GPU and NPU nodes, allowing you to run your containers in a secure, stable container runtime environment based on a high-performance network model. CCE Turbo clusters run on the Cloud Native 2.0 infrastructure that features hardware and software synergy to support passthrough networking, high security and reliability, and intelligent scheduling. Default value: CCE when the container network is not set to eni Turbo when the container network is set to eni

Parameter	Mandatory	Type	Description
type	No	String	Definition Control plane node architecture Constraints N/A Range • VirtualMachine: The control plane node is an x86 server. • ARM64: The control plane node is an Arm-based Kunpeng server. Default Value VirtualMachine or ARM64 when there are not enough VMs

Parameter	Mandatory	Type	Description
flavor	Yes	String	Details: Cluster specifications. Specifications of clusters of v1.15 or later can be changed after they are created. For details, see Modifying Cluster Specifications. Configure this parameter based on actual service requirements. Constraints: None Options: • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s1.large: a large-scale CCE cluster with one control plane node and a maximum of 1,000 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes

Parameter	Mandatory	Type	Description
			Special cluster specifications in DeC are as follows: - cce.dec.s1.small: a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes) - cce.dec.s1.medium: a medium-scale DeC CCE cluster with one control plane node and a maximum of 200 worker nodes) - cce.dec.s1.large: a large-scale DeC CCE cluster with one control plane node and a maximum of 1,000 worker nodes) - cce.dec.s1.xlarge: an ultra-large-scale DeC CCE cluster with one control plane node and a maximum of 2,000 worker nodes) - cce.dec.s2.small: a small-scale DeC CCE cluster with three control plane nodes and a maximum of 50 worker nodes) - cce.dec.s2.medium: a medium-scale DeC CCE cluster with three control plane nodes and a maximum of 200 worker nodes) - cce.dec.s2.large: a large-scale DeC CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes) - cce.dec.s2.xlarge: an ultra-large-scale DeC CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes) Default value: N/A

Parameter	Mandatory	Type	Description
			NOTE The fields in the parameters are described as follows: • s1: specifies a cluster with one control plane node. If the control plane node is faulty, the cluster will become unavailable, but running workloads in the cluster are not affected. • s2: specifies an HA cluster with three control plane nodes. If one of the control plane nodes is faulty, the cluster is still available. • dec: specifies a DeC CCE cluster. For example, cce.dec.s1.small specifies a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes. • small: specifies that a cluster can manage a maximum of 50 worker nodes. • medium: specifies that a cluster can manage a maximum of 200 worker nodes. • large: specifies that a cluster can manage a maximum of 1,000 worker nodes. • xlarge: specifies that a cluster can manage a maximum of 2,000 worker nodes.

Parameter	Mandatory	Type	Description
version	No	String	Details: Cluster version, which mirrors the baseline version of the Kubernetes community. The latest commercial version is recommended. NOTE For details about the Kubernetes version policy, see Kubernetes Version Policy. • CCE Turbo clusters of v1.19 and later versions are commercially available. You can create clusters of the latest three versions on the CCE console. To learn which cluster versions are available, log in to the CCE console, create a cluster, and check the supported cluster versions. You can call APIs to create clusters of other versions. However, these cluster versions will be gradually terminated. For details about the support policy, see the CCE announcement. Constraints: The value must be in the format of vX.Y[Z[-rN]], for example, v1.30, v1.30.0, or v1.30.0-r0. Specifying either of these values will create a cluster of version 1.30. • X: major version in the Kubernetes community • Y: minor version in the Kubernetes community • Z: patch version in the Kubernetes community • N: CCE patch version. For details about CCE patch versions, see Patch Versions. Options: N/A Default value:

Parameter	Mandatory	Type	Description
			• If this parameter is left empty, a cluster of the latest version is created by default. • If a baseline cluster version is specified but the R version is not specified, a cluster of the latest R version will be created by default. It is a good practice not to specify the R version.

Parameter	Mandatory	Type	Description
platformVersion	No	String	Details: CCE cluster platform version, which is an internal version under the cluster version (version). Platform versions are used to trace iterations in a major cluster version. They are unique within a major cluster version and recounted when the major cluster version changes. Constraints: This parameter cannot be customized, and when you create a cluster, the latest corresponding platform version is selected automatically. Options: The format of platformVersion is cce.X.Y. • X: internal feature version, which indicates changes in features, patches, or OS support in the cluster version. The value starts from 1 and increases monotonically. • Y: patch version of an internal feature version. It is used only for software package update after the feature version is released. No other modification is involved. The value starts from 0 and increases monotonically. Default value: N/A

Parameter	Mandatory	Type	Description
legacyVersion	No	String	Details: (Deprecated) Outdated version of CCE clusters, which serves no practical purpose and is only used to showcase the cluster version and platform version combination. The version number is globally unique. For example, if the cluster version is va.b and the platformVersion is cce.X.Y , the value of legacyVersion is va.b.X-rY . Constraints: None Options: N/A Default value: N/A
description	No	String	Details: Cluster description, for example, which purpose the cluster is intended to serve. By default, this parameter is left unspecified. To modify cluster description after a cluster is created, call the API for updating a specified cluster or go to the cluster details page on the CCE console. Constraints: Only UTF-8 encoding is supported. The length must be less than or equal to 200 bytes. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
customSan	No	Array of strings	Details: Custom Subject Alternative Name (SAN) in the server certificate of a cluster API server, which must comply with the SSL and X509 standard formats Constraints: Duplicate names are not allowed. Options: Values in the IP address or domain name format Default value: N/A Example: SAN 1: DNS Name=example.com SAN 2: DNS Name=www.example.com SAN 3: DNS Name=example.net SAN 4: IP Address=93.184.216.34
ipv6enable	No	Boolean	Details: Whether a cluster supports IPv6 addresses. It is supported by clusters v1.15 and later. Constraints: After IPv6 is enabled, iptables is not supported. The VPC network model does not support IPv4/IPv6 dual-stack. Options: • true: IPv4/IPv6 dual-stack is enabled. • false: Only IPv4 is used. Default value: false
hostNetwork	Yes	HostNetwork object	Details: Node network parameters, including VPC and subnet ID. This field is mandatory because nodes in a cluster communicate with each other using a VPC. Constraints: None

Parameter	Mandatory	Type	Description
containerNetwork	Yes	ContainerNetwork object	Details: Container networking parameters, including the container network model and container CIDR block Constraints: None
eniNetwork	No	EniNetwork object	Details: Configuration of the Cloud Native 2.0 network model. You need to specify this field when creating a CCE Turbo cluster. Constraints: None
serviceNetwork	No	ServiceNetwork object	Details: Service CIDR block, including IPv4 CIDR block Constraints: None
publicAccess	No	PublicAccess object	Details: Cluster API access control Constraints: None
authentication	No	Authentication object	Details: Configurations of the cluster authentication mode Constraints: None
billingMode	No	Integer	Details: Cluster billing mode Constraints: None Options: • 0: pay-per-use • 1: yearly/monthly Default value: 0

Parameter	Mandatory	Type	Description
masters	No	Array of MasterSpec objects	Details: Advanced settings of the control plane nodes in a cluster. You can specify AZs for them. Constraints: If this parameter is not specified, no value is returned.
kubernetesSvcIpRange	No	String	Details: Service CIDR blocks which kubernetes clusterIP must fall within. This parameter is available only for clusters v1.11.7 and later. If this parameter is not specified during cluster creation, the default value 10.247.0.0/16 is used. This parameter is being deprecated, so you can use serviceNetwork instead. The new field contains the IPv4 CIDR blocks. Constraints: None Options: N/A Default value: N/A
clusterTags	No	Array of ResourceTag objects	Details: Cluster resource tags Constraints: None

Parameter	Mandatory	Type	Description
kubeProxyMode	No	String	Details: Service forwarding mode Constraints: None Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables is used by default.
az	No	String	Details: AZ. For details about AZs supported by CCE, see Regions and Endpoints. NOTE This field will be deprecated. You are advised to configure the AZs for the control plane nodes using the masters field. Constraints: None

Parameter	Mandatory	Type	Description
extendParam	No	ClusterExtendedParam object	Details: Extended field to decide whether a cluster will span across AZs or belong to a specified enterprise project, or whether a dedicated CCE cluster is to be created Constraints: None
enableDistMgt	No	Boolean	Details: Whether to enable support for remote clouds Constraints: Only CCE Turbo clusters support this function. Options: • true: Support for remote clouds is enabled. • false: Support for remote clouds is not enabled. Default value: false
deletionProtection	No	Boolean	Details: Cluster deletion protection, which prevents the deletion of a cluster when this function is enabled Constraints: None Options: • true: Cluster deletion protection is enabled. • false: Cluster deletion protection is disabled. Default value: false

Parameter	Mandatory	Type	Description
configurationsOverride	No	Array of PackageConfiguration objects	Details: Overrides the default component configurations in a cluster. For details about the supported components and their parameters, see Modifying Cluster Configurations. Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored.
clusterOps	No	ClusterOps object	Details: Cluster O&M settings Constraints: None
encryptionConfiguration	No	EncryptionConfiguration object	Details: Configuration of secret encryption. Only one encryption mode can be configured. By default, the CCE-managed key is used for encryption. (Users are not aware of the key being used.) Constraints: None Options: N/A Default value: N/A

Table 4-7 HostNetwork

Parameter	Mandatory	Type	Description
vpc	Yes	String	Details: ID of the VPC where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: • Method 1: Log in to the VPC console and view the VPC ID in the VPC details page. • Method 2: Use the VPC APIs. For details, see Querying VPCs .
subnet	Yes	String	Details: Network ID of the subnet where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: • Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the network ID on the displayed page. • Method 2: Use the VPC APIs. For details, see Querying Subnets .

Parameter	Mandatory	Type	Description
SecurityGroup	No	String	Details: ID of the default node security group in a cluster Constraints: None Options: N/A Default value: - If this parameter is not specified, the system automatically creates a default node security group. - If this parameter is specified, the cluster will be bound to the specified security group. NOTE To ensure proper communication, allow traffic from certain ports to the specified node security group. For details, see How Can I Configure a Security Group Rule in a Cluster?
controlPlaneSecurityGroup	No	String	Details: ID of the master node security group in a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Table 4-8 ContainerNetwork

Parameter	Mandatory	Type	Description
mode	Yes	String	Details: Container network type Constraints: Only one container network type can be selected. Options: • overlay_l2: an overlay_l2 network (container tunnel network) built for containers by using OpenVSwitch (OVS). • vpc-router: an underlay_l2 network built for containers by using IPvlan and custom VPC routes. • eni*: a Cloud Native 2.0 network. This model has integrated cloud native elastic network interfaces (ENIs), uses VPC CIDR blocks to allocate container IP addresses, and supports passthrough networking. You can use this model when creating a CCE Turbo cluster. Default value: N/A

Parameter	Mandatory	Type	Description
cidr	No	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. If the selected CIDR block conflicts with existing ones, an error will be reported. Constraints: This parameter cannot be modified after the cluster is created. (It has been deprecated. If cidrs has been configured, the specified CIDR blocks will be ignored.) Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster. Options: Values in IPv4 CIDR format Default value: If this parameter is left blank, a unique CIDR block is randomly allocated from 172.(16–31).0.0/16 and 10.(0 16 32 48 64 80 96 112).0.0/12.

Parameter	Mandatory	Type	Description
cidrs	No	Array of ContainerCIDR objects	Details: List of container CIDR blocks. In clusters v1.21 or later, the cidrs field is used. When the cluster network type is vpc-router , you can configure a maximum of 20 container CIDR blocks. In clusters earlier than v1.21, if the cidrs field is used, the first CIDR element in the array is used as the container CIDR block. Constraints: Once you have created a cluster that uses a container tunnel network, you cannot modify the CIDR blocks. Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster.

Table 4-9 ContainerCIDR

Parameter	Mandatory	Type	Description
cidr	Yes	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. Constraints: If a CIDR block conflict occurs, an error will be reported. Options: N/A Default value: N/A

Table 4-10 EniNetwork

Parameter	Mandatory	Type	Description
eniSubnetId	No	String	Details: IPv4 subnet ID of the subnet where an elastic network interface resides Constraints: IPv6 is not supported. This field will be deprecated. Use subnets instead. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets .
eniSubnetCIDR	No	String	Details: ENI subnet CIDR block Constraints: This parameter is being deprecated. The new field subnets is recommended. Options: N/A Default value: N/A
subnets	Yes	Array of NetworkSubnet objects	Details: List of IPv4 subnet IDs Constraints: None

Table 4-11 NetworkSubnet

Parameter	Mandatory	Type	Description
subnetID	Yes	String	Details: ID of the IPv4 subnet where the container network interfaces are in Constraints: IPv6 is not supported. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets .

Table 4-12 ServiceNetwork

Parameter	Mandatory	Type	Description
IPv4CIDR	No	String	Details: Value range of the Kubernetes clusterIP IPv4 CIDR block Constraints: None Options: N/A Default value: 10.247.0.0/16

Parameter	Mandatory	Type	Description
IPv6CIDR	No	String	Details: Value range of the Kubernetes clusterIP IPv6 CIDR block Constraints: IPv6 Service CIDR blocks can be configured only for CCE Turbo clusters with IPv6 dual stack enabled. Options: N/A Default value: Default value for a CCE Turbo cluster: fc00::/112 Default value for a CCE standard cluster: fd00:1234::/120

Table 4-13 PublicAccess

Parameter	Mandatory	Type	Description
cidrs	No	Array of strings	Details: Trustlist of network CIDRs that are allowed to access cluster APIs. You are advised to allow the traffic from VPC and container network CIDRs. Constraints: This parameter is valid only when a cluster is created. Options: N/A Default value: By default, no trustlist is configured, and the value is 0.0.0.0/0 .

Table 4-14 Authentication

Parameter	Mandatory	Type	Description
mode	No	String	Details: Cluster authentication mode Constraints: None Options: - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to x509. - Clusters of Kubernetes 1.13 or later support rbac and authenticating_proxy. The parameter defaults to rbac. - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to rbac. Default value: - Clusters of Kubernetes 1.11 or earlier: x509 - Clusters of Kubernetes 1.13 or later: rbac
authenticatingProxy	No	AuthenticatingProxy object	Details: Configuration related to the authenticating_proxy mode Constraints: This field is mandatory when the authentication mode is authenticating_proxy .

Table 4-15 AuthenticatingProxy

Parameter	Mandatory	Type	Description
ca	No	String	Details: X509 CA certificate (Base64-encoded) configured in authenticating_proxy mode Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A
cert	No	String	Details: Client certificate issued by the X509 CA certificate configured in authenticating_proxy mode, which is used for authentication from kube-apiserver to the extended API server. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A

Parameter	Mandatory	Type	Description
privateKey	No	String	Details: Private key of the client certificate issued by the X509 CA certificate configured in authenticating_proxy mode, which is used for authentication from kube-apiserver to the extended API server. The private key used by the Kubernetes cluster does not support password encryption. Use an unencrypted private key. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A

Table 4-16 MasterSpec

Parameter	Mandatory	Type	Description
availabilityZone	No	String	Details: AZ where a control plane node is located. The AZ name must be specified. For details about AZs supported by CCE, see Regions and Endpoints . Constraints: None Options: N/A Default value: N/A

Table 4-17 ResourceTag

Parameter	Mandatory	Type	Description
key	No	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: _.:=-@ • The value cannot start with _sys_. Default value: N/A
value	No	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: _.:/=+-@ Default value: N/A

Table 4-18 ClusterExtendParam

Parameter	Mandatory	Type	Description
clusterAZ	No	String	Details: AZs of master nodes in a cluster For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: • AZs supported by a specified region • multi_az: (Optional) multiple AZs, which can be configured only when a cluster with multiple master nodes is used • <i>AZs of the dedicated cloud computing pool</i>: Master nodes will be deployed in the DeC AZs. It is mandatory for dedicated CCE clusters. Default value: If no AZ is specified, an AZ is randomly allocated by default.

Parameter	Mandatory	Type	Description
dssMasterVolumes	No	String	Details: Whether the system and data disks of a master node use DSS storage pools. If this parameter is unspecified or left blank, EVS disks are used by default. Constraints: This parameter is mandatory for dedicated CCE clusters. It is in the following format: <code><rootVol.dssPoolID>.<rootVol.volType>;<dataVol.dssPoolID>.<dataVol.volType></code> Specifically: • rootVol specifies the system disk. dataVol specifies a data disk. • dssPoolID specifies the ID of a DSS storage pool. • volType specifies the storage volume type of the DSS storage pool, such as SAS, SSD, SATA, ESSD, GPSSD, ESSD2, and GPSSD2. Options: N/A Default value: N/A Example: c950ee97-587c-4f24-8a74-3367e3da570f.sas;6edbc2f4-1507-44f8-ac0d-eed1d2608d38.ssd NOTE This field cannot be configured for non-dedicated CCE clusters.

Parameter	Mandatory	Type	Description
enterpriseProjectId	No	String	Details: ID of the enterprise project that a cluster belongs to Constraints: An enterprise project can be configured only after the enterprise project function is enabled. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
kubeProxyMode	No	String	Details: Service forwarding mode Constraints: This parameter has been deprecated. If both this parameter and kubeProxyMode in ClusterSpec are specified, the latter is used. Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables

Parameter	Mandatory	Type	Description
clusterExternalIP	No	String	Details: EIP of a cluster control plane node. After an EIP is bound to the control plane node, you can access the cluster management plane APIs through the EIP. Constraints: None Options: N/A Default value: N/A
alpha.cce/fixPoolMask	No	String	Details: Number of mask bits of the fixed IP address pool of the container networks. This parameter determines the maximum number of IP addresses that can be assigned to containers on a node. This, along with the maxPods setting during node creation, determines the maximum number of pods that can be created on a node. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: Only the vpc-router networks support this function. Options: Integers from 24 to 28 Default value: 24

Parameter	Mandatory	Type	Description
decMasterFlavor	No	String	Details: Master node flavor in a dedicated CCE cluster Constraints: None Options: N/A Default value: N/A
dockerUmaskMode	No	String	Details: Default UmaskMode configuration of Docker in a cluster Constraints: None Options: • secure • normal Default value: normal

Parameter	Mandatory	Type	Description
kubernetes.io/cpuManagerPolicy	No	String	Details: Cluster CPU management policy Constraints: None Options: • none or null: disables pods from exclusively occupying CPUs. Select this option if you want a large pool of shareable CPU cores. • static: enables pods to exclusively occupy CPUs. Select this option if your workload is sensitive to CPU cache and scheduling latency. In a CCE Turbo cluster, this setting is valid only for nodes where common containers, not secure containers, run. Default value: none
orderId	No	String	Details: Order ID Constraints: This parameter is returned in the response when the cluster is billed on a yearly/monthly basis with auto payment enabled (only in creation scenarios). Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
periodType	No	String	Details: Subscription term unit Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1, which is, the yearly/monthly billing mode is configured. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • month • year Default value: N/A
periodNum	No	Integer	Details: Number of subscription periods Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value ranges from 1 to 3. Default value: N/A

Parameter	Mandatory	Type	Description
isAutoRenew	No	String	Details: Whether to enable auto renewal Constraints: This field is valid only when billingMode is set to 1 . Options: • true: Auto renewal is enabled. • false: Auto renewal is not enabled. Default value: false
isAutoPay	No	String	Details: Whether to enable auto payment Constraints: This field is valid only when billingMode is set to 1 . Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: false
upgradefrom	No	String	Details: Records of how a cluster is upgraded to its current version Constraints: This field is only returned by a query API. Options: N/A Default value: N/A

Table 4-19 PackageConfiguration

Parameter	Mandatory	Type	Description
name	No	String	Details: Component name Constraints: None Options: N/A Default value: N/A
configurations	No	Array of ConfigurationItem objects	Details: Component configuration items Constraints: None

Table 4-20 ConfigurationItem

Parameter	Mandatory	Type	Description
name	No	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
value	No	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Table 4-21 ClusterOps

Parameter	Mandatory	Type	Description
alarm	Yes	AlarmInfo object	Details: Alarm assistant settings. CCE is integrated with AOM, which enables alarm functionality, allowing for easy search and configuration of alarms. The Cloud Native Cluster Monitoring add-on sends metric rule data in CCE alarm center to AOM instances. Constraints: None

Table 4-22 AlarmInfo

Parameter	Mandatory	Type	Description
topics	Yes	Array of strings	Details: Contact group list. You can enter SMN topic names and configure contact groups to manage endpoints that have subscribed to alarm messages. Constraints: None
alarmRuleTemplateId	No	String	Details: Alarm rule template ID transferred when the alarm assistant is enabled. By default, the alarm rule template in the container scenario is used. Constraints: None Options: N/A Default value: N/A
promInstanceId	No	String	Details: ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
promEnterpriseProjectID	No	String	Details: Enterprise project ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A

Table 4-23 EncryptionConfig

Parameter	Mandatory	Type	Description
mode	No	String	Details: Encryption mode. Either encryption using the CCE-managed key or KMS key can be configured. Constraints: None Options: • Default: Encryption is performed using the CCE-managed key. • KMS: KMS encryption is used. Default value: Default

Parameter	Mandatory	Type	Description
kmsKeyID	No	String	Details: KMS key ID • In the API for creating a cluster, if the mode field is set to Default, there is no need to specify this field. If the mode field is set to KMS, you can specify a value for this field. If this parameter is left blank, the default KMS key is used by default. If the default key does not exist, DEW will automatically create a default key (cce/default) for the user. To ensure proper cluster functionality, it is important to use a valid KMS key. Refrain from deleting or disabling the key before the cluster lifecycle is complete, as this may cause exceptions. Once a key has been configured for a cluster, it cannot be modified. • In the cluster query API, if the mode field is set to Default, the returned value of this field is empty. If the mode field is set to KMS, the specific key ID will be returned. Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 201

Table 4-24 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: • Cluster • cluster Default value: N/A
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: • v3 Default value: N/A
metadata	ClusterMetadata object	Details: Basic information about a cluster. Metadata is a collection of attributes. Constraints: None
spec	ClusterSpec object	Details: Detailed description about a cluster. CCE creates or updates objects by defining or updating spec . Constraints: None
status	ClusterStatus object	Details: Cluster status and ID of a cluster creation job Constraints: None

Table 4-25 ClusterMetadata

Parameter	Type	Description
name	String	Details: Cluster name Constraints: None Options: Enter 4 to 128 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: Cluster ID, which identifies a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. When you create a yearly/monthly cluster, no cluster ID will be returned in the response body. Options: N/A Default value: N/A

Parameter	Type	Description
alias	String	Details: Alias of a cluster name displayed on the CCE console. The name can be changed. A cluster alias must be unique. Constraints: In the request body for creating or updating a cluster, if the cluster alias is not specified or set to null, the cluster name will be used as the cluster alias. In the response body for creating a cluster or other response bodies, if the cluster alias is not configured, no value will be returned. Options: Enter 4 to 128 characters starting with a letter and not starting or ending with a hyphen (-). Only digits, letters, and hyphens (-) are allowed. Default value: N/A

Parameter	Type	Description
annotations	Map<String,String>	Details: Cluster annotations in the format of key-value pairs <pre>"annotations": { "key1" : "value1", "key2" : "value2" }</pre> Constraints: This field is not stored in the database and is only used to specify the add-ons to be installed in the cluster. Options: N/A Default value: N/A NOTE - Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. - You can install ICAgent during cluster creation by adding the key-value pair "cluster.install.addons.external/install": "[{\addonTemplateName \":\"icagent\"}]".
labels	Map<String,String>	Details: Cluster labels, in the format of key-value pairs Constraints: The value of this field is automatically generated by the system and is used by the frontend to identify the features supported by the cluster during the upgrade. Custom values are invalid. Options: N/A Default value: N/A

Parameter	Type	Description
creationTimestamp	String	Details: Time when a cluster was created Constraints: The value is automatically recorded during cluster creation and cannot be specified. Options: N/A Default value: N/A
updateTimestamp	String	Details: Time when a cluster was updated Constraints: The value is automatically recorded during cluster updates and cannot be specified. Options: N/A Default value: N/A
timezone	String	Details: Time zone of a cluster. The time zone must be included in IANA Time Zone Database , for example, Asia/Shanghai . Constraints: None Options: N/A Default value: N/A

Table 4-26 ClusterSpec

Parameter	Type	Description
category	String	Details: Cluster type Constraints: None Options: • CCE: CCE cluster. • Turbo: CCE Turbo cluster. CCE clusters support hybrid deployment of VMs and BMSs, and heterogeneous nodes such as GPU and NPU nodes, allowing you to run your containers in a secure, stable container runtime environment based on a high-performance network model. CCE Turbo clusters run on the Cloud Native 2.0 infrastructure that features hardware and software synergy to support passthrough networking, high security and reliability, and intelligent scheduling. Default value: CCE when the container network is not set to eni Turbo when the container network is set to eni
type	String	Definition Control plane node architecture Constraints N/A Range • VirtualMachine: The control plane node is an x86 server. • ARM64: The control plane node is an Arm-based Kunpeng server. Default Value VirtualMachine or ARM64 when there are not enough VMs

Parameter	Type	Description
flavor	String	Details: Cluster specifications. Specifications of clusters of v1.15 or later can be changed after they are created. For details, see Modifying Cluster Specifications. Configure this parameter based on actual service requirements. Constraints: None Options: • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s1.large: a large-scale CCE cluster with one control plane node and a maximum of 1,000 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes Special cluster specifications in DeC are as follows: - cce.dec.s1.small: a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes) - cce.dec.s1.medium: a medium-scale DeC CCE cluster with one control

Parameter	Type	Description
		plane node and a maximum of 200 worker nodes) - cce.dec.s1.large: a large-scale DeC CCE cluster with one control plane node and a maximum of 1,000 worker nodes) - cce.dec.s1.xlarge: an ultra-large-scale DeC CCE cluster with one control plane node and a maximum of 2,000 worker nodes) - cce.dec.s2.small: a small-scale DeC CCE cluster with three control plane nodes and a maximum of 50 worker nodes) - cce.dec.s2.medium: a medium-scale DeC CCE cluster with three control plane nodes and a maximum of 200 worker nodes) - cce.dec.s2.large: a large-scale DeC CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes) - cce.dec.s2.xlarge: an ultra-large-scale DeC CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes) Default value: N/A

Parameter	Type	Description
		NOTE The fields in the parameters are described as follows: • s1: specifies a cluster with one control plane node. If the control plane node is faulty, the cluster will become unavailable, but running workloads in the cluster are not affected. • s2: specifies an HA cluster with three control plane nodes. If one of the control plane nodes is faulty, the cluster is still available. • dec: specifies a DeC CCE cluster. For example, cce.dec.s1.small specifies a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes. • small: specifies that a cluster can manage a maximum of 50 worker nodes. • medium: specifies that a cluster can manage a maximum of 200 worker nodes. • large: specifies that a cluster can manage a maximum of 1,000 worker nodes. • xlarge: specifies that a cluster can manage a maximum of 2,000 worker nodes.

Parameter	Type	Description
version	String	Details: Cluster version, which mirrors the baseline version of the Kubernetes community. The latest commercial version is recommended. NOTE For details about the Kubernetes version policy, see Kubernetes Version Policy. • CCE Turbo clusters of v1.19 and later versions are commercially available. You can create clusters of the latest three versions on the CCE console. To learn which cluster versions are available, log in to the CCE console, create a cluster, and check the supported cluster versions. You can call APIs to create clusters of other versions. However, these cluster versions will be gradually terminated. For details about the support policy, see the CCE announcement. Constraints: The value must be in the format of vX.Y[Z[-rN]], for example, v1.30, v1.30.0, or v1.30.0-r0. Specifying either of these values will create a cluster of version 1.30. • X: major version in the Kubernetes community • Y: minor version in the Kubernetes community • Z: patch version in the Kubernetes community • N: CCE patch version. For details about CCE patch versions, see Patch Versions. Options: N/A Default value: • If this parameter is left empty, a cluster of the latest version is created by default. • If a baseline cluster version is specified but the R version is not specified, a cluster of the latest R version will be created by default. It

Parameter	Type	Description
		is a good practice not to specify the R version.
platformVersion	String	Details: CCE cluster platform version, which is an internal version under the cluster version (version). Platform versions are used to trace iterations in a major cluster version. They are unique within a major cluster version and recounted when the major cluster version changes. Constraints: This parameter cannot be customized, and when you create a cluster, the latest corresponding platform version is selected automatically. Options: The format of platformVersion is cce.X.Y. • X: internal feature version, which indicates changes in features, patches, or OS support in the cluster version. The value starts from 1 and increases monotonically. • Y: patch version of an internal feature version. It is used only for software package update after the feature version is released. No other modification is involved. The value starts from 0 and increases monotonically. Default value: N/A

Parameter	Type	Description
legacyVersion	String	Details: (Deprecated) Outdated version of CCE clusters, which serves no practical purpose and is only used to showcase the cluster version and platform version combination. The version number is globally unique. For example, if the cluster version is va.b and the platformVersion is cce.X.Y, the value of legacyVersion is va.b.X-rY. Constraints: None Options: N/A Default value: N/A
description	String	Details: Cluster description, for example, which purpose the cluster is intended to serve. By default, this parameter is left unspecified. To modify cluster description after a cluster is created, call the API for updating a specified cluster or go to the cluster details page on the CCE console. Constraints: Only UTF-8 encoding is supported. The length must be less than or equal to 200 bytes. Options: N/A Default value: N/A

Parameter	Type	Description
customSan	Array of strings	Details: Custom Subject Alternative Name (SAN) in the server certificate of a cluster API server, which must comply with the SSL and X509 standard formats Constraints: Duplicate names are not allowed. Options: Values in the IP address or domain name format Default value: N/A Example: SAN 1: DNS Name=example.com SAN 2: DNS Name=www.example.com SAN 3: DNS Name=example.net SAN 4: IP Address=93.184.216.34
ipv6enable	Boolean	Details: Whether a cluster supports IPv6 addresses. It is supported by clusters v1.15 and later. Constraints: After IPv6 is enabled, iptables is not supported. The VPC network model does not support IPv4/IPv6 dual-stack. Options: • true: IPv4/IPv6 dual-stack is enabled. • false: Only IPv4 is used. Default value: false
hostNetwork	HostNetwork object	Details: Node network parameters, including VPC and subnet ID. This field is mandatory because nodes in a cluster communicate with each other using a VPC. Constraints: None

Parameter	Type	Description
containerNetwork	ContainerNetwork object	Details: Container network parameters, including the container network model and container CIDR block Constraints: None
eniNetwork	EniNetwork object	Details: Configuration of the Cloud Native 2.0 network model. You need to specify this field when creating a CCE Turbo cluster. Constraints: None
serviceNetwork	ServiceNetwork object	Details: Service CIDR block, including IPv4 CIDR block Constraints: None
publicAccess	PublicAccess object	Details: Cluster API access control Constraints: None
authentication	Authentication object	Details: Configurations of the cluster authentication mode Constraints: None
billingMode	Integer	Details: Cluster billing mode Constraints: None Options: 0: pay-per-use 1: yearly/monthly Default value: 0

Parameter	Type	Description
masters	Array of MasterSpec objects	Details: Advanced settings of the control plane nodes in a cluster. You can specify AZs for them. Constraints: If this parameter is not specified, no value is returned.
kubernetesSvcIpRange	String	Details: Service CIDR blocks which kubernetes clusterIP must fall within. This parameter is available only for clusters v1.11.7 and later. If this parameter is not specified during cluster creation, the default value 10.247.0.0/16 is used. This parameter is being deprecated, so you can use serviceNetwork instead. The new field contains the IPv4 CIDR blocks. Constraints: None Options: N/A Default value: N/A
clusterTags	Array of ResourceTag objects	Details: Cluster resource tags Constraints: None

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: None Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables is used by default.
az	String	Details: AZ (being deprecated). For details about AZs supported by CCE, see Regions and Endpoints. Constraints: This field is only returned by a query API.
extendParam	ClusterExtendParam object	Details: Extended field to decide whether a cluster will span across AZs or belong to a specified enterprise project, or whether a dedicated CCE cluster is to be created Constraints: None

Parameter	Type	Description
supportIstio	Boolean	Details: Whether Istio is supported Constraints: None Options: • true: Istio is supported. • false: Istio is not supported. Default value: true
enableDistMgt	Boolean	Details: Whether to enable support for remote clouds Constraints: Only CCE Turbo clusters support this function. Options: • true: Support for remote clouds is enabled. • false: Support for remote clouds is not enabled. Default value: false
deletionProtection	Boolean	Details: Cluster deletion protection, which prevents the deletion of a cluster when this function is enabled Constraints: None Options: • true: Cluster deletion protection is enabled. • false: Cluster deletion protection is disabled. Default value: false

Parameter	Type	Description
configurationsO- verride	Array of PackageConfigur- ation objects	Details: Overrides the default component configurations in a cluster. For details about the supported components and their parameters, see Modifying Cluster Configurations. Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored.
clusterOps	ClusterOps object	Details: Cluster O&M settings Constraints: None
encryptionConfig	EncryptionConfig object	Details: Configuration of secret encryption. Only one encryption mode can be configured. By default, the CCE-managed key is used for encryption. (Users are not aware of the key being used.) Constraints: None Options: N/A Default value: N/A

Table 4-27 HostNetwork

Parameter	Type	Description
vpc	String	Details: ID of the VPC where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and view the VPC ID in the VPC details page. - Method 2: Use the VPC APIs. For details, see Querying VPCs.
subnet	String	Details: Network ID of the subnet where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the network ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
SecurityGroup	String	Details: ID of the default node security group in a cluster Constraints: None Options: N/A Default value: - If this parameter is not specified, the system automatically creates a default node security group. - If this parameter is specified, the cluster will be bound to the specified security group. NOTE To ensure proper communication, allow traffic from certain ports to the specified node security group. For details, see How Can I Configure a Security Group Rule in a Cluster?
controlPlaneSecurityGroup	String	Details: ID of the master node security group in a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Table 4-28 ContainerNetwork

Parameter	Type	Description
mode	String	Details: Container network type Constraints: Only one container network type can be selected. Options: • overlay_l2: an overlay_l2 network (container tunnel network) built for containers by using OpenVSwitch (OVS). • vpc-router: an underlay_l2 network built for containers by using IPvlan and custom VPC routes. • eni*: a Cloud Native 2.0 network. This model has integrated cloud native elastic network interfaces (ENIs), uses VPC CIDR blocks to allocate container IP addresses, and supports passthrough networking. You can use this model when creating a CCE Turbo cluster. Default value: N/A

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. If the selected CIDR block conflicts with existing ones, an error will be reported. Constraints: This parameter cannot be modified after the cluster is created. (It has been deprecated. If cidrs has been configured, the specified CIDR blocks will be ignored.) Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster. Options: Values in IPv4 CIDR format Default value: If this parameter is left blank, a unique CIDR block is randomly allocated from 172.(16–31).0.0/16 and 10.(0 16 32 48 64 80 96 112).0.0/12.
cidrs	Array of ContainerCIDR objects	Details: List of container CIDR blocks. In clusters v1.21 or later, the cidrs field is used. When the cluster network type is vpc-router, you can configure a maximum of 20 container CIDR blocks. In clusters earlier than v1.21, if the cidrs field is used, the first CIDR element in the array is used as the container CIDR block. Constraints: Once you have created a cluster that uses a container tunnel network, you cannot modify the CIDR blocks. Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster.

Table 4-29 ContainerCIDR

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. Constraints: If a CIDR block conflict occurs, an error will be reported. Options: N/A Default value: N/A

Table 4-30 EniNetwork

Parameter	Type	Description
eniSubnetId	String	Details: IPv4 subnet ID of the subnet where an elastic network interface resides Constraints: IPv6 is not supported. This field will be deprecated. Use subnets instead. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
eniSubnetCIDR	String	Details: ENI subnet CIDR block Constraints: This parameter is being deprecated. The new field subnets is recommended. Options: N/A Default value: N/A
subnets	Array of NetworkSubnet objects	Details: List of IPv4 subnet IDs Constraints: None

Table 4-31 NetworkSubnet

Parameter	Type	Description
subnetID	String	Details: ID of the IPv4 subnet where the container network interfaces are in Constraints: IPv6 is not supported. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Table 4-32 ServiceNetwork

Parameter	Type	Description
IPv4CIDR	String	Details: Value range of the Kubernetes clusterIP IPv4 CIDR block Constraints: None Options: N/A Default value: 10.247.0.0/16
IPv6CIDR	String	Details: Value range of the Kubernetes clusterIP IPv6 CIDR block Constraints: IPv6 Service CIDR blocks can be configured only for CCE Turbo clusters with IPv6 dual stack enabled. Options: N/A Default value: Default value for a CCE Turbo cluster: fc00::/112 Default value for a CCE standard cluster: fd00:1234::/120

Table 4-33 PublicAccess

Parameter	Type	Description
cidrs	Array of strings	Details: Trustlist of network CIDRs that are allowed to access cluster APIs. You are advised to allow the traffic from VPC and container network CIDRs. Constraints: This parameter is valid only when a cluster is created. Options: N/A Default value: By default, no trustlist is configured, and the value is 0.0.0.0/0 .

Table 4-34 Authentication

Parameter	Type	Description
mode	String	Details: Cluster authentication mode Constraints: None Options: - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to x509. - Clusters of Kubernetes 1.13 or later support rbac and authenticating_proxy. The parameter defaults to rbac. - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to rbac. Default value: - Clusters of Kubernetes 1.11 or earlier: x509 - Clusters of Kubernetes 1.13 or later: rbac

Parameter	Type	Description
authenticatingProxy	AuthenticatingProxy object	Details: Configuration related to the authenticating_proxy mode Constraints: This field is mandatory when the authentication mode is authenticating_proxy .

Table 4-35 AuthenticatingProxy

Parameter	Type	Description
ca	String	Details: X509 CA certificate (Base64-encoded) configured in authenticating_proxy mode Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A
cert	String	Details: Client certificate issued by the X509 CA certificate configured in authenticating_proxy mode, which is used for authentication from kube-apiserver to the extended API server. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A

Parameter	Type	Description
privateKey	String	Details: Private key of the client certificate issued by the X509 CA certificate configured in <code>authenticating_proxy</code> mode, which is used for authentication from kube-apiserver to the extended API server. The private key used by the Kubernetes cluster does not support password encryption. Use an unencrypted private key. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy. Options: Maximum size: 1 MB Default value: N/A

Table 4-36 MasterSpec

Parameter	Type	Description
availabilityZone	String	Details: AZ where a control plane node is located. The AZ name must be specified. For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: N/A Default value: N/A

Table 4-37 ResourceTag

Parameter	Type	Description
key	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: <code>_.:=-@</code> • The value cannot start with <code>_sys_</code>. Default value: N/A
value	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: <code>_.:=-@</code> Default value: N/A

Table 4-38 ClusterExtendParam

Parameter	Type	Description
clusterAZ	String	Details: AZs of master nodes in a cluster For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: • AZs supported by a specified region • multi_az: (Optional) multiple AZs, which can be configured only when a cluster with multiple master nodes is used • <i>AZs of the dedicated cloud computing pool:</i> Master nodes will be deployed in the DeC AZs. It is mandatory for dedicated CCE clusters. Default value: If no AZ is specified, an AZ is randomly allocated by default.

Parameter	Type	Description
dssMasterVolume s	String	Details: Whether the system and data disks of a master node use DSS storage pools. If this parameter is unspecified or left blank, EVS disks are used by default. Constraints: This parameter is mandatory for dedicated CCE clusters. It is in the following format: <pre><rootVol.dssPoolID>.<rootVol.volType>;<dataVol.dssPoolID>.<dataVol.volType></pre> Specifically: • rootVol specifies the system disk. dataVol specifies a data disk. • dssPoolID specifies the ID of a DSS storage pool. • volType specifies the storage volume type of the DSS storage pool, such as SAS, SSD, SATA, ESSD, GPSSD, ESSD2, and GPSSD2. Options: N/A Default value: N/A Example: c950ee97-587c-4f24-8a74-3367e3da570f.sas;6edbc2f4-1507-44f8-ac0d- eed1d2608d38.ssd NOTE This field cannot be configured for non-dedicated CCE clusters.
enterpriseProjec- tId	String	Details: ID of the enterprise project that a cluster belongs to Constraints: An enterprise project can be configured only after the enterprise project function is enabled. Options: N/A Default value: N/A

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: This parameter has been deprecated. If both this parameter and kubeProxyMode in ClusterSpec are specified, the latter is used. Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables
clusterExternalIP	String	Details: EIP of a cluster control plane node. After an EIP is bound to the control plane node, you can access the cluster management plane APIs through the EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ fixPoolMask	String	Details: Number of mask bits of the fixed IP address pool of the container networks. This parameter determines the maximum number of IP addresses that can be assigned to containers on a node. This, along with the maxPods setting during node creation, determines the maximum number of pods that can be created on a node. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: Only the vpc-router networks support this function. Options: Integers from 24 to 28 Default value: 24
decMasterFlavor	String	Details: Master node flavor in a dedicated CCE cluster Constraints: None Options: N/A Default value: N/A
dockerUmaskMode	String	Details: Default UmaskMode configuration of Docker in a cluster Constraints: None Options: • secure • normal Default value: normal

Parameter	Type	Description
kubernetes.io/ cpuManagerPolicy	String	Details: Cluster CPU management policy Constraints: None Options: • none or null: disables pods from exclusively occupying CPUs. Select this option if you want a large pool of shareable CPU cores. • static: enables pods to exclusively occupy CPUs. Select this option if your workload is sensitive to CPU cache and scheduling latency. In a CCE Turbo cluster, this setting is valid only for nodes where common containers, not secure containers, run. Default value: none
orderId	String	Details: Order ID Constraints: This parameter is returned in the response when the cluster is billed on a yearly/monthly basis with auto payment enabled (only in creation scenarios). Options: N/A Default value: N/A

Parameter	Type	Description
periodType	String	Details: Subscription term unit Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1, which is, the yearly/monthly billing mode is configured. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • month • year Default value: N/A
periodNum	Integer	Details: Number of subscription periods Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether to enable auto renewal Constraints: This field is valid only when billingMode is set to 1. Options: • true: Auto renewal is enabled. • false: Auto renewal is not enabled. Default value: false

Parameter	Type	Description
isAutoPay	String	Details: Whether to enable auto payment Constraints: This field is valid only when billingMode is set to 1 . Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: false
upgradefrom	String	Details: Records of how a cluster is upgraded to its current version Constraints: This field is only returned by a query API. Options: N/A Default value: N/A

Table 4-39 PackageConfiguration

Parameter	Type	Description
name	String	Details: Component name Constraints: None Options: N/A Default value: N/A
configurations	Array of ConfigurationItem objects	Details: Component configuration items Constraints: None

Table 4-40 ConfigurationItem

Parameter	Type	Description
name	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A
value	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Table 4-41 ClusterOps

Parameter	Type	Description
alarm	AlarmInfo object	Details: Alarm assistant settings. CCE is integrated with AOM, which enables alarm functionality, allowing for easy search and configuration of alarms. The Cloud Native Cluster Monitoring add-on sends metric rule data in CCE alarm center to AOM instances. Constraints: None

Table 4-42 AlarmInfo

Parameter	Type	Description
topics	Array of strings	Details: Contact group list. You can enter SMN topic names and configure contact groups to manage endpoints that have subscribed to alarm messages. Constraints: None
alarmRuleTemplated	String	Details: Alarm rule template ID transferred when the alarm assistant is enabled. By default, the alarm rule template in the container scenario is used. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
promInstancelD	String	Details: ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A
promEnterpriseProjectID	String	Details: Enterprise project ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A

Table 4-43 EncryptionConfig

Parameter	Type	Description
mode	String	Details: Encryption mode. Either encryption using the CCE-managed key or KMS key can be configured. Constraints: None Options: • Default: Encryption is performed using the CCE-managed key. • KMS: KMS encryption is used. Default value: Default
kmsKeyID	String	Details: KMS key ID • In the API for creating a cluster, if the mode field is set to Default, there is no need to specify this field. If the mode field is set to KMS, you can specify a value for this field. If this parameter is left blank, the default KMS key is used by default. If the default key does not exist, DEW will automatically create a default key (cce/default) for the user. To ensure proper cluster functionality, it is important to use a valid KMS key. Refrain from deleting or disabling the key before the cluster lifecycle is complete, as this may cause exceptions. Once a key has been configured for a cluster, it cannot be modified. • In the cluster query API, if the mode field is set to Default, the returned value of this field is empty. If the mode field is set to KMS, the specific key ID will be returned. Constraints: None Options: N/A Default value: N/A

Table 4-44 ClusterStatus

Parameter	Type	Description
phase	String	Details: Cluster status Constraints: None Options: • Available: The cluster is running properly. • Unavailable: The cluster is exhibiting unexpected behavior and needs to be manually deleted. • ScalingUp: Nodes are being added to the cluster. • ScalingDown: The cluster is being downsized to fewer nodes. • Creating: The cluster is being created. • Deleting: The cluster is being deleted. • Upgrading: The cluster is being upgraded. • Resizing: Cluster specifications are being changed. • ResizeFailed: Cluster specification changing is abnormal. • RollingBack: The cluster is being rolled back. • RollbackFailed: The cluster rollback is abnormal. • Hibernating: The cluster is being hibernated. • Hibernation: The cluster is in hibernation. • Freezing: The cluster is being frozen. • Frozen: The cluster has been frozen. • UnFreezing: The cluster is being unfrozen. • Awaking: The cluster is being woken up from hibernation.

Parameter	Type	Description
		• Empty: The cluster does not have any resources. This field has been deprecated. • Error: Resources in the cluster are abnormal. You can manually delete the cluster.
jobID	String	Details: ID of the job associated with the current cluster status. It supports: • ID of the associated job returned during cluster creation. You can use it to obtain the auxiliary job information for creating a cluster. • ID of the associated job returned when a cluster fails to be deleted or is being deleted. If this parameter is not empty, you can use the job ID to obtain the auxiliary job information for deleting a cluster. Constraints: None Options: N/A NOTE Job information is short-lived and is only used for tracking the progress of short-term jobs. Do not use it for cluster status query or other scenarios.
reason	String	Details: Why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A

Parameter	Type	Description
message	String	Details: Detailed information about why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A
endpoints	Array of ClusterEndpoints objects	Details: Access address of kube-apiserver in a cluster Constraints: None
isLocked	Boolean	Details: CBC resource locked Constraints: None Options: • true: The resource is locked by CBC. • false: The resource is not locked by CBC.
lockScene	String	Details: CBC resource locking scenario Constraints: None Options: N/A
lockSource	String	Details: Resource locked Constraints: None Options: N/A

Parameter	Type	Description
lockSourceId	String	Details: ID of a locked resource Constraints: None Options: N/A
deleteOption	Object	Details: Status of the configuration to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A
deleteStatus	Object	Details: Information of the status to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A

Table 4-45 ClusterEndpoints

Parameter	Type	Description
url	String	Details: Access address of kube-apiserver in a cluster Constraints: None Options: N/A

Parameter	Type	Description
type	String	Details: Type of the cluster access address Constraints: None Options: • Internal: address for internal network access • External: address for external network access

Example Requests

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes.

```
/api/v3/projects/{project_id}/clusters
{
  "kind": "Cluster",
  "apiVersion": "v3",
  "metadata": {
    "name": "cluster"
  },
  "spec": {
    "category": "CCE",
    "flavor": "cce.s2.small",
    "version": "v1.29",
    "hostNetwork": {
      "vpc": "030bfb19-5fa7-42ad-8a0d-c0721d268867",
      "subnet": "ca964acf-8468-4735-8229-97940ef6c881"
    },
    "containerNetwork": {
      "mode": "vpc-router",
      "cidr": "10.0.0.0/16"
    },
    "kubernetesSvcIprange": "10.247.0.0/16",
    "description": "",
    "billingMode": 0,
    "extendParam": {
      "kubeProxyMode": "iptables",
      "alpha.cce/fixPoolMask": "25",
      "enterpriseProjectId": "0"
    },
    "authentication": {
      "mode": "rbac"
    },
    "ipv6enable": false
  }
}
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and install ICAgent in the cluster.

```
/api/v3/projects/{project_id}/clusters
{
  "kind": "Cluster",
  "apiVersion": "v3",
```

```
"metadata" : {
  "name" : "cluster",
  "annotations" : {
    "cluster.install.addons.external/install" : "[{"addonTemplateName":"icagent"}]"
  }
},
"spec" : {
  "category" : "CCE",
  "flavor" : "cce.s2.small",
  "version" : "v1.29",
  "hostNetwork" : {
    "vpc" : "030bfb19-5fa7-42ad-8a0d-c0721d268867",
    "subnet" : "ca964acf-8468-4735-8229-97940ef6c881"
  },
  "containerNetwork" : {
    "mode" : "vpc-router",
    "cidr" : "10.0.0.0/16"
  },
  "kubernetesSvclpRange" : "10.247.0.0/16",
  "description" : "",
  "billingMode" : 0,
  "extendParam" : {
    "kubeProxyMode" : "iptables",
    "alpha.cce/fixPoolMask" : "25",
    "enterpriseProjectId" : "0"
  },
  "authentication" : {
    "mode" : "rbac"
  },
  "ipv6enable" : false
}
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and configure the custom default security group for nodes in the cluster.

```
/api/v3/projects/{project_id}/clusters

{
  "kind" : "Cluster",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "cluster"
  },
  "spec" : {
    "category" : "CCE",
    "flavor" : "cce.s2.small",
    "version" : "v1.29",
    "hostNetwork" : {
      "vpc" : "030bfb19-5fa7-42ad-8a0d-c0721d268867",
      "subnet" : "ca964acf-8468-4735-8229-97940ef6c881",
      "SecurityGroup" : "a4ef108c-2ec6-492f-a6c4-7b64e25ae490"
    },
    "containerNetwork" : {
      "mode" : "vpc-router",
      "cidr" : "10.0.0.0/16"
    },
    "kubernetesSvclpRange" : "10.247.0.0/16",
    "description" : "",
    "billingMode" : 0,
    "extendParam" : {
      "kubeProxyMode" : "iptables",
      "alpha.cce/fixPoolMask" : "25",
      "enterpriseProjectId" : "0"
    },
    "authentication" : {
      "mode" : "rbac"
    },
    "ipv6enable" : false
  }
}
```

- Create an HA CCE cluster of v1.25 with three master nodes and a maximum of 50 worker nodes billed on a pay-per-use basis.

/api/v3/projects/{project_id}/clusters

```
{
  "kind": "Cluster",
  "apiVersion": "v3",
  "metadata": {
    "name": "cluster"
  },
  "spec": {
    "category": "Turbo",
    "flavor": "cce.s2.small",
    "version": "v1.25",
    "type": "VirtualMachine",
    "hostNetwork": {
      "vpc": "030bfb19-5fa7-42ad-8a0d-c0721d268867",
      "subnet": "ca964acf-8468-4735-8229-97940ef6c881"
    },
    "containerNetwork": {
      "mode": "eni"
    },
    "eniNetwork": {
      "eniSubnetId": "861fb11d-2f0e-4c10-a98a-166dc26e4ff7",
      "eniSubnetCIDR": "192.168.0.0/24",
      "subnets": [ {
        "subnetID": "861fb11d-2f0e-4c10-a98a-166dc26e4ff7"
      } ]
    },
    "serviceNetwork": {
      "IPv4CIDR": "10.247.0.0/16"
    },
    "description": "",
    "billingMode": 0,
    "extendParam": {
      "kubeProxyMode": "iptables",
      "enterpriseProjectId": "0"
    },
    "authentication": {
      "mode": "rbac"
    },
    "ipv6enable": false
  }
}
```

- Configure the trustlist network CIDR for accessing the cluster API when creating a cluster.

/api/v3/projects/{project_id}/clusters

```
{
  "kind": "Cluster",
  "apiVersion": "v3",
  "metadata": {
    "name": "cluster"
  },
  "spec": {
    "category": "CCE",
    "flavor": "cce.s2.small",
    "version": "v1.19",
    "hostNetwork": {
      "vpc": "030bfb19-5fa7-42ad-8a0d-c0721d268867",
      "subnet": "ca964acf-8468-4735-8229-97940ef6c881"
    },
    "publicAccess": {
      "cidrs": [ "192.168.0.0/16" ]
    }
  },
}
```

```
"containerNetwork" : {
  "mode" : "vpc-router",
  "cidr" : "10.0.0.0/16"
},
"serviceNetwork" : {
  "IPv4CIDR" : "10.247.0.0/16",
  "IPv6CIDR" : "fc00::/112"
},
"kubernetesSvcIppRange" : "10.247.0.0/16",
"description" : "",
"billingMode" : 0,
"extendParam" : {
  "kubeProxyMode" : "iptables",
  "alpha.cce/fixPoolMask" : "25",
  "enterpriseProjectId" : "0"
},
"authentication" : {
  "mode" : "rbac"
},
"ipv6enable" : false
}
```

Example Responses

Status code: 201

The cluster creation job is successfully delivered.

```
{
  "kind" : "Cluster",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "cluster",
    "uid" : "1df09f9a-5b9e-11ef-8f52-0255ac10003e",
    "creationTimestamp" : "2024-08-16 07:06:53.704389459 +0000 UTC",
    "updateTimestamp" : "2024-08-16 07:06:53.704389529 +0000 UTC",
    "annotations" : {
      "jobid" : "1e50bfbe-5b9e-11ef-8f52-0255ac10003e",
      "resourceJobId" : "1df0ec6b-5b9e-11ef-8f52-0255ac10003e"
    },
    "timezone" : "Asia/Shanghai"
  },
  "spec" : {
    "publicAccess" : { },
    "category" : "CCE",
    "type" : "VirtualMachine",
    "enableAutopilot" : false,
    "flavor" : "cce.s1.small",
    "version" : "v1.29",
    "platformVersion" : "cce.4.0",
    "configurationsOverride" : [ {
      "name" : "kube-apiserver",
      "configurations" : [ {
        "name" : "support-overload",
        "value" : true
      } ]
    } ],
    "hostNetwork" : {
      "vpc" : "0538a5d0-9a65-4c1d-a8bf-e9acee237980",
      "subnet" : "bc81be88-6e34-4b02-83bd-df0a1f7672c5"
    },
    "containerNetwork" : {
      "mode" : "vpc-router",
      "cidr" : "172.17.0.0/16",
      "cidrs" : [ {
        "cidr" : "172.17.0.0/16"
      } ]
    }
  },
}
```

```
"eniNetwork" : { },
"serviceNetwork" : {
  "IPv4CIDR" : "10.247.0.0/16"
},
"authentication" : {
  "mode" : "rbac",
  "authenticatingProxy" : { }
},
"billingMode" : 0,
"kubernetesSvcIpRange" : "10.247.0.0/16",
"kubeProxyMode" : "iptables",
"extendParam" : {
  "alpha.cce/fixPoolMask" : "25",
  "enterpriseProjectId" : "0",
  "orderId" : ""
}
},
"status" : {
  "phase" : "Creating",
  "jobID" : "1e50bfbe-5b9e-11ef-8f52-0255ac10003e"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class CreateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateClusterRequest request = new CreateClusterRequest();
        Cluster body = new Cluster();
```

```
ClusterExtendParam extendParamSpec = new ClusterExtendParam();
extendParamSpec.withEnterpriseProjectId("0")
    .withKubeProxyMode("iptables")
    .withAlphaCceFixPoolMask("25");
Authentication authenticationSpec = new Authentication();
authenticationSpec.withMode("rbac");
ContainerNetwork containerNetworkSpec = new ContainerNetwork();
containerNetworkSpec.withMode(ContainerNetwork.ModeEnum.fromValue("vpc-router"))
    .withCidr("10.0.0.0/16");
HostNetwork hostNetworkSpec = new HostNetwork();
hostNetworkSpec.withVpc("030bfb19-5fa7-42ad-8a0d-c0721d268867")
    .withSubnet("ca964acf-8468-4735-8229-97940ef6c881");
ClusterSpec specbody = new ClusterSpec();
specbody.withCategory(ClusterSpec.CategoryEnum.fromValue("CCE"))
    .withFlavor("cce.s2.small")
    .withVersion("v1.29")
    .withDescription("")
    .withIpv6enable(false)
    .withHostNetwork(hostNetworkSpec)
    .withContainerNetwork(containerNetworkSpec)
    .withAuthentication(authenticationSpec)
    .withBillingMode(0)
    .withKubernetesSvcIpRange("10.247.0.0/16")
    .withExtendParam(extendParamSpec);
ClusterMetadata metadatabody = new ClusterMetadata();
metadatabody.withName("cluster");
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("Cluster");
request.withBody(body);
try {
    CreateClusterResponse response = client.createCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and install ICAgent in the cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.Map;
import java.util.HashMap;

public class CreateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
```

security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
CreateClusterRequest request = new CreateClusterRequest();
Cluster body = new Cluster();
ClusterExtendParam extendParamSpec = new ClusterExtendParam();
extendParamSpec.withEnterpriseProjectId("0")
    .withKubeProxyMode("iptables")
    .withAlphaCceFixPoolMask("25");
Authentication authenticationSpec = new Authentication();
authenticationSpec.withMode("rbac");
ContainerNetwork containerNetworkSpec = new ContainerNetwork();
containerNetworkSpec.withMode(ContainerNetwork.ModeEnum.fromValue("vpc-router"))
    .withCidr("10.0.0.0/16");
HostNetwork hostNetworkSpec = new HostNetwork();
hostNetworkSpec.withVpc("030bfb19-5fa7-42ad-8a0d-c0721d268867")
    .withSubnet("ca964acf-8468-4735-8229-97940ef6c881");
ClusterSpec specbody = new ClusterSpec();
specbody.withCategory(ClusterSpec.CategoryEnum.fromValue("CCE"))
    .withFlavor("cce.s2.small")
    .withVersion("v1.29")
    .withDescription("")
    .withIpv6enable(false)
    .withHostNetwork(hostNetworkSpec)
    .withContainerNetwork(containerNetworkSpec)
    .withAuthentication(authenticationSpec)
    .withBillingMode(0)
    .withKubernetesSvcIpRange("10.247.0.0/16")
    .withExtendParam(extendParamSpec);
Map<String, String> listMetadataAnnotations = new HashMap<>();
listMetadataAnnotations.put("cluster.install.addons.external/install",
"{{addonTemplateName:icagent}}");
ClusterMetadata metadatabody = new ClusterMetadata();
metadatabody.withName("cluster")
    .withAnnotations(listMetadataAnnotations);
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("Cluster");
request.withBody(body);
try {
    CreateClusterResponse response = client.createCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrMsg());
}
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and configure the custom default security group for nodes in the cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class CreateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        CreateClusterRequest request = new CreateClusterRequest();
        Cluster body = new Cluster();
        ClusterExtendParam extendParamSpec = new ClusterExtendParam();
        extendParamSpec.withEnterpriseProjectId("0")
            .withKubeProxyMode("iptables")
            .withAlphaCceFixPoolMask("25");
        Authentication authenticationSpec = new Authentication();
        authenticationSpec.withMode("rbac");
        ContainerNetwork containerNetworkSpec = new ContainerNetwork();
        containerNetworkSpec.withMode(ContainerNetwork.ModeEnum.fromValue("vpc-router"))
            .withCidr("10.0.0.0/16");
        HostNetwork hostNetworkSpec = new HostNetwork();
        hostNetworkSpec.withVpc("030bfb19-5fa7-42ad-8a0d-c0721d268867")
            .withSubnet("ca964acf-8468-4735-8229-97940ef6c881")
            .withSecurityGroup("a4ef108c-2ec6-492f-a6c4-7b64e25ae490");
        ClusterSpec specbody = new ClusterSpec();
        specbody.withCategory(ClusterSpec.CategoryEnum.fromValue("CCE"))
            .withFlavor("cce.s2.small")
            .withVersion("v1.29")
            .withDescription("")
            .withIpv6enable(false)
            .withHostNetwork(hostNetworkSpec)
            .withContainerNetwork(containerNetworkSpec)
            .withAuthentication(authenticationSpec)
            .withBillingMode(0)
            .withKubernetesSvcIpRange("10.247.0.0/16")
            .withExtendParam(extendParamSpec);
        ClusterMetadata metadatabody = new ClusterMetadata();
        metadatabody.withName("cluster");
    }
}
```



```
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("Cluster");
request.withBody(body);
try {
    CreateClusterResponse response = client.createCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- Create an HA CCE cluster of v1.25 with three master nodes and a maximum of 50 worker nodes billed on a pay-per-use basis.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateClusterRequest request = new CreateClusterRequest();
        Cluster body = new Cluster();
        ClusterExtendParam extendParamSpec = new ClusterExtendParam();
        extendParamSpec.withEnterpriseProjectId("0")
            .withKubeProxyMode("iptables");
        Authentication authenticationSpec = new Authentication();
        authenticationSpec.withMode("rbac");
        ServiceNetwork serviceNetworkSpec = new ServiceNetwork();
        serviceNetworkSpec.withIpv4CIDR("10.247.0.0/16");
```

```
List<NetworkSubnet> listEniNetworkSubnets = new ArrayList<>();
listEniNetworkSubnets.add(
    new NetworkSubnet()
        .withSubnetID("861fb11d-2f0e-4c10-a98a-166dc26e4ff7")
);
EniNetwork eniNetworkSpec = new EniNetwork();
eniNetworkSpec.withEniSubnetId("861fb11d-2f0e-4c10-a98a-166dc26e4ff7")
    .withEniSubnetCIDR("192.168.0.0/24")
    .withSubnets(listEniNetworkSubnets);
ContainerNetwork containerNetworkSpec = new ContainerNetwork();
containerNetworkSpec.withMode(ContainerNetwork.ModeEnum.fromValue("eni"));
HostNetwork hostNetworkSpec = new HostNetwork();
hostNetworkSpec.withVpc("030bfb19-5fa7-42ad-8a0d-c0721d268867")
    .withSubnet("ca964acf-8468-4735-8229-97940ef6c881");
ClusterSpec specbody = new ClusterSpec();
specbody.withCategory(ClusterSpec.CategoryEnum.fromValue("Turbo"))
    .withType(ClusterSpec.TypeEnum.fromValue("VirtualMachine"))
    .withFlavor("cce.s2.small")
    .withVersion("v1.25")
    .withDescription("")
    .withIpv6enable(false)
    .withHostNetwork(hostNetworkSpec)
    .withContainerNetwork(containerNetworkSpec)
    .withEniNetwork(eniNetworkSpec)
    .withServiceNetwork(serviceNetworkSpec)
    .withAuthentication(authenticationSpec)
    .withBillingMode(0)
    .withExtendParam(extendParamSpec);
ClusterMetadata metadatabody = new ClusterMetadata();
metadatabody.withName("cluster");
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("Cluster");
request.withBody(body);
try {
    CreateClusterResponse response = client.createCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrMsg());
}
}
```

- Configure the trustlist network CIDR for accessing the cluster API when creating a cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class CreateClusterSolution {

    public static void main(String[] args) {
```

// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
CreateClusterRequest request = new CreateClusterRequest();
Cluster body = new Cluster();
ClusterExtendParam extendParamSpec = new ClusterExtendParam();
extendParamSpec.withEnterpriseProjectId("0")
    .withKubeProxyMode("iptables")
    .withAlphaCceFixPoolMask("25");
Authentication authenticationSpec = new Authentication();
authenticationSpec.withMode("rbac");
ServiceNetwork serviceNetworkSpec = new ServiceNetwork();
serviceNetworkSpec.withIpv4CIDR("10.247.0.0/16");
ContainerNetwork containerNetworkSpec = new ContainerNetwork();
containerNetworkSpec.withMode(ContainerNetwork.ModeEnum.fromValue("vpc-router"))
    .withCidr("10.0.0.0/16");
HostNetwork hostNetworkSpec = new HostNetwork();
hostNetworkSpec.withVpc("030bfb19-5fa7-42ad-8a0d-c0721d268867")
    .withSubnet("ca964acf-8468-4735-8229-97940ef6c881");
ClusterSpec specbody = new ClusterSpec();
specbody.withCategory(ClusterSpec.CategoryEnum.fromValue("CCE"))
    .withFlavor("cce.s2.small")
    .withVersion("v1.19")
    .withDescription("")
    .withIpv6enable(false)
    .withHostNetwork(hostNetworkSpec)
    .withContainerNetwork(containerNetworkSpec)
    .withServiceNetwork(serviceNetworkSpec)
    .withAuthentication(authenticationSpec)
    .withBillingMode(0)
    .withKubernetesSvclpRange("10.247.0.0/16")
    .withExtendParam(extendParamSpec);
ClusterMetadata metadatabody = new ClusterMetadata();
metadatabody.withName("cluster");
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("Cluster");
request.withBody(body);
try {
    CreateClusterResponse response = client.createCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrMsg());
}
```

```
}  
}
```

Python

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes.

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    # environment variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before  
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    # environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = CceClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(CceRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = CreateClusterRequest()  
        extendParamSpec = ClusterExtendParam(  
            enterprise_project_id="0",  
            kube_proxy_mode="iptables",  
            alpha_cce_fix_pool_mask="25"  
        )  
        authenticationSpec = Authentication(  
            mode="rbac"  
        )  
        containerNetworkSpec = ContainerNetwork(  
            mode="vpc-router",  
            cidr="10.0.0.0/16"  
        )  
        hostNetworkSpec = HostNetwork(  
            vpc="030bfb19-5fa7-42ad-8a0d-c0721d268867",  
            subnet="ca964acf-8468-4735-8229-97940ef6c881"  
        )  
        specbody = ClusterSpec(  
            category="CCE",  
            flavor="cce.s2.small",  
            version="v1.29",  
            description="",  
            ipv6enable=False,  
            host_network=hostNetworkSpec,  
            container_network=containerNetworkSpec,  
            authentication=authenticationSpec,  
            billing_mode=0,  
            kubernetes_svc_ip_range="10.247.0.0/16",  
            extend_param=extendParamSpec  
        )  
        metadatabody = ClusterMetadata(  
            name="cluster"  
        )  
        request.body = Cluster(  

```

```
spec=specbody,
metadata=metadatabody,
api_version="v3",
kind="Cluster"
)
response = client.create_cluster(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and install ICAgent in the cluster.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateClusterRequest()
        extendParamSpec = ClusterExtendParam(
            enterprise_project_id="0",
            kube_proxy_mode="iptables",
            alpha_cce_fix_pool_mask="25"
        )
        authenticationSpec = Authentication(
            mode="rbac"
        )
        containerNetworkSpec = ContainerNetwork(
            mode="vpc-router",
            cidr="10.0.0.0/16"
        )
        hostNetworkSpec = HostNetwork(
            vpc="030bfb19-5fa7-42ad-8a0d-c0721d268867",
            subnet="ca964acf-8468-4735-8229-97940ef6c881"
        )
        specbody = ClusterSpec(
            category="CCE",
            flavor="cce.s2.small",
            version="v1.29",
            description="",
            ipv6enable=False,
            host_network=hostNetworkSpec,
            container_network=containerNetworkSpec,
            authentication=authenticationSpec,
```

```
        billing_mode=0,
        kubernetes_svc_ip_range="10.247.0.0/16",
        extend_param=extendParamSpec
    )
    listAnnotationsMetadata = {
        "cluster.install.addons.external/install": "[{"addonTemplateName":"icagent"}]"
    }
    metadatabody = ClusterMetadata(
        name="cluster",
        annotations=listAnnotationsMetadata
    )
    request.body = Cluster(
        spec=specbody,
        metadata=metadatabody,
        api_version="v3",
        kind="Cluster"
    )
    response = client.create_cluster(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and configure the custom default security group for nodes in the cluster.

coding: utf-8

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateClusterRequest()
        extendParamSpec = ClusterExtendParam(
            enterprise_project_id="0",
            kube_proxy_mode="iptables",
            alpha_cce_fix_pool_mask="25"
        )
        authenticationSpec = Authentication(
            mode="rbac"
        )
        containerNetworkSpec = ContainerNetwork(
            mode="vpc-router",
            cidr="10.0.0.0/16"
        )
        hostNetworkSpec = HostNetwork(
```

```
        vpc="030bfb19-5fa7-42ad-8a0d-c0721d268867",
        subnet="ca964acf-8468-4735-8229-97940ef6c881",
        security_group="a4ef108c-2ec6-492f-a6c4-7b64e25ae490"
    )
    specbody = ClusterSpec(
        category="CCE",
        flavor="cce.s2.small",
        version="v1.29",
        description="",
        ipv6enable=False,
        host_network=hostNetworkSpec,
        container_network=containerNetworkSpec,
        authentication=authenticationSpec,
        billing_mode=0,
        kubernetes_svc_ip_range="10.247.0.0/16",
        extend_param=extendParamSpec
    )
    metadatabody = ClusterMetadata(
        name="cluster"
    )
    request.body = Cluster(
        spec=specbody,
        metadata=metadatabody,
        api_version="v3",
        kind="Cluster"
    )
    response = client.create_cluster(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Create an HA CCE cluster of v1.25 with three master nodes and a maximum of 50 worker nodes billed on a pay-per-use basis.

coding: utf-8

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateClusterRequest()
        extendParamSpec = ClusterExtendParam(
            enterprise_project_id="0",
            kube_proxy_mode="iptables"
        )
        authenticationSpec = Authentication(
```

```
        mode="rbac"
    )
    serviceNetworkSpec = ServiceNetwork(
        i_pv4_cidr="10.247.0.0/16"
    )
    listSubnetsEniNetwork = [
        NetworkSubnet(
            subnet_id="861fb11d-2f0e-4c10-a98a-166dc26e4ff7"
        )
    ]
    eniNetworkSpec = EniNetwork(
        eni_subnet_id="861fb11d-2f0e-4c10-a98a-166dc26e4ff7",
        eni_subnet_cidr="192.168.0.0/24",
        subnets=listSubnetsEniNetwork
    )
    containerNetworkSpec = ContainerNetwork(
        mode="eni"
    )
    hostNetworkSpec = HostNetwork(
        vpc="030bfb19-5fa7-42ad-8a0d-c0721d268867",
        subnet="ca964acf-8468-4735-8229-97940ef6c881"
    )
    specbody = ClusterSpec(
        category="Turbo",
        type="VirtualMachine",
        flavor="cce.s2.small",
        version="v1.25",
        description="",
        ipv6enable=False,
        host_network=hostNetworkSpec,
        container_network=containerNetworkSpec,
        eni_network=eniNetworkSpec,
        service_network=serviceNetworkSpec,
        authentication=authenticationSpec,
        billing_mode=0,
        extend_param=extendParamSpec
    )
    metadatabody = ClusterMetadata(
        name="cluster"
    )
    request.body = Cluster(
        spec=specbody,
        metadata=metadatabody,
        api_version="v3",
        kind="Cluster"
    )
    response = client.create_cluster(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Configure the trustlist network CIDR for accessing the cluster API when creating a cluster.

coding: utf-8

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```

```
if __name__ == "__main__":
```

```
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
```

```
    # In this example, AK and SK are stored in environment variables for authentication. Before
```


running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateClusterRequest()
    extendParamSpec = ClusterExtendParam(
        enterprise_project_id="0",
        kube_proxy_mode="iptables",
        alpha_cce_fix_pool_mask="25"
    )
    authenticationSpec = Authentication(
        mode="rbac"
    )
    serviceNetworkSpec = ServiceNetwork(
        i_pv4_cidr="10.247.0.0/16"
    )
    containerNetworkSpec = ContainerNetwork(
        mode="vpc-router",
        cidr="10.0.0.0/16"
    )
    hostNetworkSpec = HostNetwork(
        vpc="030bfb19-5fa7-42ad-8a0d-c0721d268867",
        subnet="ca964acf-8468-4735-8229-97940ef6c881"
    )
    specbody = ClusterSpec(
        category="CCE",
        flavor="cce.s2.small",
        version="v1.19",
        description="",
        ipv6enable=False,
        host_network=hostNetworkSpec,
        container_network=containerNetworkSpec,
        service_network=serviceNetworkSpec,
        authentication=authenticationSpec,
        billing_mode=0,
        kubernetes_svc_ip_range="10.247.0.0/16",
        extend_param=extendParamSpec
    )
    metabodbody = ClusterMetadata(
        name="cluster"
    )
    request.body = Cluster(
        spec=specbody,
        metadata=metabodbody,
        api_version="v3",
        kind="Cluster"
    )
    response = client.create_cluster(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateClusterRequest{
        enterpriseProjectIdExtendParam:= "0"
        kubeProxyModeExtendParam:= "iptables"
        alphaCceFixPoolMaskExtendParam:= "25"
        extendParamSpec := &model.ClusterExtendParam{
            EnterpriseProjectId: &enterpriseProjectIdExtendParam,
            KubeProxyMode: &kubeProxyModeExtendParam,
            AlphaCceFixPoolMask: &alphaCceFixPoolMaskExtendParam,
        }
        modeAuthentication:= "rbac"
        authenticationSpec := &model.Authentication{
            Mode: &modeAuthentication,
        }
        cidrContainerNetwork:= "10.0.0.0/16"
        containerNetworkSpec := &model.ContainerNetwork{
            Mode: model.GetContainerNetworkModeEnum().VPC_ROUTER,
            Cidr: &cidrContainerNetwork,
        }
        hostNetworkSpec := &model.HostNetwork{
            Vpc: "030bfb19-5fa7-42ad-8a0d-c0721d268867",
            Subnet: "ca964acf-8468-4735-8229-97940ef6c881",
        }
        categorySpec:= model.GetClusterSpecCategoryEnum().CCE
        versionSpec:= "v1.29"
        descriptionSpec:= ""
        ipv6enableSpec:= false
        billingModeSpec:= int32(0)
        kubernetesSvcIpRangeSpec:= "10.247.0.0/16"
        specbody := &model.ClusterSpec{
            Category: &categorySpec,
            Flavor: "cce.s2.small",
            Version: &versionSpec,
            Description: &descriptionSpec,
            Ipv6enable: &ipv6enableSpec,
            HostNetwork: hostNetworkSpec,
            ContainerNetwork: containerNetworkSpec,
```

```
    Authentication: authenticationSpec,
    BillingMode: &billingModeSpec,
    KubernetesSvcIpRange: &kubernetesSvcIpRangeSpec,
    ExtendParam: extendParamSpec,
  }
  metadatabody := &model.ClusterMetadata{
    Name: "cluster",
  }
  request.Body = &model.Cluster{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Cluster",
  }
  response, err := client.CreateCluster(request)
  if err == nil {
    fmt.Printf("%v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and install ICAgent in the cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateClusterRequest{
        enterpriseProjectIdExtendParam:= "0"
        kubeProxyModeExtendParam:= "iptables"
        alphaCceFixPoolMaskExtendParam:= "25"
        extendParamSpec := &model.ClusterExtendParam{
            EnterpriseProjectId: &enterpriseProjectIdExtendParam,
            KubeProxyMode: &kubeProxyModeExtendParam,
            AlphaCceFixPoolMask: &alphaCceFixPoolMaskExtendParam,
        }
        modeAuthentication:= "rbac"
        authenticationSpec := &model.Authentication{
```

```
    Mode: &modeAuthentication,
  }
  cidrContainerNetwork:= "10.0.0.0/16"
  containerNetworkSpec := &model.ContainerNetwork{
    Mode: model.GetContainerNetworkModeEnum().VPC_ROUTER,
    Cidr: &cidrContainerNetwork,
  }
  hostNetworkSpec := &model.HostNetwork{
    Vpc: "030bfb19-5fa7-42ad-8a0d-c0721d268867",
    Subnet: "ca964acf-8468-4735-8229-97940ef6c881",
  }
  categorySpec:= model.GetClusterSpecCategoryEnum().CCE
  versionSpec:= "v1.29"
  descriptionSpec:= ""
  ipv6enableSpec:= false
  billingModeSpec:= int32(0)
  kubernetesSvcIpRangeSpec:= "10.247.0.0/16"
  specbody := &model.ClusterSpec{
    Category: &categorySpec,
    Flavor: "cce.s2.small",
    Version: &versionSpec,
    Description: &descriptionSpec,
    Ipv6enable: &ipv6enableSpec,
    HostNetwork: hostNetworkSpec,
    ContainerNetwork: containerNetworkSpec,
    Authentication: authenticationSpec,
    BillingMode: &billingModeSpec,
    KubernetesSvcIpRange: &kubernetesSvcIpRangeSpec,
    ExtendParam: extendParamSpec,
  }
  var listAnnotationsMetadata = map[string]string{
    "cluster.install.addons.external/install": "[{"addonTemplateName":"icagent"}]",
  }
  metadatabody := &model.ClusterMetadata{
    Name: "cluster",
    Annotations: listAnnotationsMetadata,
  }
  request.Body = &model.Cluster{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Cluster",
  }
  response, err := client.CreateCluster(request)
  if err == nil {
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

- Create an HA CCE cluster of v1.29 that is billed on a pay-per-use basis and has three master nodes and a maximum of 50 worker nodes and configure the custom default security group for nodes in the cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
```

running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateClusterRequest{
    enterpriseProjectIdExtendParam:= "0"
    kubeProxyModeExtendParam:= "iptables"
    alphaCceFixPoolMaskExtendParam:= "25"
    extendParamSpec := &model.ClusterExtendParam{
        EnterpriseProjectId: &enterpriseProjectIdExtendParam,
        KubeProxyMode: &kubeProxyModeExtendParam,
        AlphaCceFixPoolMask: &alphaCceFixPoolMaskExtendParam,
    }
    modeAuthentication:= "rbac"
    authenticationSpec := &model.Authentication{
        Mode: &modeAuthentication,
    }
    cidrContainerNetwork:= "10.0.0.0/16"
    containerNetworkSpec := &model.ContainerNetwork{
        Mode: model.GetContainerNetworkModeEnum().VPC_ROUTER,
        Cidr: &cidrContainerNetwork,
    }
    securityGroupHostNetwork:= "a4ef108c-2ec6-492f-a6c4-7b64e25ae490"
    hostNetworkSpec := &model.HostNetwork{
        Vpc: "030bfb19-5fa7-42ad-8a0d-c0721d268867",
        Subnet: "ca964acf-8468-4735-8229-97940ef6c881",
        SecurityGroup: &securityGroupHostNetwork,
    }
    categorySpec:= model.GetClusterSpecCategoryEnum().CCE
    versionSpec:= "v1.29"
    descriptionSpec:= ""
    ipv6enableSpec:= false
    billingModeSpec:= int32(0)
    kubernetesSvcIpRangeSpec:= "10.247.0.0/16"
    specbody := &model.ClusterSpec{
        Category: &categorySpec,
        Flavor: "cce.s2.small",
        Version: &versionSpec,
        Description: &descriptionSpec,
        Ipv6enable: &ipv6enableSpec,
        HostNetwork: hostNetworkSpec,
        ContainerNetwork: containerNetworkSpec,
        Authentication: authenticationSpec,
        BillingMode: &billingModeSpec,
        KubernetesSvcIpRange: &kubernetesSvcIpRangeSpec,
        ExtendParam: extendParamSpec,
    }
    metadatabody := &model.ClusterMetadata{
        Name: "cluster",
    }
}
request.Body = &model.Cluster{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Cluster",
}
```

```
}
response, err := client.CreateCluster(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Create an HA CCE cluster of v1.25 with three master nodes and a maximum of 50 worker nodes billed on a pay-per-use basis.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateClusterRequest{
        enterpriseProjectIdExtendParam:= "0"
        kubeProxyModeExtendParam:= "iptables"
        extendParamSpec := &model.ClusterExtendParam{
            EnterpriseProjectId: &enterpriseProjectIdExtendParam,
            KubeProxyMode: &kubeProxyModeExtendParam,
        }
        modeAuthentication:= "rbac"
        authenticationSpec := &model.Authentication{
            Mode: &modeAuthentication,
        }
        ipv4CIDRServiceNetwork:= "10.247.0.0/16"
        serviceNetworkSpec := &model.ServiceNetwork{
            IPv4CIDR: &ipv4CIDRServiceNetwork,
        }
        var listSubnetsEniNetwork = []model.NetworkSubnet{
            {
                SubnetID: "861fb11d-2f0e-4c10-a98a-166dc26e4ff7",
            },
        }
        eniSubnetCIDREniNetwork:= "192.168.0.0/24"
        eniNetworkSpec := &model.EniNetwork{
            EniSubnetId: "861fb11d-2f0e-4c10-a98a-166dc26e4ff7",
            EniSubnetCIDR: &eniSubnetCIDREniNetwork,
            Subnets: listSubnetsEniNetwork,
        }
    }
```

```
}
containerNetworkSpec := &model.ContainerNetwork{
    Mode: model.GetContainerNetworkModeEnum().ENI,
}
hostNetworkSpec := &model.HostNetwork{
    Vpc: "030bfb19-5fa7-42ad-8a0d-c0721d268867",
    Subnet: "ca964acf-8468-4735-8229-97940ef6c881",
}
categorySpec:= model.GetClusterSpecCategoryEnum().TURBO
typeSpec:= model.GetClusterSpecTypeEnum().VIRTUAL_MACHINE
versionSpec:= "v1.25"
descriptionSpec:= ""
ipv6enableSpec:= false
billingModeSpec:= int32(0)
specbody := &model.ClusterSpec{
    Category: &categorySpec,
    Type: &typeSpec,
    Flavor: "cce.s2.small",
    Version: &versionSpec,
    Description: &descriptionSpec,
    Ipv6enable: &ipv6enableSpec,
    HostNetwork: hostNetworkSpec,
    ContainerNetwork: containerNetworkSpec,
    EniNetwork: eniNetworkSpec,
    ServiceNetwork: serviceNetworkSpec,
    Authentication: authenticationSpec,
    BillingMode: &billingModeSpec,
    ExtendParam: extendParamSpec,
}
metadatabody := &model.ClusterMetadata{
    Name: "cluster",
}
request.Body = &model.Cluster{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Cluster",
}
response, err := client.CreateCluster(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Configure the trustlist network CIDR for accessing the cluster API when creating a cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateClusterRequest{
    enterpriseProjectIdExtendParam:= "0"
    kubeProxyModeExtendParam:= "iptables"
    alphaCceFixPoolMaskExtendParam:= "25"
    extendParamSpec := &model.ClusterExtendParam{
        EnterpriseProjectId: &enterpriseProjectIdExtendParam,
        KubeProxyMode: &kubeProxyModeExtendParam,
        AlphaCceFixPoolMask: &alphaCceFixPoolMaskExtendParam,
    }
    modeAuthentication:= "rbac"
    authenticationSpec := &model.Authentication{
        Mode: &modeAuthentication,
    }
    ipv4CIDRServiceNetwork:= "10.247.0.0/16"
    serviceNetworkSpec := &model.ServiceNetwork{
        IPv4CIDR: &ipv4CIDRServiceNetwork,
    }
    cidrContainerNetwork:= "10.0.0.0/16"
    containerNetworkSpec := &model.ContainerNetwork{
        Mode: model.GetContainerNetworkModeEnum().VPC_ROUTER,
        Cidr: &cidrContainerNetwork,
    }
    hostNetworkSpec := &model.HostNetwork{
        Vpc: "030bfb19-5fa7-42ad-8a0d-c0721d268867",
        Subnet: "ca964acf-8468-4735-8229-97940ef6c881",
    }
    categorySpec:= model.GetClusterSpecCategoryEnum().CCE
    versionSpec:= "v1.19"
    descriptionSpec:= ""
    ipv6enableSpec:= false
    billingModeSpec:= int32(0)
    kubernetesSvcIpRangeSpec:= "10.247.0.0/16"
    specbody := &model.ClusterSpec{
        Category: &categorySpec,
        Flavor: "cce.s2.small",
        Version: &versionSpec,
        Description: &descriptionSpec,
        Ipv6enable: &ipv6enableSpec,
        HostNetwork: hostNetworkSpec,
        ContainerNetwork: containerNetworkSpec,
        ServiceNetwork: serviceNetworkSpec,
        Authentication: authenticationSpec,
        BillingMode: &billingModeSpec,
        KubernetesSvcIpRange: &kubernetesSvcIpRangeSpec,
        ExtendParam: extendParamSpec,
    }
    metadatabody := &model.ClusterMetadata{
        Name: "cluster",
    }
}
request.Body = &model.Cluster{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Cluster",
}
response, err := client.CreateCluster(request)
if err == nil {
```



```
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	The cluster creation job is successfully delivered.

Error Codes

See [Error Codes](#).

4.2.2 Reading a Specified Cluster

Function

This API is used to obtain details about a specified cluster.

NOTE

The URL for cluster management is in the format of `https://Endpoint/uri`. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}

Table 4-46 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-47 Query Parameters

Parameter	Mandatory	Type	Description
detail	No	String	Details: Whether the details of a cluster are obtained. Constraints: N/A Options: If this parameter is set to true, the total number of nodes (totalNodesNumber), number of functional nodes (activeNodesNumber), total CPUs (totalNodesCPU), total memory size (totalNodesMemory), and installed add-ons (installedAddonInstances) will be added to the annotation. The details of installed add-ons include the add-on name (addonTemplateName), version (version), and status (status). Default value: None

Request Parameters

Table 4-48 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-49 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: • Cluster • cluster Default value: N/A
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: • v3 Default value: N/A
metadata	ClusterMetadata object	Details: Basic information about a cluster. Metadata is a collection of attributes. Constraints: None
spec	ClusterSpec object	Detailed description about the cluster. CCE creates or updates objects by defining or updating spec .
status	ClusterStatus object	Details: Cluster status Constraints: None

Table 4-50 ClusterMetadata

Parameter	Type	Description
name	String	Details: Cluster name Constraints: None Options: Enter 4 to 128 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: Cluster ID, which identifies a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. When you create a yearly/monthly cluster, no cluster ID will be returned in the response body. Options: N/A Default value: N/A

Parameter	Type	Description
alias	String	Details: Alias of a cluster name displayed on the CCE console. The name can be changed. A cluster alias must be unique. Constraints: In the request body for creating or updating a cluster, if the cluster alias is not specified or set to null, the cluster name will be used as the cluster alias. In the response body for creating a cluster or other response bodies, if the cluster alias is not configured, no value will be returned. Options: Enter 4 to 128 characters starting with a letter and not starting or ending with a hyphen (-). Only digits, letters, and hyphens (-) are allowed. Default value: N/A

Parameter	Type	Description
annotations	Map<String,String>	Details: Cluster annotations in the format of key-value pairs <pre>"annotations": { "key1" : "value1", "key2" : "value2" }</pre> Constraints: This field is not stored in the database and is only used to specify the add-ons to be installed in the cluster. Options: N/A Default value: N/A NOTE - Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. - You can install ICAgent during cluster creation by adding the key-value pair "cluster.install.addons.external/install": "[{\addonTemplateName \":\"icagent\"}]".
labels	Map<String,String>	Details: Cluster labels, in the format of key-value pairs Constraints: The value of this field is automatically generated by the system and is used by the frontend to identify the features supported by the cluster during the upgrade. Custom values are invalid. Options: N/A Default value: N/A

Parameter	Type	Description
creationTimestamp	String	Details: Time when a cluster was created Constraints: The value is automatically recorded during cluster creation and cannot be specified. Options: N/A Default value: N/A
updateTimestamp	String	Details: Time when a cluster was updated Constraints: The value is automatically recorded during cluster updates and cannot be specified. Options: N/A Default value: N/A
timezone	String	Details: Time zone of a cluster. The time zone must be included in IANA Time Zone Database , for example, Asia/Shanghai . Constraints: None Options: N/A Default value: N/A

Table 4-51 ClusterSpec

Parameter	Type	Description
category	String	Details: Cluster type Constraints: None Options: • CCE: CCE cluster. • Turbo: CCE Turbo cluster. CCE clusters support hybrid deployment of VMs and BMSs, and heterogeneous nodes such as GPU and NPU nodes, allowing you to run your containers in a secure, stable container runtime environment based on a high-performance network model. CCE Turbo clusters run on the Cloud Native 2.0 infrastructure that features hardware and software synergy to support passthrough networking, high security and reliability, and intelligent scheduling. Default value: CCE when the container network is not set to eni Turbo when the container network is set to eni
type	String	Definition Control plane node architecture Constraints N/A Range • VirtualMachine: The control plane node is an x86 server. • ARM64: The control plane node is an Arm-based Kunpeng server. Default Value VirtualMachine or ARM64 when there are not enough VMs

Parameter	Type	Description
flavor	String	Details: Cluster specifications. Specifications of clusters of v1.15 or later can be changed after they are created. For details, see Modifying Cluster Specifications. Configure this parameter based on actual service requirements. Constraints: None Options: • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s1.large: a large-scale CCE cluster with one control plane node and a maximum of 1,000 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes Special cluster specifications in DeC are as follows: - cce.dec.s1.small: a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes) - cce.dec.s1.medium: a medium-scale DeC CCE cluster with one control

Parameter	Type	Description
		plane node and a maximum of 200 worker nodes) - cce.dec.s1.large: a large-scale DeC CCE cluster with one control plane node and a maximum of 1,000 worker nodes) - cce.dec.s1.xlarge: an ultra-large-scale DeC CCE cluster with one control plane node and a maximum of 2,000 worker nodes) - cce.dec.s2.small: a small-scale DeC CCE cluster with three control plane nodes and a maximum of 50 worker nodes) - cce.dec.s2.medium: a medium-scale DeC CCE cluster with three control plane nodes and a maximum of 200 worker nodes) - cce.dec.s2.large: a large-scale DeC CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes) - cce.dec.s2.xlarge: an ultra-large-scale DeC CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes) Default value: N/A

Parameter	Type	Description
		NOTE The fields in the parameters are described as follows: • s1: specifies a cluster with one control plane node. If the control plane node is faulty, the cluster will become unavailable, but running workloads in the cluster are not affected. • s2: specifies an HA cluster with three control plane nodes. If one of the control plane nodes is faulty, the cluster is still available. • dec: specifies a DeC CCE cluster. For example, cce.dec.s1.small specifies a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes. • small: specifies that a cluster can manage a maximum of 50 worker nodes. • medium: specifies that a cluster can manage a maximum of 200 worker nodes. • large: specifies that a cluster can manage a maximum of 1,000 worker nodes. • xlarge: specifies that a cluster can manage a maximum of 2,000 worker nodes.

Parameter	Type	Description
version	String	Details: Cluster version, which mirrors the baseline version of the Kubernetes community. The latest commercial version is recommended. NOTE For details about the Kubernetes version policy, see Kubernetes Version Policy. • CCE Turbo clusters of v1.19 and later versions are commercially available. You can create clusters of the latest three versions on the CCE console. To learn which cluster versions are available, log in to the CCE console, create a cluster, and check the supported cluster versions. You can call APIs to create clusters of other versions. However, these cluster versions will be gradually terminated. For details about the support policy, see the CCE announcement. Constraints: The value must be in the format of vX.Y[Z[-rN]], for example, v1.30, v1.30.0, or v1.30.0-r0. Specifying either of these values will create a cluster of version 1.30. • X: major version in the Kubernetes community • Y: minor version in the Kubernetes community • Z: patch version in the Kubernetes community • N: CCE patch version. For details about CCE patch versions, see Patch Versions. Options: N/A Default value: • If this parameter is left empty, a cluster of the latest version is created by default. • If a baseline cluster version is specified but the R version is not specified, a cluster of the latest R version will be created by default. It

Parameter	Type	Description
		is a good practice not to specify the R version.
platformVersion	String	Details: CCE cluster platform version, which is an internal version under the cluster version (version). Platform versions are used to trace iterations in a major cluster version. They are unique within a major cluster version and recounted when the major cluster version changes. Constraints: This parameter cannot be customized, and when you create a cluster, the latest corresponding platform version is selected automatically. Options: The format of platformVersion is cce.X.Y. • X: internal feature version, which indicates changes in features, patches, or OS support in the cluster version. The value starts from 1 and increases monotonically. • Y: patch version of an internal feature version. It is used only for software package update after the feature version is released. No other modification is involved. The value starts from 0 and increases monotonically. Default value: N/A

Parameter	Type	Description
legacyVersion	String	Details: (Deprecated) Outdated version of CCE clusters, which serves no practical purpose and is only used to showcase the cluster version and platform version combination. The version number is globally unique. For example, if the cluster version is va.b and the platformVersion is cce.X.Y , the value of legacyVersion is va.b.X-rY . Constraints: None Options: N/A Default value: N/A
description	String	Details: Cluster description, for example, which purpose the cluster is intended to serve. By default, this parameter is left unspecified. To modify cluster description after a cluster is created, call the API for updating a specified cluster or go to the cluster details page on the CCE console. Constraints: Only UTF-8 encoding is supported. The length must be less than or equal to 200 bytes. Options: N/A Default value: N/A

Parameter	Type	Description
customSan	Array of strings	Details: Custom Subject Alternative Name (SAN) in the server certificate of a cluster API server, which must comply with the SSL and X509 standard formats Constraints: Duplicate names are not allowed. Options: Values in the IP address or domain name format Default value: N/A Example: SAN 1: DNS Name=example.com SAN 2: DNS Name=www.example.com SAN 3: DNS Name=example.net SAN 4: IP Address=93.184.216.34
ipv6enable	Boolean	Details: Whether a cluster supports IPv6 addresses. It is supported by clusters v1.15 and later. Constraints: After IPv6 is enabled, iptables is not supported. The VPC network model does not support IPv4/IPv6 dual-stack. Options: • true: IPv4/IPv6 dual-stack is enabled. • false: Only IPv4 is used. Default value: false
hostNetwork	HostNetwork object	Details: Node network parameters, including VPC and subnet ID. This field is mandatory because nodes in a cluster communicate with each other using a VPC. Constraints: None

Parameter	Type	Description
containerNetwork	ContainerNetwork object	Details: Container network parameters, including the container network model and container CIDR block Constraints: None
eniNetwork	EniNetwork object	Details: Configuration of the Cloud Native 2.0 network model. You need to specify this field when creating a CCE Turbo cluster. Constraints: None
serviceNetwork	ServiceNetwork object	Details: Service CIDR block, including IPv4 CIDR block Constraints: None
publicAccess	PublicAccess object	Details: Cluster API access control Constraints: None
authentication	Authentication object	Details: Configurations of the cluster authentication mode Constraints: None
billingMode	Integer	Details: Cluster billing mode Constraints: None Options: 0: pay-per-use 1: yearly/monthly Default value: 0

Parameter	Type	Description
masters	Array of MasterSpec objects	Details: Advanced settings of the control plane nodes in a cluster. You can specify AZs for them. Constraints: If this parameter is not specified, no value is returned.
kubernetesSvcIpRange	String	Details: Service CIDR blocks which kubernetes clusterIP must fall within. This parameter is available only for clusters v1.11.7 and later. If this parameter is not specified during cluster creation, the default value 10.247.0.0/16 is used. This parameter is being deprecated, so you can use serviceNetwork instead. The new field contains the IPv4 CIDR blocks. Constraints: None Options: N/A Default value: N/A
clusterTags	Array of ResourceTag objects	Details: Cluster resource tags Constraints: None

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: None Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables is used by default.
az	String	Details: AZ (being deprecated). For details about AZs supported by CCE, see Regions and Endpoints . Constraints: This field is only returned by a query API.
extendParam	ClusterExtendParam object	Details: Extended field to decide whether a cluster will span across AZs or belong to a specified enterprise project, or whether a dedicated CCE cluster is to be created Constraints: None

Parameter	Type	Description
supportIstio	Boolean	Details: Whether Istio is supported Constraints: None Options: • true: Istio is supported. • false: Istio is not supported. Default value: true
enableDistMgt	Boolean	Details: Whether to enable support for remote clouds Constraints: Only CCE Turbo clusters support this function. Options: • true: Support for remote clouds is enabled. • false: Support for remote clouds is not enabled. Default value: false
deletionProtection	Boolean	Details: Cluster deletion protection, which prevents the deletion of a cluster when this function is enabled Constraints: None Options: • true: Cluster deletion protection is enabled. • false: Cluster deletion protection is disabled. Default value: false

Parameter	Type	Description
configurationsO- verride	Array of PackageConfigur ation objects	Details: Overrides the default component configurations in a cluster. For details about the supported components and their parameters, see Modifying Cluster Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored.
clusterOps	ClusterOps object	Details: Cluster O&M settings Constraints: None
encryptionConfig	EncryptionConfig object	Details: Configuration of secret encryption. Only one encryption mode can be configured. By default, the CCE-managed key is used for encryption. (Users are not aware of the key being used.) Constraints: None Options: N/A Default value: N/A

Table 4-52 HostNetwork

Parameter	Type	Description
vpc	String	Details: ID of the VPC where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and view the VPC ID in the VPC details page. - Method 2: Use the VPC APIs. For details, see Querying VPCs.
subnet	String	Details: Network ID of the subnet where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the network ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
SecurityGroup	String	Details: ID of the default node security group in a cluster Constraints: None Options: N/A Default value: - If this parameter is not specified, the system automatically creates a default node security group. - If this parameter is specified, the cluster will be bound to the specified security group. NOTE To ensure proper communication, allow traffic from certain ports to the specified node security group. For details, see How Can I Configure a Security Group Rule in a Cluster?
controlPlaneSecurityGroup	String	Details: ID of the master node security group in a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Table 4-53 ContainerNetwork

Parameter	Type	Description
mode	String	Details: Container network type Constraints: Only one container network type can be selected. Options: • overlay_l2: an overlay_l2 network (container tunnel network) built for containers by using OpenVSwitch (OVS). • vpc-router: an underlay_l2 network built for containers by using IPvlan and custom VPC routes. • eni*: a Cloud Native 2.0 network. This model has integrated cloud native elastic network interfaces (ENIs), uses VPC CIDR blocks to allocate container IP addresses, and supports passthrough networking. You can use this model when creating a CCE Turbo cluster. Default value: N/A

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. If the selected CIDR block conflicts with existing ones, an error will be reported. Constraints: This parameter cannot be modified after the cluster is created. (It has been deprecated. If cidrs has been configured, the specified CIDR blocks will be ignored.) Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster. Options: Values in IPv4 CIDR format Default value: If this parameter is left blank, a unique CIDR block is randomly allocated from 172.(16–31).0.0/16 and 10.(0 16 32 48 64 80 96 112).0.0/12.
cidrs	Array of ContainerCIDR objects	Details: List of container CIDR blocks. In clusters v1.21 or later, the cidrs field is used. When the cluster network type is vpc-router, you can configure a maximum of 20 container CIDR blocks. In clusters earlier than v1.21, if the cidrs field is used, the first CIDR element in the array is used as the container CIDR block. Constraints: Once you have created a cluster that uses a container tunnel network, you cannot modify the CIDR blocks. Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster.

Table 4-54 ContainerCIDR

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. Constraints: If a CIDR block conflict occurs, an error will be reported. Options: N/A Default value: N/A

Table 4-55 EniNetwork

Parameter	Type	Description
eniSubnetId	String	Details: IPv4 subnet ID of the subnet where an elastic network interface resides Constraints: IPv6 is not supported. This field will be deprecated. Use subnets instead. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
eniSubnetCIDR	String	Details: ENI subnet CIDR block Constraints: This parameter is being deprecated. The new field subnets is recommended. Options: N/A Default value: N/A
subnets	Array of NetworkSubnet objects	Details: List of IPv4 subnet IDs Constraints: None

Table 4-56 NetworkSubnet

Parameter	Type	Description
subnetID	String	Details: ID of the IPv4 subnet where the container network interfaces are in Constraints: IPv6 is not supported. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Table 4-57 ServiceNetwork

Parameter	Type	Description
IPv4CIDR	String	Details: Value range of the Kubernetes clusterIP IPv4 CIDR block Constraints: None Options: N/A Default value: 10.247.0.0/16
IPv6CIDR	String	Details: Value range of the Kubernetes clusterIP IPv6 CIDR block Constraints: IPv6 Service CIDR blocks can be configured only for CCE Turbo clusters with IPv6 dual stack enabled. Options: N/A Default value: Default value for a CCE Turbo cluster: fc00::/112 Default value for a CCE standard cluster: fd00:1234::/120

Table 4-58 PublicAccess

Parameter	Type	Description
cidrs	Array of strings	Details: Trustlist of network CIDRs that are allowed to access cluster APIs. You are advised to allow the traffic from VPC and container network CIDRs. Constraints: This parameter is valid only when a cluster is created. Options: N/A Default value: By default, no trustlist is configured, and the value is 0.0.0.0/0 .

Table 4-59 Authentication

Parameter	Type	Description
mode	String	Details: Cluster authentication mode Constraints: None Options: - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to x509. - Clusters of Kubernetes 1.13 or later support rbac and authenticating_proxy. The parameter defaults to rbac. - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to rbac. Default value: - Clusters of Kubernetes 1.11 or earlier: x509 - Clusters of Kubernetes 1.13 or later: rbac

Parameter	Type	Description
authenticatingProxy	AuthenticatingProxy object	Details: Configuration related to the authenticating_proxy mode Constraints: This field is mandatory when the authentication mode is authenticating_proxy .

Table 4-60 AuthenticatingProxy

Parameter	Type	Description
ca	String	Details: X509 CA certificate (Base64-encoded) configured in authenticating_proxy mode Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A
cert	String	Details: Client certificate issued by the X509 CA certificate configured in authenticating_proxy mode, which is used for authentication from kube-apiserver to the extended API server. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A

Parameter	Type	Description
privateKey	String	Details: Private key of the client certificate issued by the X509 CA certificate configured in <code>authenticating_proxy</code> mode, which is used for authentication from kube-apiserver to the extended API server. The private key used by the Kubernetes cluster does not support password encryption. Use an unencrypted private key. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy. Options: Maximum size: 1 MB Default value: N/A

Table 4-61 MasterSpec

Parameter	Type	Description
availabilityZone	String	Details: AZ where a control plane node is located. The AZ name must be specified. For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: N/A Default value: N/A

Table 4-62 ResourceTag

Parameter	Type	Description
key	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: <code>_.:=+@</code> • The value cannot start with <code>_sys_</code>. Default value: N/A
value	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: <code>_.:/=+@</code> Default value: N/A

Table 4-63 ClusterExtendParam

Parameter	Type	Description
clusterAZ	String	Details: AZs of master nodes in a cluster For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: • AZs supported by a specified region • multi_az: (Optional) multiple AZs, which can be configured only when a cluster with multiple master nodes is used • <i>AZs of the dedicated cloud computing pool:</i> Master nodes will be deployed in the DeC AZs. It is mandatory for dedicated CCE clusters. Default value: If no AZ is specified, an AZ is randomly allocated by default.

Parameter	Type	Description
dssMasterVolume s	String	Details: Whether the system and data disks of a master node use DSS storage pools. If this parameter is unspecified or left blank, EVS disks are used by default. Constraints: This parameter is mandatory for dedicated CCE clusters. It is in the following format: <pre><rootVol.dssPoolID>.<rootVol.volType>;<dataVol.dssPoolID>.<dataVol.volType></pre> Specifically: • rootVol specifies the system disk. dataVol specifies a data disk. • dssPoolID specifies the ID of a DSS storage pool. • volType specifies the storage volume type of the DSS storage pool, such as SAS, SSD, SATA, ESSD, GPSSD, ESSD2, and GPSSD2. Options: N/A Default value: N/A Example: c950ee97-587c-4f24-8a74-3367e3da570f.sas;6edbc2f4-1507-44f8-ac0d- eed1d2608d38.ssd NOTE This field cannot be configured for non-dedicated CCE clusters.
enterpriseProjec- tId	String	Details: ID of the enterprise project that a cluster belongs to Constraints: An enterprise project can be configured only after the enterprise project function is enabled. Options: N/A Default value: N/A

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: This parameter has been deprecated. If both this parameter and kubeProxyMode in ClusterSpec are specified, the latter is used. Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables
clusterExternalIP	String	Details: EIP of a cluster control plane node. After an EIP is bound to the control plane node, you can access the cluster management plane APIs through the EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ fixPoolMask	String	Details: Number of mask bits of the fixed IP address pool of the container networks. This parameter determines the maximum number of IP addresses that can be assigned to containers on a node. This, along with the maxPods setting during node creation, determines the maximum number of pods that can be created on a node. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: Only the vpc-router networks support this function. Options: Integers from 24 to 28 Default value: 24
decMasterFlavor	String	Details: Master node flavor in a dedicated CCE cluster Constraints: None Options: N/A Default value: N/A
dockerUmaskMode	String	Details: Default UmaskMode configuration of Docker in a cluster Constraints: None Options: • secure • normal Default value: normal

Parameter	Type	Description
kubernetes.io/ cpuManagerPolicy	String	Details: Cluster CPU management policy Constraints: None Options: • none or null: disables pods from exclusively occupying CPUs. Select this option if you want a large pool of shareable CPU cores. • static: enables pods to exclusively occupy CPUs. Select this option if your workload is sensitive to CPU cache and scheduling latency. In a CCE Turbo cluster, this setting is valid only for nodes where common containers, not secure containers, run. Default value: none
orderId	String	Details: Order ID Constraints: This parameter is returned in the response when the cluster is billed on a yearly/monthly basis with auto payment enabled (only in creation scenarios). Options: N/A Default value: N/A

Parameter	Type	Description
periodType	String	Details: Subscription term unit Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1, which is, the yearly/monthly billing mode is configured. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • month • year Default value: N/A
periodNum	Integer	Details: Number of subscription periods Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether to enable auto renewal Constraints: This field is valid only when billingMode is set to 1. Options: • true: Auto renewal is enabled. • false: Auto renewal is not enabled. Default value: false

Parameter	Type	Description
isAutoPay	String	Details: Whether to enable auto payment Constraints: This field is valid only when billingMode is set to 1 . Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: false
upgradefrom	String	Details: Records of how a cluster is upgraded to its current version Constraints: This field is only returned by a query API. Options: N/A Default value: N/A

Table 4-64 PackageConfiguration

Parameter	Type	Description
name	String	Details: Component name Constraints: None Options: N/A Default value: N/A
configurations	Array of ConfigurationItem objects	Details: Component configuration items Constraints: None

Table 4-65 ConfigurationItem

Parameter	Type	Description
name	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A
value	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Table 4-66 ClusterOps

Parameter	Type	Description
alarm	AlarmInfo object	Details: Alarm assistant settings. CCE is integrated with AOM, which enables alarm functionality, allowing for easy search and configuration of alarms. The Cloud Native Cluster Monitoring add-on sends metric rule data in CCE alarm center to AOM instances. Constraints: None

Table 4-67 AlarmInfo

Parameter	Type	Description
topics	Array of strings	Details: Contact group list. You can enter SMN topic names and configure contact groups to manage endpoints that have subscribed to alarm messages. Constraints: None
alarmRuleTemplated	String	Details: Alarm rule template ID transferred when the alarm assistant is enabled. By default, the alarm rule template in the container scenario is used. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
promInstancelD	String	Details: ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A
promEnterpriseProjectID	String	Details: Enterprise project ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A

Table 4-68 EncryptionConfig

Parameter	Type	Description
mode	String	Details: Encryption mode. Either encryption using the CCE-managed key or KMS key can be configured. Constraints: None Options: • Default: Encryption is performed using the CCE-managed key. • KMS: KMS encryption is used. Default value: Default
kmsKeyID	String	Details: KMS key ID • In the API for creating a cluster, if the mode field is set to Default, there is no need to specify this field. If the mode field is set to KMS, you can specify a value for this field. If this parameter is left blank, the default KMS key is used by default. If the default key does not exist, DEW will automatically create a default key (cce/default) for the user. To ensure proper cluster functionality, it is important to use a valid KMS key. Refrain from deleting or disabling the key before the cluster lifecycle is complete, as this may cause exceptions. Once a key has been configured for a cluster, it cannot be modified. • In the cluster query API, if the mode field is set to Default, the returned value of this field is empty. If the mode field is set to KMS, the specific key ID will be returned. Constraints: None Options: N/A Default value: N/A

Table 4-69 ClusterStatus

Parameter	Type	Description
phase	String	Details: Cluster status Constraints: None Options: • Available: The cluster is running properly. • Unavailable: The cluster is exhibiting unexpected behavior and needs to be manually deleted. • ScalingUp: Nodes are being added to the cluster. • ScalingDown: The cluster is being downsized to fewer nodes. • Creating: The cluster is being created. • Deleting: The cluster is being deleted. • Upgrading: The cluster is being upgraded. • Resizing: Cluster specifications are being changed. • ResizeFailed: Cluster specification changing is abnormal. • RollingBack: The cluster is being rolled back. • RollbackFailed: The cluster rollback is abnormal. • Hibernating: The cluster is being hibernated. • Hibernation: The cluster is in hibernation. • Freezing: The cluster is being frozen. • Frozen: The cluster has been frozen. • UnFreezing: The cluster is being unfrozen. • Awaking: The cluster is being woken up from hibernation.

Parameter	Type	Description
		• Empty: The cluster does not have any resources. This field has been deprecated. • Error: Resources in the cluster are abnormal. You can manually delete the cluster.
jobID	String	Details: ID of the job associated with the current cluster status. It supports: • ID of the associated job returned during cluster creation. You can use it to obtain the auxiliary job information for creating a cluster. • ID of the associated job returned when a cluster fails to be deleted or is being deleted. If this parameter is not empty, you can use the job ID to obtain the auxiliary job information for deleting a cluster. Constraints: None Options: N/A NOTE Job information is short-lived and is only used for tracking the progress of short-term jobs. Do not use it for cluster status query or other scenarios.
reason	String	Details: Why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A

Parameter	Type	Description
message	String	Details: Detailed information about why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A
endpoints	Array of ClusterEndpoints objects	Details: Access address of kube-apiserver in a cluster Constraints: None
isLocked	Boolean	Details: CBC resource locked Constraints: None Options: • true: The resource is locked by CBC. • false: The resource is not locked by CBC.
lockScene	String	Details: CBC resource locking scenario Constraints: None Options: N/A
lockSource	String	Details: Resource locked Constraints: None Options: N/A

Parameter	Type	Description
lockSourceId	String	Details: ID of a locked resource Constraints: None Options: N/A
deleteOption	Object	Details: Status of the configuration to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A
deleteStatus	Object	Details: Information of the status to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A

Table 4-70 ClusterEndpoints

Parameter	Type	Description
url	String	Details: Access address of kube-apiserver in a cluster Constraints: None Options: N/A

Parameter	Type	Description
type	String	Details: Type of the cluster access address Constraints: None Options: • Internal: address for internal network access • External: address for external network access

Example Requests

None

Example Responses

Status code: 200

Information about the specified cluster is successfully obtained.

```
{
  "kind" : "Cluster",
  "apiVersion" : "v3",
  "metadata" : {
    "alias" : "mycluster",
    "name" : "mycluster",
    "uid" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
    "creationTimestamp" : "2018-08-02 03:48:58.968214406 +0000 UTC",
    "updateTimestamp" : "2018-08-02 04:05:29.386391813 +0000 UTC"
  },
  "spec" : {
    "type" : "VirtualMachine",
    "flavor" : "cce.s1.small",
    "version" : "v1.25",
    "platformVersion" : "cce.6.0",
    "legacyVersion" : "v1.25.6-r0",
    "description" : "this is a demo cluster",
    "customSan" : [ "192.168.1.0", "example.com" ],
    "hostNetwork" : {
      "vpc" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
      "subnet" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
      "controlPlaneSecurityGroup" : "14834251-ac69-460a-bfbd-7ac84274c52b"
    },
    "containerNetwork" : {
      "mode" : "overlay_l2",
      "cidr" : "172.16.0.0/16"
    },
    "authentication" : {
      "mode" : "x509",
      "authenticatingProxy" : { }
    },
    "billingMode" : 0,
    "deletionProtection" : false
  },
  "status" : {
    "phase" : "Available",
    "endpoints" : [ {
```

```
"url" : "https://192.168.0.11:5443",  
"type" : "Internal"  
}]  
}  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
public class ShowClusterSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        CceClient client = CceClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ShowClusterRequest request = new ShowClusterRequest();  
        request.withClusterId("{cluster_id}");  
        try {  
            ShowClusterResponse response = client.showCluster(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowClusterRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowClusterRequest{}
```

```
request.ClusterId = "{cluster_id}"
response, err := client.ShowCluster(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Information about the specified cluster is successfully obtained.

Error Codes

See [Error Codes](#).

4.2.3 Listing Clusters in a Specified Project

Function

This API is used to obtain details about all clusters in a specified project.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters

Table 4-71 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Table 4-72 Query Parameters

Parameter	Mandatory	Type	Description
detail	No	String	Details: Whether the details of a cluster are obtained Constraints: N/A Options: If this parameter is set to true, the total number of nodes (totalNodesNumber), number of functional nodes (activeNodesNumber), total CPUs (totalNodesCPU), total memory size (totalNodesMemory), and installed add-ons (installedAddonInstances) will be added to the annotation. The details of installed add-ons include the add-on name (addonTemplateName), version (version), status (status), and whether node pool scaling is supported (supportNodePoolScaleGroup). Default value: None

Parameter	Mandatory	Type	Description
status	No	String	Details: Cluster status Constraints: N/A Options: • Available: The cluster is running properly. • Unavailable: An error occurred in the cluster, which made it unavailable and now requires manual deletion. • ScalingUp: Nodes are being added to the cluster. • ScalingDown: The cluster is being downsized to fewer nodes. • Creating: The cluster is being created. • Deleting: The cluster is being deleted. • Upgrading: The cluster is being upgraded. • Resizing: Cluster specifications are being modified. • ResizeFailed: Modify cluster specifications failed. • RollingBack: The cluster is being rolled back. • RollbackFailed: Roll back the cluster failed. • Hibernating: The cluster is being hibernated. • Hibernation: The cluster is in hibernation. • Freezing: The cluster is being frozen. • Frozen: The cluster has been frozen. • UnFreezing: The cluster is being unfrozen.

Parameter	Mandatory	Type	Description
			• Awaking: The cluster is being woken up from hibernation. • Empty: The cluster does not have any resources. This field has been deprecated. • Error: An error occurred in cluster resources, which require manual deletion. Default value: N/A
type	No	String	Details: Cluster type Constraints: N/A Options: • VirtualMachine: CCE cluster • ARM64: Kunpeng cluster Default value: N/A
version	No	String	Details: Obtain cluster versions. Constraints: N/A Value range: Cluster versions Default value: None

Request Parameters

Table 4-73 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-74 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: Cluster
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
items	Array of Cluster objects	Details: Details of all clusters in the current project. You can find clusters based on the value of items.metadata.name . Constraints: N/A Options: N/A Default value: N/A

Table 4-75 Cluster

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: • Cluster • cluster Default value: N/A
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: • v3 Default value: N/A
metadata	ClusterMetadata object	Details: Basic information about a cluster. Metadata is a collection of attributes. Constraints: None
spec	ClusterSpec object	Details: Detailed description about a cluster. CCE creates or updates objects by defining or updating spec . Constraints: None
status	ClusterStatus object	Details: Cluster status and ID of a cluster creation job Constraints: None

Table 4-76 ClusterMetadata

Parameter	Type	Description
name	String	Details: Cluster name Constraints: None Options: Enter 4 to 128 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: Cluster ID, which identifies a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. When you create a yearly/monthly cluster, no cluster ID will be returned in the response body. Options: N/A Default value: N/A

Parameter	Type	Description
alias	String	Details: Alias of a cluster name displayed on the CCE console. The name can be changed. A cluster alias must be unique. Constraints: In the request body for creating or updating a cluster, if the cluster alias is not specified or set to null, the cluster name will be used as the cluster alias. In the response body for creating a cluster or other response bodies, if the cluster alias is not configured, no value will be returned. Options: Enter 4 to 128 characters starting with a letter and not starting or ending with a hyphen (-). Only digits, letters, and hyphens (-) are allowed. Default value: N/A

Parameter	Type	Description
annotations	Map<String,String>	Details: Cluster annotations in the format of key-value pairs <pre>"annotations": { "key1" : "value1", "key2" : "value2" }</pre> Constraints: This field is not stored in the database and is only used to specify the add-ons to be installed in the cluster. Options: N/A Default value: N/A NOTE - Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. - You can install ICAgent during cluster creation by adding the key-value pair "cluster.install.addons.external/install": "[{\addonTemplateName \":\"icagent\"}]".
labels	Map<String,String>	Details: Cluster labels, in the format of key-value pairs Constraints: The value of this field is automatically generated by the system and is used by the frontend to identify the features supported by the cluster during the upgrade. Custom values are invalid. Options: N/A Default value: N/A

Parameter	Type	Description
creationTimestamp	String	Details: Time when a cluster was created Constraints: The value is automatically recorded during cluster creation and cannot be specified. Options: N/A Default value: N/A
updateTimestamp	String	Details: Time when a cluster was updated Constraints: The value is automatically recorded during cluster updates and cannot be specified. Options: N/A Default value: N/A
timezone	String	Details: Time zone of a cluster. The time zone must be included in IANA Time Zone Database , for example, Asia/Shanghai . Constraints: None Options: N/A Default value: N/A

Table 4-77 ClusterSpec

Parameter	Type	Description
category	String	Details: Cluster type Constraints: None Options: • CCE: CCE cluster. • Turbo: CCE Turbo cluster. CCE clusters support hybrid deployment of VMs and BMSs, and heterogeneous nodes such as GPU and NPU nodes, allowing you to run your containers in a secure, stable container runtime environment based on a high-performance network model. CCE Turbo clusters run on the Cloud Native 2.0 infrastructure that features hardware and software synergy to support passthrough networking, high security and reliability, and intelligent scheduling. Default value: CCE when the container network is not set to eni Turbo when the container network is set to eni
type	String	Definition Control plane node architecture Constraints N/A Range • VirtualMachine: The control plane node is an x86 server. • ARM64: The control plane node is an Arm-based Kunpeng server. Default Value VirtualMachine or ARM64 when there are not enough VMs

Parameter	Type	Description
flavor	String	Details: Cluster specifications. Specifications of clusters of v1.15 or later can be changed after they are created. For details, see Modifying Cluster Specifications. Configure this parameter based on actual service requirements. Constraints: None Options: • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s1.large: a large-scale CCE cluster with one control plane node and a maximum of 1,000 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes Special cluster specifications in DeC are as follows: - cce.dec.s1.small: a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes) - cce.dec.s1.medium: a medium-scale DeC CCE cluster with one control

Parameter	Type	Description
		plane node and a maximum of 200 worker nodes) - cce.dec.s1.large: a large-scale DeC CCE cluster with one control plane node and a maximum of 1,000 worker nodes) - cce.dec.s1.xlarge: an ultra-large-scale DeC CCE cluster with one control plane node and a maximum of 2,000 worker nodes) - cce.dec.s2.small: a small-scale DeC CCE cluster with three control plane nodes and a maximum of 50 worker nodes) - cce.dec.s2.medium: a medium-scale DeC CCE cluster with three control plane nodes and a maximum of 200 worker nodes) - cce.dec.s2.large: a large-scale DeC CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes) - cce.dec.s2.xlarge: an ultra-large-scale DeC CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes) Default value: N/A

Parameter	Type	Description
		NOTE The fields in the parameters are described as follows: • s1: specifies a cluster with one control plane node. If the control plane node is faulty, the cluster will become unavailable, but running workloads in the cluster are not affected. • s2: specifies an HA cluster with three control plane nodes. If one of the control plane nodes is faulty, the cluster is still available. • dec: specifies a DeC CCE cluster. For example, cce.dec.s1.small specifies a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes. • small: specifies that a cluster can manage a maximum of 50 worker nodes. • medium: specifies that a cluster can manage a maximum of 200 worker nodes. • large: specifies that a cluster can manage a maximum of 1,000 worker nodes. • xlarge: specifies that a cluster can manage a maximum of 2,000 worker nodes.

Parameter	Type	Description
version	String	Details: Cluster version, which mirrors the baseline version of the Kubernetes community. The latest commercial version is recommended. NOTE For details about the Kubernetes version policy, see Kubernetes Version Policy. • CCE Turbo clusters of v1.19 and later versions are commercially available. You can create clusters of the latest three versions on the CCE console. To learn which cluster versions are available, log in to the CCE console, create a cluster, and check the supported cluster versions. You can call APIs to create clusters of other versions. However, these cluster versions will be gradually terminated. For details about the support policy, see the CCE announcement. Constraints: The value must be in the format of vX.Y[Z[-rN]], for example, v1.30, v1.30.0, or v1.30.0-r0. Specifying either of these values will create a cluster of version 1.30. • X: major version in the Kubernetes community • Y: minor version in the Kubernetes community • Z: patch version in the Kubernetes community • N: CCE patch version. For details about CCE patch versions, see Patch Versions. Options: N/A Default value: • If this parameter is left empty, a cluster of the latest version is created by default. • If a baseline cluster version is specified but the R version is not specified, a cluster of the latest R version will be created by default. It

Parameter	Type	Description
		is a good practice not to specify the R version.
platformVersion	String	Details: CCE cluster platform version, which is an internal version under the cluster version (version). Platform versions are used to trace iterations in a major cluster version. They are unique within a major cluster version and recounted when the major cluster version changes. Constraints: This parameter cannot be customized, and when you create a cluster, the latest corresponding platform version is selected automatically. Options: The format of platformVersion is cce.X.Y. • X: internal feature version, which indicates changes in features, patches, or OS support in the cluster version. The value starts from 1 and increases monotonically. • Y: patch version of an internal feature version. It is used only for software package update after the feature version is released. No other modification is involved. The value starts from 0 and increases monotonically. Default value: N/A

Parameter	Type	Description
legacyVersion	String	Details: (Deprecated) Outdated version of CCE clusters, which serves no practical purpose and is only used to showcase the cluster version and platform version combination. The version number is globally unique. For example, if the cluster version is va.b and the platformVersion is cce.X.Y, the value of legacyVersion is va.b.X-rY. Constraints: None Options: N/A Default value: N/A
description	String	Details: Cluster description, for example, which purpose the cluster is intended to serve. By default, this parameter is left unspecified. To modify cluster description after a cluster is created, call the API for updating a specified cluster or go to the cluster details page on the CCE console. Constraints: Only UTF-8 encoding is supported. The length must be less than or equal to 200 bytes. Options: N/A Default value: N/A

Parameter	Type	Description
customSan	Array of strings	Details: Custom Subject Alternative Name (SAN) in the server certificate of a cluster API server, which must comply with the SSL and X509 standard formats Constraints: Duplicate names are not allowed. Options: Values in the IP address or domain name format Default value: N/A Example: SAN 1: DNS Name=example.com SAN 2: DNS Name=www.example.com SAN 3: DNS Name=example.net SAN 4: IP Address=93.184.216.34
ipv6enable	Boolean	Details: Whether a cluster supports IPv6 addresses. It is supported by clusters v1.15 and later. Constraints: After IPv6 is enabled, iptables is not supported. The VPC network model does not support IPv4/IPv6 dual-stack. Options: • true: IPv4/IPv6 dual-stack is enabled. • false: Only IPv4 is used. Default value: false
hostNetwork	HostNetwork object	Details: Node network parameters, including VPC and subnet ID. This field is mandatory because nodes in a cluster communicate with each other using a VPC. Constraints: None

Parameter	Type	Description
containerNetwork	ContainerNetwork object	Details: Container network parameters, including the container network model and container CIDR block Constraints: None
eniNetwork	EniNetwork object	Details: Configuration of the Cloud Native 2.0 network model. You need to specify this field when creating a CCE Turbo cluster. Constraints: None
serviceNetwork	ServiceNetwork object	Details: Service CIDR block, including IPv4 CIDR block Constraints: None
publicAccess	PublicAccess object	Details: Cluster API access control Constraints: None
authentication	Authentication object	Details: Configurations of the cluster authentication mode Constraints: None
billingMode	Integer	Details: Cluster billing mode Constraints: None Options: 0: pay-per-use 1: yearly/monthly Default value: 0

Parameter	Type	Description
masters	Array of MasterSpec objects	Details: Advanced settings of the control plane nodes in a cluster. You can specify AZs for them. Constraints: If this parameter is not specified, no value is returned.
kubernetesSvcIpRange	String	Details: Service CIDR blocks which kubernetes clusterIP must fall within. This parameter is available only for clusters v1.11.7 and later. If this parameter is not specified during cluster creation, the default value 10.247.0.0/16 is used. This parameter is being deprecated, so you can use serviceNetwork instead. The new field contains the IPv4 CIDR blocks. Constraints: None Options: N/A Default value: N/A
clusterTags	Array of ResourceTag objects	Details: Cluster resource tags Constraints: None

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: None Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables is used by default.
az	String	Details: AZ (being deprecated). For details about AZs supported by CCE, see Regions and Endpoints . Constraints: This field is only returned by a query API.
extendParam	ClusterExtendParam object	Details: Extended field to decide whether a cluster will span across AZs or belong to a specified enterprise project, or whether a dedicated CCE cluster is to be created Constraints: None

Parameter	Type	Description
supportIstio	Boolean	Details: Whether Istio is supported Constraints: None Options: • true: Istio is supported. • false: Istio is not supported. Default value: true
enableDistMgt	Boolean	Details: Whether to enable support for remote clouds Constraints: Only CCE Turbo clusters support this function. Options: • true: Support for remote clouds is enabled. • false: Support for remote clouds is not enabled. Default value: false
deletionProtection	Boolean	Details: Cluster deletion protection, which prevents the deletion of a cluster when this function is enabled Constraints: None Options: • true: Cluster deletion protection is enabled. • false: Cluster deletion protection is disabled. Default value: false

Parameter	Type	Description
configurationsO- verride	Array of PackageConfigur- ation objects	Details: Overrides the default component configurations in a cluster. For details about the supported components and their parameters, see Modifying Cluster Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored.
clusterOps	ClusterOps object	Details: Cluster O&M settings Constraints: None
encryptionConfig	EncryptionConfig object	Details: Configuration of secret encryption. Only one encryption mode can be configured. By default, the CCE-managed key is used for encryption. (Users are not aware of the key being used.) Constraints: None Options: N/A Default value: N/A

Table 4-78 HostNetwork

Parameter	Type	Description
vpc	String	Details: ID of the VPC where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and view the VPC ID in the VPC details page. - Method 2: Use the VPC APIs. For details, see Querying VPCs.
subnet	String	Details: Network ID of the subnet where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the network ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
SecurityGroup	String	Details: ID of the default node security group in a cluster Constraints: None Options: N/A Default value: • If this parameter is not specified, the system automatically creates a default node security group. • If this parameter is specified, the cluster will be bound to the specified security group. NOTE To ensure proper communication, allow traffic from certain ports to the specified node security group. For details, see How Can I Configure a Security Group Rule in a Cluster?
controlPlaneSecurityGroup	String	Details: ID of the master node security group in a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Table 4-79 ContainerNetwork

Parameter	Type	Description
mode	String	Details: Container network type Constraints: Only one container network type can be selected. Options: • overlay_l2: an overlay_l2 network (container tunnel network) built for containers by using OpenVSwitch (OVS). • vpc-router: an underlay_l2 network built for containers by using IPvlan and custom VPC routes. • eni*: a Cloud Native 2.0 network. This model has integrated cloud native elastic network interfaces (ENIs), uses VPC CIDR blocks to allocate container IP addresses, and supports passthrough networking. You can use this model when creating a CCE Turbo cluster. Default value: N/A

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. If the selected CIDR block conflicts with existing ones, an error will be reported. Constraints: This parameter cannot be modified after the cluster is created. (It has been deprecated. If cidrs has been configured, the specified CIDR blocks will be ignored.) Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster. Options: Values in IPv4 CIDR format Default value: If this parameter is left blank, a unique CIDR block is randomly allocated from 172.(16–31).0.0/16 and 10.(0 16 32 48 64 80 96 112).0.0/12.
cidrs	Array of ContainerCIDR objects	Details: List of container CIDR blocks. In clusters v1.21 or later, the cidrs field is used. When the cluster network type is vpc-router, you can configure a maximum of 20 container CIDR blocks. In clusters earlier than v1.21, if the cidrs field is used, the first CIDR element in the array is used as the container CIDR block. Constraints: Once you have created a cluster that uses a container tunnel network, you cannot modify the CIDR blocks. Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster.

Table 4-80 ContainerCIDR

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. Constraints: If a CIDR block conflict occurs, an error will be reported. Options: N/A Default value: N/A

Table 4-81 EniNetwork

Parameter	Type	Description
eniSubnetId	String	Details: IPv4 subnet ID of the subnet where an elastic network interface resides Constraints: IPv6 is not supported. This field will be deprecated. Use subnets instead. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
eniSubnetCIDR	String	Details: ENI subnet CIDR block Constraints: This parameter is being deprecated. The new field subnets is recommended. Options: N/A Default value: N/A
subnets	Array of NetworkSubnet objects	Details: List of IPv4 subnet IDs Constraints: None

Table 4-82 NetworkSubnet

Parameter	Type	Description
subnetID	String	Details: ID of the IPv4 subnet where the container network interfaces are in Constraints: IPv6 is not supported. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Table 4-83 ServiceNetwork

Parameter	Type	Description
IPv4CIDR	String	Details: Value range of the Kubernetes clusterIP IPv4 CIDR block Constraints: None Options: N/A Default value: 10.247.0.0/16
IPv6CIDR	String	Details: Value range of the Kubernetes clusterIP IPv6 CIDR block Constraints: IPv6 Service CIDR blocks can be configured only for CCE Turbo clusters with IPv6 dual stack enabled. Options: N/A Default value: Default value for a CCE Turbo cluster: fc00::/112 Default value for a CCE standard cluster: fd00:1234::/120

Table 4-84 PublicAccess

Parameter	Type	Description
cidrs	Array of strings	Details: Trustlist of network CIDRs that are allowed to access cluster APIs. You are advised to allow the traffic from VPC and container network CIDRs. Constraints: This parameter is valid only when a cluster is created. Options: N/A Default value: By default, no trustlist is configured, and the value is 0.0.0.0/0 .

Table 4-85 Authentication

Parameter	Type	Description
mode	String	Details: Cluster authentication mode Constraints: None Options: - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to x509. - Clusters of Kubernetes 1.13 or later support rbac and authenticating_proxy. The parameter defaults to rbac. - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to rbac. Default value: - Clusters of Kubernetes 1.11 or earlier: x509 - Clusters of Kubernetes 1.13 or later: rbac

Parameter	Type	Description
authenticatingProxy	AuthenticatingProxy object	Details: Configuration related to the authenticating_proxy mode Constraints: This field is mandatory when the authentication mode is authenticating_proxy .

Table 4-86 AuthenticatingProxy

Parameter	Type	Description
ca	String	Details: X509 CA certificate (Base64-encoded) configured in authenticating_proxy mode Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A
cert	String	Details: Client certificate issued by the X509 CA certificate configured in authenticating_proxy mode, which is used for authentication from kube-apiserver to the extended API server. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A

Parameter	Type	Description
privateKey	String	Details: Private key of the client certificate issued by the X509 CA certificate configured in <code>authenticating_proxy</code> mode, which is used for authentication from kube-apiserver to the extended API server. The private key used by the Kubernetes cluster does not support password encryption. Use an unencrypted private key. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy. Options: Maximum size: 1 MB Default value: N/A

Table 4-87 MasterSpec

Parameter	Type	Description
availabilityZone	String	Details: AZ where a control plane node is located. The AZ name must be specified. For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: N/A Default value: N/A

Table 4-88 ResourceTag

Parameter	Type	Description
key	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: <code>_.:=+@</code> • The value cannot start with <code>_sys_</code>. Default value: N/A
value	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: <code>_.:/=+@</code> Default value: N/A

Table 4-89 ClusterExtendParam

Parameter	Type	Description
clusterAZ	String	Details: AZs of master nodes in a cluster For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: • AZs supported by a specified region • multi_az: (Optional) multiple AZs, which can be configured only when a cluster with multiple master nodes is used • <i>AZs of the dedicated cloud computing pool:</i> Master nodes will be deployed in the DeC AZs. It is mandatory for dedicated CCE clusters. Default value: If no AZ is specified, an AZ is randomly allocated by default.

Parameter	Type	Description
dssMasterVolume s	String	Details: Whether the system and data disks of a master node use DSS storage pools. If this parameter is unspecified or left blank, EVS disks are used by default. Constraints: This parameter is mandatory for dedicated CCE clusters. It is in the following format: <pre><rootVol.dssPoolID>.<rootVol.volType>;<dataVol.dssPoolID>.<dataVol.volType></pre> Specifically: • rootVol specifies the system disk. dataVol specifies a data disk. • dssPoolID specifies the ID of a DSS storage pool. • volType specifies the storage volume type of the DSS storage pool, such as SAS, SSD, SATA, ESSD, GPSSD, ESSD2, and GPSSD2. Options: N/A Default value: N/A Example: <pre>c950ee97-587c-4f24-8a74-3367e3da570f.sas;6edbc2f4-1507-44f8-ac0d- eed1d2608d38.ssd</pre> NOTE This field cannot be configured for non-dedicated CCE clusters.
enterpriseProjec- tId	String	Details: ID of the enterprise project that a cluster belongs to Constraints: An enterprise project can be configured only after the enterprise project function is enabled. Options: N/A Default value: N/A

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: This parameter has been deprecated. If both this parameter and kubeProxyMode in ClusterSpec are specified, the latter is used. Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables
clusterExternalIP	String	Details: EIP of a cluster control plane node. After an EIP is bound to the control plane node, you can access the cluster management plane APIs through the EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ fixPoolMask	String	Details: Number of mask bits of the fixed IP address pool of the container networks. This parameter determines the maximum number of IP addresses that can be assigned to containers on a node. This, along with the maxPods setting during node creation, determines the maximum number of pods that can be created on a node. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: Only the vpc-router networks support this function. Options: Integers from 24 to 28 Default value: 24
decMasterFlavor	String	Details: Master node flavor in a dedicated CCE cluster Constraints: None Options: N/A Default value: N/A
dockerUmaskMode	String	Details: Default UmaskMode configuration of Docker in a cluster Constraints: None Options: • secure • normal Default value: normal

Parameter	Type	Description
kubernetes.io/ cpuManagerPolicy	String	Details: Cluster CPU management policy Constraints: None Options: • none or null: disables pods from exclusively occupying CPUs. Select this option if you want a large pool of shareable CPU cores. • static: enables pods to exclusively occupy CPUs. Select this option if your workload is sensitive to CPU cache and scheduling latency. In a CCE Turbo cluster, this setting is valid only for nodes where common containers, not secure containers, run. Default value: none
orderId	String	Details: Order ID Constraints: This parameter is returned in the response when the cluster is billed on a yearly/monthly basis with auto payment enabled (only in creation scenarios). Options: N/A Default value: N/A

Parameter	Type	Description
periodType	String	Details: Subscription term unit Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1, which is, the yearly/monthly billing mode is configured. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • month • year Default value: N/A
periodNum	Integer	Details: Number of subscription periods Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether to enable auto renewal Constraints: This field is valid only when billingMode is set to 1. Options: • true: Auto renewal is enabled. • false: Auto renewal is not enabled. Default value: false

Parameter	Type	Description
isAutoPay	String	Details: Whether to enable auto payment Constraints: This field is valid only when billingMode is set to 1 . Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: false
upgradefrom	String	Details: Records of how a cluster is upgraded to its current version Constraints: This field is only returned by a query API. Options: N/A Default value: N/A

Table 4-90 PackageConfiguration

Parameter	Type	Description
name	String	Details: Component name Constraints: None Options: N/A Default value: N/A
configurations	Array of ConfigurationItem objects	Details: Component configuration items Constraints: None

Table 4-91 ConfigurationItem

Parameter	Type	Description
name	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A
value	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Table 4-92 ClusterOps

Parameter	Type	Description
alarm	AlarmInfo object	Details: Alarm assistant settings. CCE is integrated with AOM, which enables alarm functionality, allowing for easy search and configuration of alarms. The Cloud Native Cluster Monitoring add-on sends metric rule data in CCE alarm center to AOM instances. Constraints: None

Table 4-93 AlarmInfo

Parameter	Type	Description
topics	Array of strings	Details: Contact group list. You can enter SMN topic names and configure contact groups to manage endpoints that have subscribed to alarm messages. Constraints: None
alarmRuleTemplated	String	Details: Alarm rule template ID transferred when the alarm assistant is enabled. By default, the alarm rule template in the container scenario is used. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
promInstanceID	String	Details: ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A
promEnterpriseProjectID	String	Details: Enterprise project ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A

Table 4-94 EncryptionConfig

Parameter	Type	Description
mode	String	Details: Encryption mode. Either encryption using the CCE-managed key or KMS key can be configured. Constraints: None Options: • Default: Encryption is performed using the CCE-managed key. • KMS: KMS encryption is used. Default value: Default
kmsKeyID	String	Details: KMS key ID • In the API for creating a cluster, if the mode field is set to Default, there is no need to specify this field. If the mode field is set to KMS, you can specify a value for this field. If this parameter is left blank, the default KMS key is used by default. If the default key does not exist, DEW will automatically create a default key (cce/default) for the user. To ensure proper cluster functionality, it is important to use a valid KMS key. Refrain from deleting or disabling the key before the cluster lifecycle is complete, as this may cause exceptions. Once a key has been configured for a cluster, it cannot be modified. • In the cluster query API, if the mode field is set to Default, the returned value of this field is empty. If the mode field is set to KMS, the specific key ID will be returned. Constraints: None Options: N/A Default value: N/A

Table 4-95 ClusterStatus

Parameter	Type	Description
phase	String	Details: Cluster status Constraints: None Options: • Available: The cluster is running properly. • Unavailable: The cluster is exhibiting unexpected behavior and needs to be manually deleted. • ScalingUp: Nodes are being added to the cluster. • ScalingDown: The cluster is being downsized to fewer nodes. • Creating: The cluster is being created. • Deleting: The cluster is being deleted. • Upgrading: The cluster is being upgraded. • Resizing: Cluster specifications are being changed. • ResizeFailed: Cluster specification changing is abnormal. • RollingBack: The cluster is being rolled back. • RollbackFailed: The cluster rollback is abnormal. • Hibernating: The cluster is being hibernated. • Hibernation: The cluster is in hibernation. • Freezing: The cluster is being frozen. • Frozen: The cluster has been frozen. • UnFreezing: The cluster is being unfrozen. • Awaking: The cluster is being woken up from hibernation.

Parameter	Type	Description
		• Empty: The cluster does not have any resources. This field has been deprecated. • Error: Resources in the cluster are abnormal. You can manually delete the cluster.
jobID	String	Details: ID of the job associated with the current cluster status. It supports: • ID of the associated job returned during cluster creation. You can use it to obtain the auxiliary job information for creating a cluster. • ID of the associated job returned when a cluster fails to be deleted or is being deleted. If this parameter is not empty, you can use the job ID to obtain the auxiliary job information for deleting a cluster. Constraints: None Options: N/A NOTE Job information is short-lived and is only used for tracking the progress of short-term jobs. Do not use it for cluster status query or other scenarios.
reason	String	Details: Why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A

Parameter	Type	Description
message	String	Details: Detailed information about why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A
endpoints	Array of ClusterEndpoints objects	Details: Access address of kube-apiserver in a cluster Constraints: None
isLocked	Boolean	Details: CBC resource locked Constraints: None Options: true: The resource is locked by CBC. false: The resource is not locked by CBC.
lockScene	String	Details: CBC resource locking scenario Constraints: None Options: N/A
lockSource	String	Details: Resource locked Constraints: None Options: N/A

Parameter	Type	Description
lockSourceId	String	Details: ID of a locked resource Constraints: None Options: N/A
deleteOption	Object	Details: Status of the configuration to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A
deleteStatus	Object	Details: Information of the status to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A

Table 4-96 ClusterEndpoints

Parameter	Type	Description
url	String	Details: Access address of kube-apiserver in a cluster Constraints: None Options: N/A

Parameter	Type	Description
type	String	Details: Type of the cluster access address Constraints: None Options: • Internal: address for internal network access • External: address for external network access

Example Requests

None

Example Responses

Status code: 200

The cluster list is successfully obtained.

```
{
  "kind" : "Cluster",
  "apiVersion" : "v3",
  "items" : [ {
    "kind" : "Cluster",
    "apiVersion" : "v3",
    "metadata" : {
      "alias" : "mycluster",
      "name" : "mycluster",
      "uid" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
      "creationTimestamp" : "2018-08-02 03:48:58.968214406 +0000 UTC",
      "updateTimestamp" : "2018-08-02 04:05:29.386391813 +0000 UTC"
    },
    "spec" : {
      "type" : "VirtualMachine",
      "flavor" : "cce.s1.small",
      "version" : "v1.25",
      "platformVersion" : "cce.6.0",
      "legacyVersion" : "v1.25.6-r0",
      "description" : "awesome cluster",
      "customSan" : [ "192.168.1.0", "example.com" ],
      "hostNetwork" : {
        "vpc" : "f0c12911-4fdb-4284-9230-7ffb0860826a",
        "subnet" : "ac274229-fd2e-4695-9f01-a0c1372b8006",
        "controlPlaneSecurityGroup" : "14834251-ac69-460a-bfbd-7ac84274c52b"
      },
      "containerNetwork" : {
        "mode" : "overlay_l2",
        "cidr" : "172.16.0.0/16"
      },
      "authentication" : {
        "mode" : "x509",
        "authenticatingProxy" : { }
      },
      "billingMode" : 0
    },
    "status" : {
```

```
"phase" : "Available",
"endpoints" : [ {
  "url" : "https://192.168.0.11:5443",
  "type" : "Internal"
}]
}
}]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListClustersSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ListClustersRequest request = new ListClustersRequest();
        try {
            ListClustersResponse response = client.listClusters(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListClustersRequest()
        response = client.list_clusters(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

```
Build()  
  
request := &model.ListClustersRequest{}  
response, err := client.ListClusters(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The cluster list is successfully obtained.

Error Codes

See [Error Codes](#).

4.2.4 Updating a Specified Cluster

Function

This API is used to update information about a specified cluster.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}

Table 4-97 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-98 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-99 Request body parameters

Parameter	Mandatory	Type	Description
spec	Yes	ClusterInformationSpec object	Details: Detailed cluster parameters Constraints: None Options: N/A Default value: N/A
metadata	No	ClusterMetadataForUpdate object	Details: Basic cluster information, including name-related fields Constraints: None Options: N/A Default value: N/A

Table 4-100 ClusterInformationSpec

Parameter	Mandatory	Type	Description
description	No	String	Details: Description of a cluster Constraints: Only clusters in the Available , ScalingUp , or ScalingDown state can be modified. Value range: 0-200. The value cannot contain the following special characters: ~\$%^&*<>[]{}()'"# Default value: None

Parameter	Mandatory	Type	Description
customSan	No	Array of strings	Details: Custom Subject Alternative Name (SAN) in the server certificate of a cluster API server, which must comply with the SSL and X509 standard formats Constraints: Duplicate names are not allowed. Options: Values in the IP address or domain name format Default value: N/A Example: SAN 1: DNS Name=example.com SAN 2: DNS Name=www.example.com SAN 3: DNS Name=example.net SAN 4: IP Address=93.184.216.34
containerNetwork	No	ContainerNetworkUpdate object	Details: Container networking parameters, including container CIDR blocks Constraints: None Options: N/A Default value: N/A
eniNetwork	No	EniNetworkUpdate object	Details: Cloud native 2.0 network settings, including the container subnets of a CCE Turbo cluster Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
deletionProtection	No	Boolean	Details: Cluster deletion protection, which prevents the deletion of a cluster when this function is enabled Constraints: None Options: • true: Cluster deletion protection is enabled. • false: Cluster deletion protection is disabled. Default value: false
hostNetwork	No	hostNetwork object	Details: Node network parameters, including the default security group settings Constraints: None Options: N/A Default value: N/A

Table 4-101 ContainerNetworkUpdate

Parameter	Mandatory	Type	Description
cidrs	No	Array of ContainerCIDR objects	Details: List of container CIDR blocks. If your cluster is of v1.21 or later and it uses a VPC network, you can add container CIDR blocks. Ensure that the total number of blocks does not exceed 20. Constraints: The configuration cannot be modified after the cluster is updated. Options: N/A Default value: N/A

Table 4-102 ContainerCIDR

Parameter	Mandatory	Type	Description
cidr	Yes	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. Constraints: If a CIDR block conflict occurs, an error will be reported. Options: N/A Default value: N/A

Table 4-103 EniNetworkUpdate

Parameter	Mandatory	Type	Description
subnets	No	Array of NetworkSubnet objects	Details: List of IPv4 subnet IDs CCE Turbo clusters v1.19.10 or later support multiple container subnets and allow for incremental updates to the subnet list. Constraints: The request body must contain all existing subnets. Options: N/A Default value: N/A

Table 4-104 NetworkSubnet

Parameter	Mandatory	Type	Description
subnetID	Yes	String	Details: ID of the IPv4 subnet where the container network interfaces are in Constraints: IPv6 is not supported. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets .

Table 4-105 hostNetwork

Parameter	Mandatory	Type	Description
SecurityGroup	No	String	Details: To ensure proper communication, the default security group of the cluster nodes needs to allow certain ports. For details, see How Can I Configure a Security Group Rule in a Cluster? Constraints: The modified security group applies only to nodes newly created or accepted. For existing nodes, you need to manually modify the security group rules for them. Options: N/A Default value: N/A

Table 4-106 ClusterMetadataForUpdate

Parameter	Mandatory	Type	Description
alias	No	String	Details: Alias of a cluster name Constraints: A cluster alias must be unique. Options: Enter 4 to 128 characters starting with a letter and not starting or ending with a hyphen (-). Only digits, letters, and hyphens (-) are allowed. If this parameter is left blank, the alias is not changed. Default value: N/A

Response Parameters

Status code: 200

Table 4-107 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: • Cluster • cluster Default value: N/A
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: • v3 Default value: N/A
metadata	ClusterMetadata object	Details: Basic information about a cluster. Metadata is a collection of attributes. Constraints: None
spec	ClusterSpec object	Details: Detailed description about a cluster. CCE creates or updates objects by defining or updating spec . Constraints: None
status	ClusterStatus object	Details: Cluster status and ID of a cluster creation job Constraints: None

Table 4-108 ClusterMetadata

Parameter	Type	Description
name	String	Details: Cluster name Constraints: None Options: Enter 4 to 128 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: Cluster ID, which identifies a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. When you create a yearly/monthly cluster, no cluster ID will be returned in the response body. Options: N/A Default value: N/A

Parameter	Type	Description
alias	String	Details: Alias of a cluster name displayed on the CCE console. The name can be changed. A cluster alias must be unique. Constraints: In the request body for creating or updating a cluster, if the cluster alias is not specified or set to null, the cluster name will be used as the cluster alias. In the response body for creating a cluster or other response bodies, if the cluster alias is not configured, no value will be returned. Options: Enter 4 to 128 characters starting with a letter and not starting or ending with a hyphen (-). Only digits, letters, and hyphens (-) are allowed. Default value: N/A

Parameter	Type	Description
annotations	Map<String,String>	Details: Cluster annotations in the format of key-value pairs <pre>"annotations": { "key1" : "value1", "key2" : "value2" }</pre> Constraints: This field is not stored in the database and is only used to specify the add-ons to be installed in the cluster. Options: N/A Default value: N/A NOTE - Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. - You can install ICAgent during cluster creation by adding the key-value pair "cluster.install.addons.external/install": "[{\addonTemplateName \":\"icagent\"}]".
labels	Map<String,String>	Details: Cluster labels, in the format of key-value pairs Constraints: The value of this field is automatically generated by the system and is used by the frontend to identify the features supported by the cluster during the upgrade. Custom values are invalid. Options: N/A Default value: N/A

Parameter	Type	Description
creationTimestamp	String	Details: Time when a cluster was created Constraints: The value is automatically recorded during cluster creation and cannot be specified. Options: N/A Default value: N/A
updateTimestamp	String	Details: Time when a cluster was updated Constraints: The value is automatically recorded during cluster updates and cannot be specified. Options: N/A Default value: N/A
timezone	String	Details: Time zone of a cluster. The time zone must be included in IANA Time Zone Database , for example, Asia/Shanghai . Constraints: None Options: N/A Default value: N/A

Table 4-109 ClusterSpec

Parameter	Type	Description
category	String	Details: Cluster type Constraints: None Options: • CCE: CCE cluster. • Turbo: CCE Turbo cluster. CCE clusters support hybrid deployment of VMs and BMSs, and heterogeneous nodes such as GPU and NPU nodes, allowing you to run your containers in a secure, stable container runtime environment based on a high-performance network model. CCE Turbo clusters run on the Cloud Native 2.0 infrastructure that features hardware and software synergy to support passthrough networking, high security and reliability, and intelligent scheduling. Default value: CCE when the container network is not set to eni Turbo when the container network is set to eni
type	String	Definition Control plane node architecture Constraints N/A Range • VirtualMachine: The control plane node is an x86 server. • ARM64: The control plane node is an Arm-based Kunpeng server. Default Value VirtualMachine or ARM64 when there are not enough VMs

Parameter	Type	Description
flavor	String	Details: Cluster specifications. Specifications of clusters of v1.15 or later can be changed after they are created. For details, see Modifying Cluster Specifications. Configure this parameter based on actual service requirements. Constraints: None Options: • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s1.large: a large-scale CCE cluster with one control plane node and a maximum of 1,000 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes Special cluster specifications in DeC are as follows: - cce.dec.s1.small: a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes) - cce.dec.s1.medium: a medium-scale DeC CCE cluster with one control

Parameter	Type	Description
		plane node and a maximum of 200 worker nodes) - cce.dec.s1.large: a large-scale DeC CCE cluster with one control plane node and a maximum of 1,000 worker nodes) - cce.dec.s1.xlarge: an ultra-large-scale DeC CCE cluster with one control plane node and a maximum of 2,000 worker nodes) - cce.dec.s2.small: a small-scale DeC CCE cluster with three control plane nodes and a maximum of 50 worker nodes) - cce.dec.s2.medium: a medium-scale DeC CCE cluster with three control plane nodes and a maximum of 200 worker nodes) - cce.dec.s2.large: a large-scale DeC CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes) - cce.dec.s2.xlarge: an ultra-large-scale DeC CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes) Default value: N/A

Parameter	Type	Description
		NOTE The fields in the parameters are described as follows: • s1: specifies a cluster with one control plane node. If the control plane node is faulty, the cluster will become unavailable, but running workloads in the cluster are not affected. • s2: specifies an HA cluster with three control plane nodes. If one of the control plane nodes is faulty, the cluster is still available. • dec: specifies a DeC CCE cluster. For example, cce.dec.s1.small specifies a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes. • small: specifies that a cluster can manage a maximum of 50 worker nodes. • medium: specifies that a cluster can manage a maximum of 200 worker nodes. • large: specifies that a cluster can manage a maximum of 1,000 worker nodes. • xlarge: specifies that a cluster can manage a maximum of 2,000 worker nodes.

Parameter	Type	Description
version	String	Details: Cluster version, which mirrors the baseline version of the Kubernetes community. The latest commercial version is recommended. NOTE For details about the Kubernetes version policy, see Kubernetes Version Policy. • CCE Turbo clusters of v1.19 and later versions are commercially available. You can create clusters of the latest three versions on the CCE console. To learn which cluster versions are available, log in to the CCE console, create a cluster, and check the supported cluster versions. You can call APIs to create clusters of other versions. However, these cluster versions will be gradually terminated. For details about the support policy, see the CCE announcement. Constraints: The value must be in the format of vX.Y[Z[-rN]], for example, v1.30, v1.30.0, or v1.30.0-r0. Specifying either of these values will create a cluster of version 1.30. • X: major version in the Kubernetes community • Y: minor version in the Kubernetes community • Z: patch version in the Kubernetes community • N: CCE patch version. For details about CCE patch versions, see Patch Versions. Options: N/A Default value: • If this parameter is left empty, a cluster of the latest version is created by default. • If a baseline cluster version is specified but the R version is not specified, a cluster of the latest R version will be created by default. It

Parameter	Type	Description
		is a good practice not to specify the R version.
platformVersion	String	Details: CCE cluster platform version, which is an internal version under the cluster version (version). Platform versions are used to trace iterations in a major cluster version. They are unique within a major cluster version and recounted when the major cluster version changes. Constraints: This parameter cannot be customized, and when you create a cluster, the latest corresponding platform version is selected automatically. Options: The format of platformVersion is cce.X.Y. • X: internal feature version, which indicates changes in features, patches, or OS support in the cluster version. The value starts from 1 and increases monotonically. • Y: patch version of an internal feature version. It is used only for software package update after the feature version is released. No other modification is involved. The value starts from 0 and increases monotonically. Default value: N/A

Parameter	Type	Description
legacyVersion	String	Details: (Deprecated) Outdated version of CCE clusters, which serves no practical purpose and is only used to showcase the cluster version and platform version combination. The version number is globally unique. For example, if the cluster version is va.b and the platformVersion is cce.X.Y, the value of legacyVersion is va.b.X-rY. Constraints: None Options: N/A Default value: N/A
description	String	Details: Cluster description, for example, which purpose the cluster is intended to serve. By default, this parameter is left unspecified. To modify cluster description after a cluster is created, call the API for updating a specified cluster or go to the cluster details page on the CCE console. Constraints: Only UTF-8 encoding is supported. The length must be less than or equal to 200 bytes. Options: N/A Default value: N/A

Parameter	Type	Description
customSan	Array of strings	Details: Custom Subject Alternative Name (SAN) in the server certificate of a cluster API server, which must comply with the SSL and X509 standard formats Constraints: Duplicate names are not allowed. Options: Values in the IP address or domain name format Default value: N/A Example: SAN 1: DNS Name=example.com SAN 2: DNS Name=www.example.com SAN 3: DNS Name=example.net SAN 4: IP Address=93.184.216.34
ipv6enable	Boolean	Details: Whether a cluster supports IPv6 addresses. It is supported by clusters v1.15 and later. Constraints: After IPv6 is enabled, iptables is not supported. The VPC network model does not support IPv4/IPv6 dual-stack. Options: • true: IPv4/IPv6 dual-stack is enabled. • false: Only IPv4 is used. Default value: false
hostNetwork	HostNetwork object	Details: Node network parameters, including VPC and subnet ID. This field is mandatory because nodes in a cluster communicate with each other using a VPC. Constraints: None

Parameter	Type	Description
containerNetwork	ContainerNetwork object	Details: Container network parameters, including the container network model and container CIDR block Constraints: None
eniNetwork	EniNetwork object	Details: Configuration of the Cloud Native 2.0 network model. You need to specify this field when creating a CCE Turbo cluster. Constraints: None
serviceNetwork	ServiceNetwork object	Details: Service CIDR block, including IPv4 CIDR block Constraints: None
publicAccess	PublicAccess object	Details: Cluster API access control Constraints: None
authentication	Authentication object	Details: Configurations of the cluster authentication mode Constraints: None
billingMode	Integer	Details: Cluster billing mode Constraints: None Options: • 0: pay-per-use • 1: yearly/monthly Default value: 0

Parameter	Type	Description
masters	Array of MasterSpec objects	Details: Advanced settings of the control plane nodes in a cluster. You can specify AZs for them. Constraints: If this parameter is not specified, no value is returned.
kubernetesSvcIpRange	String	Details: Service CIDR blocks which kubernetes clusterIP must fall within. This parameter is available only for clusters v1.11.7 and later. If this parameter is not specified during cluster creation, the default value 10.247.0.0/16 is used. This parameter is being deprecated, so you can use serviceNetwork instead. The new field contains the IPv4 CIDR blocks. Constraints: None Options: N/A Default value: N/A
clusterTags	Array of ResourceTag objects	Details: Cluster resource tags Constraints: None

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: None Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables is used by default.
az	String	Details: AZ (being deprecated). For details about AZs supported by CCE, see Regions and Endpoints . Constraints: This field is only returned by a query API.
extendParam	ClusterExtendParam object	Details: Extended field to decide whether a cluster will span across AZs or belong to a specified enterprise project, or whether a dedicated CCE cluster is to be created Constraints: None

Parameter	Type	Description
supportIstio	Boolean	Details: Whether Istio is supported Constraints: None Options: • true: Istio is supported. • false: Istio is not supported. Default value: true
enableDistMgt	Boolean	Details: Whether to enable support for remote clouds Constraints: Only CCE Turbo clusters support this function. Options: • true: Support for remote clouds is enabled. • false: Support for remote clouds is not enabled. Default value: false
deletionProtection	Boolean	Details: Cluster deletion protection, which prevents the deletion of a cluster when this function is enabled Constraints: None Options: • true: Cluster deletion protection is enabled. • false: Cluster deletion protection is disabled. Default value: false

Parameter	Type	Description
configurationsO- verride	Array of PackageConfigur- ation objects	Details: Overrides the default component configurations in a cluster. For details about the supported components and their parameters, see Modifying Cluster Configurations. Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored.
clusterOps	ClusterOps object	Details: Cluster O&M settings Constraints: None
encryptionConfig	EncryptionConfig object	Details: Configuration of secret encryption. Only one encryption mode can be configured. By default, the CCE-managed key is used for encryption. (Users are not aware of the key being used.) Constraints: None Options: N/A Default value: N/A

Table 4-110 HostNetwork

Parameter	Type	Description
vpc	String	Details: ID of the VPC where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and view the VPC ID in the VPC details page. - Method 2: Use the VPC APIs. For details, see Querying VPCs.
subnet	String	Details: Network ID of the subnet where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the network ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
SecurityGroup	String	Details: ID of the default node security group in a cluster Constraints: None Options: N/A Default value: - If this parameter is not specified, the system automatically creates a default node security group. - If this parameter is specified, the cluster will be bound to the specified security group. NOTE To ensure proper communication, allow traffic from certain ports to the specified node security group. For details, see How Can I Configure a Security Group Rule in a Cluster?
controlPlaneSecurityGroup	String	Details: ID of the master node security group in a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Table 4-111 ContainerNetwork

Parameter	Type	Description
mode	String	Details: Container network type Constraints: Only one container network type can be selected. Options: • overlay_l2: an overlay_l2 network (container tunnel network) built for containers by using OpenVSwitch (OVS). • vpc-router: an underlay_l2 network built for containers by using IPvlan and custom VPC routes. • eni*: a Cloud Native 2.0 network. This model has integrated cloud native elastic network interfaces (ENIs), uses VPC CIDR blocks to allocate container IP addresses, and supports passthrough networking. You can use this model when creating a CCE Turbo cluster. Default value: N/A

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. If the selected CIDR block conflicts with existing ones, an error will be reported. Constraints: This parameter cannot be modified after the cluster is created. (It has been deprecated. If cidrs has been configured, the specified CIDR blocks will be ignored.) Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster. Options: Values in IPv4 CIDR format Default value: If this parameter is left blank, a unique CIDR block is randomly allocated from 172.(16–31).0.0/16 and 10.(0 16 32 48 64 80 96 112).0.0/12.
cidrs	Array of ContainerCIDR objects	Details: List of container CIDR blocks. In clusters v1.21 or later, the cidrs field is used. When the cluster network type is vpc-router, you can configure a maximum of 20 container CIDR blocks. In clusters earlier than v1.21, if the cidrs field is used, the first CIDR element in the array is used as the container CIDR block. Constraints: Once you have created a cluster that uses a container tunnel network, you cannot modify the CIDR blocks. Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster.

Table 4-112 ContainerCIDR

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. Constraints: If a CIDR block conflict occurs, an error will be reported. Options: N/A Default value: N/A

Table 4-113 EniNetwork

Parameter	Type	Description
eniSubnetId	String	Details: IPv4 subnet ID of the subnet where an elastic network interface resides Constraints: IPv6 is not supported. This field will be deprecated. Use subnets instead. Options: N/A Default value: N/A To obtain the value: • Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. • Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
eniSubnetCIDR	String	Details: ENI subnet CIDR block Constraints: This parameter is being deprecated. The new field subnets is recommended. Options: N/A Default value: N/A
subnets	Array of NetworkSubnet objects	Details: List of IPv4 subnet IDs Constraints: None

Table 4-114 NetworkSubnet

Parameter	Type	Description
subnetID	String	Details: ID of the IPv4 subnet where the container network interfaces are in Constraints: IPv6 is not supported. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Table 4-115 ServiceNetwork

Parameter	Type	Description
IPv4CIDR	String	Details: Value range of the Kubernetes clusterIP IPv4 CIDR block Constraints: None Options: N/A Default value: 10.247.0.0/16
IPv6CIDR	String	Details: Value range of the Kubernetes clusterIP IPv6 CIDR block Constraints: IPv6 Service CIDR blocks can be configured only for CCE Turbo clusters with IPv6 dual stack enabled. Options: N/A Default value: Default value for a CCE Turbo cluster: fc00::/112 Default value for a CCE standard cluster: fd00:1234::/120

Table 4-116 PublicAccess

Parameter	Type	Description
cidrs	Array of strings	Details: Trustlist of network CIDRs that are allowed to access cluster APIs. You are advised to allow the traffic from VPC and container network CIDRs. Constraints: This parameter is valid only when a cluster is created. Options: N/A Default value: By default, no trustlist is configured, and the value is 0.0.0.0/0 .

Table 4-117 Authentication

Parameter	Type	Description
mode	String	Details: Cluster authentication mode Constraints: None Options: - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to x509. - Clusters of Kubernetes 1.13 or later support rbac and authenticating_proxy. The parameter defaults to rbac. - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to rbac. Default value: - Clusters of Kubernetes 1.11 or earlier: x509 - Clusters of Kubernetes 1.13 or later: rbac

Parameter	Type	Description
authenticatingProxy	AuthenticatingProxy object	Details: Configuration related to the authenticating_proxy mode Constraints: This field is mandatory when the authentication mode is authenticating_proxy .

Table 4-118 AuthenticatingProxy

Parameter	Type	Description
ca	String	Details: X509 CA certificate (Base64-encoded) configured in authenticating_proxy mode Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A
cert	String	Details: Client certificate issued by the X509 CA certificate configured in authenticating_proxy mode, which is used for authentication from kube-apiserver to the extended API server. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A

Parameter	Type	Description
privateKey	String	Details: Private key of the client certificate issued by the X509 CA certificate configured in <code>authenticating_proxy</code> mode, which is used for authentication from kube-apiserver to the extended API server. The private key used by the Kubernetes cluster does not support password encryption. Use an unencrypted private key. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy. Options: Maximum size: 1 MB Default value: N/A

Table 4-119 MasterSpec

Parameter	Type	Description
availabilityZone	String	Details: AZ where a control plane node is located. The AZ name must be specified. For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: N/A Default value: N/A

Table 4-120 ResourceTag

Parameter	Type	Description
key	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: <code>_.:=+@</code> • The value cannot start with <code>_sys_</code>. Default value: N/A
value	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: <code>_.:/=+@</code> Default value: N/A

Table 4-121 ClusterExtendParam

Parameter	Type	Description
clusterAZ	String	Details: AZs of master nodes in a cluster For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: • AZs supported by a specified region • multi_az: (Optional) multiple AZs, which can be configured only when a cluster with multiple master nodes is used • <i>AZs of the dedicated cloud computing pool:</i> Master nodes will be deployed in the DeC AZs. It is mandatory for dedicated CCE clusters. Default value: If no AZ is specified, an AZ is randomly allocated by default.

Parameter	Type	Description
dssMasterVolume s	String	Details: Whether the system and data disks of a master node use DSS storage pools. If this parameter is unspecified or left blank, EVS disks are used by default. Constraints: This parameter is mandatory for dedicated CCE clusters. It is in the following format: <pre><rootVol.dssPoolID>.<rootVol.volType>;<dataVol.dssPoolID>.<dataVol.volType></pre> Specifically: • rootVol specifies the system disk. dataVol specifies a data disk. • dssPoolID specifies the ID of a DSS storage pool. • volType specifies the storage volume type of the DSS storage pool, such as SAS, SSD, SATA, ESSD, GPSSD, ESSD2, and GPSSD2. Options: N/A Default value: N/A Example: c950ee97-587c-4f24-8a74-3367e3da570f.sas;6edbc2f4-1507-44f8-ac0d- eed1d2608d38.ssd NOTE This field cannot be configured for non-dedicated CCE clusters.
enterpriseProjec- tId	String	Details: ID of the enterprise project that a cluster belongs to Constraints: An enterprise project can be configured only after the enterprise project function is enabled. Options: N/A Default value: N/A

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: This parameter has been deprecated. If both this parameter and kubeProxyMode in ClusterSpec are specified, the latter is used. Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables
clusterExternalIP	String	Details: EIP of a cluster control plane node. After an EIP is bound to the control plane node, you can access the cluster management plane APIs through the EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ fixPoolMask	String	Details: Number of mask bits of the fixed IP address pool of the container networks. This parameter determines the maximum number of IP addresses that can be assigned to containers on a node. This, along with the maxPods setting during node creation, determines the maximum number of pods that can be created on a node. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: Only the vpc-router networks support this function. Options: Integers from 24 to 28 Default value: 24
decMasterFlavor	String	Details: Master node flavor in a dedicated CCE cluster Constraints: None Options: N/A Default value: N/A
dockerUmaskMode	String	Details: Default UmaskMode configuration of Docker in a cluster Constraints: None Options: • secure • normal Default value: normal

Parameter	Type	Description
kubernetes.io/ cpuManagerPolicy	String	Details: Cluster CPU management policy Constraints: None Options: • none or null: disables pods from exclusively occupying CPUs. Select this option if you want a large pool of shareable CPU cores. • static: enables pods to exclusively occupy CPUs. Select this option if your workload is sensitive to CPU cache and scheduling latency. In a CCE Turbo cluster, this setting is valid only for nodes where common containers, not secure containers, run. Default value: none
orderId	String	Details: Order ID Constraints: This parameter is returned in the response when the cluster is billed on a yearly/monthly basis with auto payment enabled (only in creation scenarios). Options: N/A Default value: N/A

Parameter	Type	Description
periodType	String	Details: Subscription term unit Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1, which is, the yearly/monthly billing mode is configured. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • month • year Default value: N/A
periodNum	Integer	Details: Number of subscription periods Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether to enable auto renewal Constraints: This field is valid only when billingMode is set to 1. Options: • true: Auto renewal is enabled. • false: Auto renewal is not enabled. Default value: false

Parameter	Type	Description
isAutoPay	String	Details: Whether to enable auto payment Constraints: This field is valid only when billingMode is set to 1 . Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: false
upgradefrom	String	Details: Records of how a cluster is upgraded to its current version Constraints: This field is only returned by a query API. Options: N/A Default value: N/A

Table 4-122 PackageConfiguration

Parameter	Type	Description
name	String	Details: Component name Constraints: None Options: N/A Default value: N/A
configurations	Array of ConfigurationItem objects	Details: Component configuration items Constraints: None

Table 4-123 ConfigurationItem

Parameter	Type	Description
name	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A
value	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Table 4-124 ClusterOps

Parameter	Type	Description
alarm	AlarmInfo object	Details: Alarm assistant settings. CCE is integrated with AOM, which enables alarm functionality, allowing for easy search and configuration of alarms. The Cloud Native Cluster Monitoring add-on sends metric rule data in CCE alarm center to AOM instances. Constraints: None

Table 4-125 AlarmInfo

Parameter	Type	Description
topics	Array of strings	Details: Contact group list. You can enter SMN topic names and configure contact groups to manage endpoints that have subscribed to alarm messages. Constraints: None
alarmRuleTemplated	String	Details: Alarm rule template ID transferred when the alarm assistant is enabled. By default, the alarm rule template in the container scenario is used. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
promInstanceID	String	Details: ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A
promEnterpriseProjectID	String	Details: Enterprise project ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A

Table 4-126 EncryptionConfig

Parameter	Type	Description
mode	String	Details: Encryption mode. Either encryption using the CCE-managed key or KMS key can be configured. Constraints: None Options: • Default: Encryption is performed using the CCE-managed key. • KMS: KMS encryption is used. Default value: Default
kmsKeyID	String	Details: KMS key ID • In the API for creating a cluster, if the mode field is set to Default, there is no need to specify this field. If the mode field is set to KMS, you can specify a value for this field. If this parameter is left blank, the default KMS key is used by default. If the default key does not exist, DEW will automatically create a default key (cce/default) for the user. To ensure proper cluster functionality, it is important to use a valid KMS key. Refrain from deleting or disabling the key before the cluster lifecycle is complete, as this may cause exceptions. Once a key has been configured for a cluster, it cannot be modified. • In the cluster query API, if the mode field is set to Default, the returned value of this field is empty. If the mode field is set to KMS, the specific key ID will be returned. Constraints: None Options: N/A Default value: N/A

Table 4-127 ClusterStatus

Parameter	Type	Description
phase	String	Details: Cluster status Constraints: None Options: • Available: The cluster is running properly. • Unavailable: The cluster is exhibiting unexpected behavior and needs to be manually deleted. • ScalingUp: Nodes are being added to the cluster. • ScalingDown: The cluster is being downsized to fewer nodes. • Creating: The cluster is being created. • Deleting: The cluster is being deleted. • Upgrading: The cluster is being upgraded. • Resizing: Cluster specifications are being changed. • ResizeFailed: Cluster specification changing is abnormal. • RollingBack: The cluster is being rolled back. • RollbackFailed: The cluster rollback is abnormal. • Hibernating: The cluster is being hibernated. • Hibernation: The cluster is in hibernation. • Freezing: The cluster is being frozen. • Frozen: The cluster has been frozen. • UnFreezing: The cluster is being unfrozen. • Awaking: The cluster is being woken up from hibernation.

Parameter	Type	Description
		• Empty: The cluster does not have any resources. This field has been deprecated. • Error: Resources in the cluster are abnormal. You can manually delete the cluster.
jobID	String	Details: ID of the job associated with the current cluster status. It supports: • ID of the associated job returned during cluster creation. You can use it to obtain the auxiliary job information for creating a cluster. • ID of the associated job returned when a cluster fails to be deleted or is being deleted. If this parameter is not empty, you can use the job ID to obtain the auxiliary job information for deleting a cluster. Constraints: None Options: N/A NOTE Job information is short-lived and is only used for tracking the progress of short-term jobs. Do not use it for cluster status query or other scenarios.
reason	String	Details: Why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A

Parameter	Type	Description
message	String	Details: Detailed information about why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A
endpoints	Array of ClusterEndpoints objects	Details: Access address of kube-apiserver in a cluster Constraints: None
isLocked	Boolean	Details: CBC resource locked Constraints: None Options: • true: The resource is locked by CBC. • false: The resource is not locked by CBC.
lockScene	String	Details: CBC resource locking scenario Constraints: None Options: N/A
lockSource	String	Details: Resource locked Constraints: None Options: N/A

Parameter	Type	Description
lockSourceId	String	Details: ID of a locked resource Constraints: None Options: N/A
deleteOption	Object	Details: Status of the configuration to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A
deleteStatus	Object	Details: Information of the status to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A

Table 4-128 ClusterEndpoints

Parameter	Type	Description
url	String	Details: Access address of kube-apiserver in a cluster Constraints: None Options: N/A

Parameter	Type	Description
type	String	Details: Type of the cluster access address Constraints: None Options: • Internal: address for internal network access • External: address for external network access

Example Requests

- Updating only the description of a cluster

```
{
  "spec" : {
    "description" : "new description"
  }
}
```

- Updating only the custom SAN of the cluster certificate

```
{
  "spec" : {
    "customSan" : [ "192.168.1.0", "example.com" ]
  }
}
```

- Updating both the cluster description and custom certificate SAN

```
{
  "spec" : {
    "description" : "new description",
    "customSan" : [ "192.168.1.0", "example.com" ]
  }
}
```

- Configure a CIDR block for a cluster that uses VPCs and is of a version later than v1.21.

```
{
  "spec" : {
    "containerNetwork" : {
      "cidrs" : [ {
        "cidr" : "10.10.0.0/16"
      }, {
        "cidr" : "10.11.0.0/16"
      } ]
    }
  }
}
```

- Modify the default security group of a node in a cluster.

/api/v3/projects/{project_id}/clusters/{cluster_id}

```
{
  "spec" : {
    "hostNetwork" : {
      "SecurityGroup" : "6ee29825-8f49-4796-b33a-fc76f84a59ae"
    }
  }
}
```

Example Responses

Status code: 200

Information about the specified cluster is successfully updated.

```
{
  "kind" : "Cluster",
  "apiVersion" : "v3",
  "metadata" : {
    "alias" : "mycluster",
    "name" : "mycluster",
    "uid" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
    "creationTimestamp" : "2018-08-02 03:48:58.968214406 +0000 UTC",
    "updateTimestamp" : "2018-08-02 06:39:36.844676088 +0000 UTC"
  },
  "spec" : {
    "type" : "VirtualMachine",
    "flavor" : "cce.s1.small",
    "version" : "v1.7.3-r13",
    "description" : "new description",
    "customSan" : [ "192.168.1.0", "example.com" ],
    "hostNetwork" : {
      "vpc" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
      "subnet" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
      "SecurityGroup" : "6ee29825-8f49-4796-b33a-fc76f84a59ae"
    },
    "containerNetwork" : {
      "mode" : "overlay_l2",
      "cidr" : "172.17.0.0/16"
    },
    "authentication" : {
      "mode" : "x509",
      "authenticatingProxy" : { }
    },
    "billingMode" : 0
  },
  "status" : {
    "phase" : "Available",
    "endpoints" : [ {
      "url" : "https://192.168.0.11:5443",
      "type" : "Internal"
    } ]
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Updating only the description of a cluster

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpdateClusterSolution {
```

```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");
    String projectId = "{project_id}";

    ICredential auth = new BasicCredentials()
        .withProjectId(projectId)
        .withAk(ak)
        .withSk(sk);

    CceClient client = CceClient.newBuilder()
        .withCredential(auth)
        .withRegion(CceRegion.valueOf("<YOUR REGION>"))
        .build();
    UpdateClusterRequest request = new UpdateClusterRequest();
    request.withClusterId("{cluster_id}");
    ClusterInformation body = new ClusterInformation();
    ClusterInformationSpec specbody = new ClusterInformationSpec();
    specbody.withDescription("new description");
    body.withSpec(specbody);
    request.withBody(body);
    try {
        UpdateClusterResponse response = client.updateCluster(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

- Updating only the custom SAN of the cluster certificate

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```

```
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();

UpdateClusterRequest request = new UpdateClusterRequest();
request.withClusterId("{cluster_id}");
ClusterInformation body = new ClusterInformation();
List<String> listSpecCustomSan = new ArrayList<>();
listSpecCustomSan.add("192.168.1.0");
listSpecCustomSan.add("example.com");
ClusterInformationSpec specbody = new ClusterInformationSpec();
specbody.withCustomSan(listSpecCustomSan);
body.withSpec(specbody);
request.withBody(body);
try {
    UpdateClusterResponse response = client.updateCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- Updating both the cluster description and custom certificate SAN

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);
```

```
CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateClusterRequest request = new UpdateClusterRequest();
request.withClusterId("{cluster_id}");
ClusterInformation body = new ClusterInformation();
List<String> listSpecCustomSan = new ArrayList<>();
listSpecCustomSan.add("192.168.1.0");
listSpecCustomSan.add("example.com");
ClusterInformationSpec specbody = new ClusterInformationSpec();
specbody.withDescription("new description")
    .withCustomSan(listSpecCustomSan);
body.withSpec(specbody);
request.withBody(body);
try {
    UpdateClusterResponse response = client.updateCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- Configure a CIDR block for a cluster that uses VPCs and is of a version later than v1.21.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
```

```
.withRegion(CceRegion.valueOf("<YOUR REGION>"))
.build();
UpdateClusterRequest request = new UpdateClusterRequest();
request.withClusterId("{cluster_id}");
ClusterInformation body = new ClusterInformation();
List<ContainerCIDR> listContainerNetworkCidrs = new ArrayList<>();
listContainerNetworkCidrs.add(
    new ContainerCIDR()
        .withCidr("10.10.0.0/16")
);
listContainerNetworkCidrs.add(
    new ContainerCIDR()
        .withCidr("10.11.0.0/16")
);
ContainerNetworkUpdate containerNetworkSpec = new ContainerNetworkUpdate();
containerNetworkSpec.withCidrs(listContainerNetworkCidrs);
ClusterInformationSpec specbody = new ClusterInformationSpec();
specbody.withContainerNetwork(containerNetworkSpec);
body.withSpec(specbody);
request.withBody(body);
try {
    UpdateClusterResponse response = client.updateCluster(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- Modify the default security group of a node in a cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpdateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
```

```
        .withRegion(CceRegion.valueOf("<YOUR REGION>"))
        .build();
    UpdateClusterRequest request = new UpdateClusterRequest();
    request.withClusterId("{cluster_id}");
    ClusterInformation body = new ClusterInformation();
    ClusterInformationSpecHostNetwork hostNetworkSpec = new
ClusterInformationSpecHostNetwork();
    hostNetworkSpec.withSecurityGroup("6ee29825-8f49-4796-b33a-fc76f84a59ae");
    ClusterInformationSpec specbody = new ClusterInformationSpec();
    specbody.withHostNetwork(hostNetworkSpec);
    body.withSpec(specbody);
    request.withBody(body);
    try {
        UpdateClusterResponse response = client.updateCluster(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

- Updating only the description of a cluster

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateClusterRequest()
        request.cluster_id = "{cluster_id}"
        specbody = ClusterInformationSpec(
            description="new description"
        )
        request.body = ClusterInformation(
            spec=specbody
        )
        response = client.update_cluster(request)
        print(response)
```



```
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Updating only the custom SAN of the cluster certificate

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateClusterRequest()
        request.cluster_id = "{cluster_id}"
        listCustomSanSpec = [
            "192.168.1.0",
            "example.com"
        ]
        specbody = ClusterInformationSpec(
            custom_san=listCustomSanSpec
        )
        request.body = ClusterInformation(
            spec=specbody
        )
        response = client.update_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- Updating both the cluster description and custom certificate SAN

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
```

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = UpdateClusterRequest()
    request.cluster_id = "{cluster_id}"
    listCustomSanSpec = [
        "192.168.1.0",
        "example.com"
    ]
    specbody = ClusterInformationSpec(
        description="new description",
        custom_san=listCustomSanSpec
    )
    request.body = ClusterInformation(
        spec=specbody
    )
    response = client.update_cluster(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Configure a CIDR block for a cluster that uses VPCs and is of a version later than v1.21.

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateClusterRequest()
        request.cluster_id = "{cluster_id}"
        listCidrsContainerNetwork = [
            ContainerCIDR(
                cidr="10.10.0.0/16"
            ),
            ContainerCIDR(
```

```
        cidr="10.11.0.0/16"
    )
]
containerNetworkSpec = ContainerNetworkUpdate(
    cidrs=listCidrsContainerNetwork
)
specbody = ClusterInformationSpec(
    container_network=containerNetworkSpec
)
request.body = ClusterInformation(
    spec=specbody
)
response = client.update_cluster(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Modify the default security group of a node in a cluster.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateClusterRequest()
        request.cluster_id = "{cluster_id}"
        hostNetworkSpec = ClusterInformationSpecHostNetwork(
            security_group="6ee29825-8f49-4796-b33a-fc76f84a59ae"
        )
        specbody = ClusterInformationSpec(
            host_network=hostNetworkSpec
        )
        request.body = ClusterInformation(
            spec=specbody
        )
        response = client.update_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- Updating only the description of a cluster

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateClusterRequest{
        request.ClusterId = "{cluster_id}"
        descriptionSpec := "new description"
        specbody := &model.ClusterInformationSpec{
            Description: &descriptionSpec,
        }
        request.Body = &model.ClusterInformation{
            Spec: specbody,
        }
        response, err := client.UpdateCluster(request)
        if err == nil {
            fmt.Printf("%v\n", response)
        } else {
            fmt.Println(err)
        }
    }
}
```

- Updating only the custom SAN of the cluster certificate

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
```

running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateClusterRequest{}
request.ClusterId = "{cluster_id}"
var listCustomSanSpec = []string{
    "192.168.1.0",
    "example.com",
}
specbody := &model.ClusterInformationSpec{
    CustomSan: &listCustomSanSpec,
}
request.Body = &model.ClusterInformation{
    Spec: specbody,
}
response, err := client.UpdateCluster(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Updating both the cluster description and custom certificate SAN

package main

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    environment
```

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
```

```
Build()

request := &model.UpdateClusterRequest{}
request.ClusterId = "{cluster_id}"
var listCustomSanSpec = []string{
    "192.168.1.0",
    "example.com",
}
descriptionSpec := "new description"
specbody := &model.ClusterInformationSpec{
    Description: &descriptionSpec,
    CustomSan: &listCustomSanSpec,
}
request.Body = &model.ClusterInformation{
    Spec: specbody,
}
response, err := client.UpdateCluster(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Configure a CIDR block for a cluster that uses VPCs and is of a version later than v1.21.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateClusterRequest{}
    request.ClusterId = "{cluster_id}"
    var listCidrsContainerNetwork = []model.ContainerCidr{
        {
            Cidr: "10.10.0.0/16",
        },
        {
            Cidr: "10.11.0.0/16",
        },
    }
}
```

```
containerNetworkSpec := &model.ContainerNetworkUpdate{
    Cidrs: &listCidrsContainerNetwork,
}
specbody := &model.ClusterInformationSpec{
    ContainerNetwork: containerNetworkSpec,
}
request.Body = &model.ClusterInformation{
    Spec: specbody,
}
response, err := client.UpdateCluster(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Modify the default security group of a node in a cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateClusterRequest{}
    request.ClusterId = "{cluster_id}"
    securityGroupId := "6ee29825-8f49-4796-b33a-fc76f84a59ae"
    hostNetworkSpec := &model.ClusterInformationSpecHostNetwork{
        SecurityGroup: &securityGroupHostNetwork,
    }
    specbody := &model.ClusterInformationSpec{
        HostNetwork: hostNetworkSpec,
    }
    request.Body = &model.ClusterInformation{
        Spec: specbody,
    }
    response, err := client.UpdateCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Information about the specified cluster is successfully updated.

Error Codes

See [Error Codes](#).

4.2.5 Deleting a Cluster

Function

This API is used to delete a specified cluster.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /api/v3/projects/{project_id}/clusters/{cluster_id}

Table 4-129 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-130 Query Parameters

Parameter	Mandatory	Type	Description
delete_efs	No	String	Details: Whether to delete an SFS Turbo volume Constraints: N/A Options: • true or block: The system starts to delete the object. If the deletion fails, subsequent processes will be blocked. • try: The system starts to delete the object. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • false or skip: The deletion is skipped. This is the default option. Default value: false

Parameter	Mandatory	Type	Description
delete_eni	No	String	Details: Whether to delete an ENI port (native ENI) Constraints: N/A Options: • true or block: The system starts to delete the object. If the deletion fails, subsequent processes will be blocked. This is the default option. • try: The system starts to delete the object. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • false or skip: The deletion is skipped. Default value: block
delete_evs	No	String	Details: Whether to delete an EVS disk Constraints: N/A Options: • true or block: The system starts to delete the object. If the deletion fails, subsequent processes will be blocked. • try: The system starts to delete the object. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • false or skip: The deletion is skipped. This is the default option. Default value: false

Parameter	Mandatory	Type	Description
delete_net	No	String	Details: Whether to delete cluster Service or ingress resources, such as a load balancer Constraints: N/A Options: • true or block: The system starts to delete the object. If the deletion fails, subsequent processes will be blocked. This is the default option. • try: The system starts to delete the object. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • false or skip: The deletion is skipped. Default value: block

Parameter	Mandatory	Type	Description
delete_obs	No	String	Details: Whether to delete an OBS volume Constraints: N/A Options: • true or block: The system starts to delete the object. If the deletion fails, subsequent processes will be blocked. • try: The system starts to delete the object. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • false or skip: The deletion is skipped. This is the default option. Default value: false
delete_sfs	No	String	Details: Whether to delete an SFS volume Constraints: N/A Options: • true or block: The system starts to delete the object. If the deletion fails, subsequent processes will be blocked. • try: The system starts to delete the object. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • false or skip: The deletion is skipped. This is the default option. Default value: false

Parameter	Mandatory	Type	Description
delete_sfs30	No	String	Details: Whether to delete an SFS 3.0 volume Constraints: N/A Options: • true or block: The system starts to delete the object. If the deletion fails, subsequent processes will be blocked. • try: The system starts to delete the object. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • false or skip: The deletion is skipped. This is the default option. Default value: false

Parameter	Mandatory	Type	Description
lts_reclaim_policy	No	String	Details: Whether to delete an LTS resource, such as a log group or a log stream Constraints: N/A Options: • Delete_Log_Group: The system starts to delete a log group. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. • Delete_Master_Log_Stream: The system starts to delete a master log stream. If the deletion fails, no deletion will be retried, and subsequent processes will proceed. This is the default option. • Retain: The deletion is skipped. Default value: Delete_Master_Log_Stream

Parameter	Mandatory	Type	Description
tobedeleted	No	String	Details: Whether to use the preset deletion mode for yearly/monthly clusters. This parameter is valid only for yearly/monthly clusters. If this parameter is used, the cluster does not delete resources. All query parameters of this request will be preset in the cluster database for identifying the resources to be deleted when a yearly/monthly cluster is unsubscribed from. This request can be executed for multiple times. Each request overwrites the deletion parameters preset last time. Constraints: This parameter must be used together with other deletion parameters. You can specify parameter values or the system uses the default ones. Options: • true: specifies the preset mode. Only the query parameters are preset and the deletion is not performed. Default value: N/A

Parameter	Mandatory	Type	Description
ondemand_node_policy	No	String	Details: Policy for handling all pay-per-use nodes in a cluster Constraints: N/A Options: • delete: The servers will be deleted. • reset: The servers will undergo a reset, which will result in the retention of the servers themselves, but not the data stored on them. • retain: The servers will be kept without undergoing a reset, and the data stored on them will also be retained. Default value: NOTE If this parameter is not specified, pay-per-use nodes will be deleted by default and managed nodes will be retained.
periodic_node_policy	No	String	Details: Policy for handling all yearly/monthly nodes in a cluster Constraints: N/A Options: • reset: The servers will undergo a reset, which will result in the retention of the servers themselves, but not the data stored on them. • retain: The servers will be kept without undergoing a reset, and the data stored on them will also be retained. Default value: retain

Request Parameters

Table 4-131 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-132 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: • Cluster • cluster Default value: N/A
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: • v3 Default value: N/A
metadata	ClusterMetadata object	Details: Basic information about a cluster. Metadata is a collection of attributes. Constraints: None
spec	ClusterSpec object	Details: Detailed description about a cluster. CCE creates or updates objects by defining or updating spec . Constraints: None
status	ClusterStatus object	Details: Cluster status and ID of a cluster creation job Constraints: None

Table 4-133 ClusterMetadata

Parameter	Type	Description
name	String	Details: Cluster name Constraints: None Options: Enter 4 to 128 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: Cluster ID, which identifies a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. When you create a yearly/monthly cluster, no cluster ID will be returned in the response body. Options: N/A Default value: N/A

Parameter	Type	Description
alias	String	Details: Alias of a cluster name displayed on the CCE console. The name can be changed. A cluster alias must be unique. Constraints: In the request body for creating or updating a cluster, if the cluster alias is not specified or set to null, the cluster name will be used as the cluster alias. In the response body for creating a cluster or other response bodies, if the cluster alias is not configured, no value will be returned. Options: Enter 4 to 128 characters starting with a letter and not starting or ending with a hyphen (-). Only digits, letters, and hyphens (-) are allowed. Default value: N/A

Parameter	Type	Description
annotations	Map<String,String>	Details: Cluster annotations in the format of key-value pairs <pre>"annotations": { "key1" : "value1", "key2" : "value2" }</pre> Constraints: This field is not stored in the database and is only used to specify the add-ons to be installed in the cluster. Options: N/A Default value: N/A NOTE - Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. - You can install ICAgent during cluster creation by adding the key-value pair "cluster.install.addons.external/install": "[{\addonTemplateName \":\"icagent\"}]".
labels	Map<String,String>	Details: Cluster labels, in the format of key-value pairs Constraints: The value of this field is automatically generated by the system and is used by the frontend to identify the features supported by the cluster during the upgrade. Custom values are invalid. Options: N/A Default value: N/A

Parameter	Type	Description
creationTimestamp	String	Details: Time when a cluster was created Constraints: The value is automatically recorded during cluster creation and cannot be specified. Options: N/A Default value: N/A
updateTimestamp	String	Details: Time when a cluster was updated Constraints: The value is automatically recorded during cluster updates and cannot be specified. Options: N/A Default value: N/A
timezone	String	Details: Time zone of a cluster. The time zone must be included in IANA Time Zone Database , for example, Asia/Shanghai . Constraints: None Options: N/A Default value: N/A

Table 4-134 ClusterSpec

Parameter	Type	Description
category	String	Details: Cluster type Constraints: None Options: • CCE: CCE cluster. • Turbo: CCE Turbo cluster. CCE clusters support hybrid deployment of VMs and BMSs, and heterogeneous nodes such as GPU and NPU nodes, allowing you to run your containers in a secure, stable container runtime environment based on a high-performance network model. CCE Turbo clusters run on the Cloud Native 2.0 infrastructure that features hardware and software synergy to support passthrough networking, high security and reliability, and intelligent scheduling. Default value: CCE when the container network is not set to eni Turbo when the container network is set to eni
type	String	Definition Control plane node architecture Constraints N/A Range • VirtualMachine: The control plane node is an x86 server. • ARM64: The control plane node is an Arm-based Kunpeng server. Default Value VirtualMachine or ARM64 when there are not enough VMs

Parameter	Type	Description
flavor	String	Details: Cluster specifications. Specifications of clusters of v1.15 or later can be changed after they are created. For details, see Modifying Cluster Specifications. Configure this parameter based on actual service requirements. Constraints: None Options: • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s1.large: a large-scale CCE cluster with one control plane node and a maximum of 1,000 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes Special cluster specifications in DeC are as follows: - cce.dec.s1.small: a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes) - cce.dec.s1.medium: a medium-scale DeC CCE cluster with one control

Parameter	Type	Description
		plane node and a maximum of 200 worker nodes) - cce.dec.s1.large: a large-scale DeC CCE cluster with one control plane node and a maximum of 1,000 worker nodes) - cce.dec.s1.xlarge: an ultra-large-scale DeC CCE cluster with one control plane node and a maximum of 2,000 worker nodes) - cce.dec.s2.small: a small-scale DeC CCE cluster with three control plane nodes and a maximum of 50 worker nodes) - cce.dec.s2.medium: a medium-scale DeC CCE cluster with three control plane nodes and a maximum of 200 worker nodes) - cce.dec.s2.large: a large-scale DeC CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes) - cce.dec.s2.xlarge: an ultra-large-scale DeC CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes) Default value: N/A

Parameter	Type	Description
		NOTE The fields in the parameters are described as follows: • s1: specifies a cluster with one control plane node. If the control plane node is faulty, the cluster will become unavailable, but running workloads in the cluster are not affected. • s2: specifies an HA cluster with three control plane nodes. If one of the control plane nodes is faulty, the cluster is still available. • dec: specifies a DeC CCE cluster. For example, cce.dec.s1.small specifies a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes. • small: specifies that a cluster can manage a maximum of 50 worker nodes. • medium: specifies that a cluster can manage a maximum of 200 worker nodes. • large: specifies that a cluster can manage a maximum of 1,000 worker nodes. • xlarge: specifies that a cluster can manage a maximum of 2,000 worker nodes.

Parameter	Type	Description
version	String	Details: Cluster version, which mirrors the baseline version of the Kubernetes community. The latest commercial version is recommended. NOTE For details about the Kubernetes version policy, see Kubernetes Version Policy. • CCE Turbo clusters of v1.19 and later versions are commercially available. You can create clusters of the latest three versions on the CCE console. To learn which cluster versions are available, log in to the CCE console, create a cluster, and check the supported cluster versions. You can call APIs to create clusters of other versions. However, these cluster versions will be gradually terminated. For details about the support policy, see the CCE announcement. Constraints: The value must be in the format of vX.Y[Z[-rN]], for example, v1.30, v1.30.0, or v1.30.0-r0. Specifying either of these values will create a cluster of version 1.30. • X: major version in the Kubernetes community • Y: minor version in the Kubernetes community • Z: patch version in the Kubernetes community • N: CCE patch version. For details about CCE patch versions, see Patch Versions. Options: N/A Default value: • If this parameter is left empty, a cluster of the latest version is created by default. • If a baseline cluster version is specified but the R version is not specified, a cluster of the latest R version will be created by default. It

Parameter	Type	Description
		is a good practice not to specify the R version.
platformVersion	String	Details: CCE cluster platform version, which is an internal version under the cluster version (version). Platform versions are used to trace iterations in a major cluster version. They are unique within a major cluster version and recounted when the major cluster version changes. Constraints: This parameter cannot be customized, and when you create a cluster, the latest corresponding platform version is selected automatically. Options: The format of platformVersion is cce.X.Y. • X: internal feature version, which indicates changes in features, patches, or OS support in the cluster version. The value starts from 1 and increases monotonically. • Y: patch version of an internal feature version. It is used only for software package update after the feature version is released. No other modification is involved. The value starts from 0 and increases monotonically. Default value: N/A

Parameter	Type	Description
legacyVersion	String	Details: (Deprecated) Outdated version of CCE clusters, which serves no practical purpose and is only used to showcase the cluster version and platform version combination. The version number is globally unique. For example, if the cluster version is va.b and the platformVersion is cce.X.Y, the value of legacyVersion is va.b.X-rY. Constraints: None Options: N/A Default value: N/A
description	String	Details: Cluster description, for example, which purpose the cluster is intended to serve. By default, this parameter is left unspecified. To modify cluster description after a cluster is created, call the API for updating a specified cluster or go to the cluster details page on the CCE console. Constraints: Only UTF-8 encoding is supported. The length must be less than or equal to 200 bytes. Options: N/A Default value: N/A

Parameter	Type	Description
customSan	Array of strings	Details: Custom Subject Alternative Name (SAN) in the server certificate of a cluster API server, which must comply with the SSL and X509 standard formats Constraints: Duplicate names are not allowed. Options: Values in the IP address or domain name format Default value: N/A Example: SAN 1: DNS Name=example.com SAN 2: DNS Name=www.example.com SAN 3: DNS Name=example.net SAN 4: IP Address=93.184.216.34
ipv6enable	Boolean	Details: Whether a cluster supports IPv6 addresses. It is supported by clusters v1.15 and later. Constraints: After IPv6 is enabled, iptables is not supported. The VPC network model does not support IPv4/IPv6 dual-stack. Options: • true: IPv4/IPv6 dual-stack is enabled. • false: Only IPv4 is used. Default value: false
hostNetwork	HostNetwork object	Details: Node network parameters, including VPC and subnet ID. This field is mandatory because nodes in a cluster communicate with each other using a VPC. Constraints: None

Parameter	Type	Description
containerNetwork	ContainerNetwork object	Details: Container network parameters, including the container network model and container CIDR block Constraints: None
eniNetwork	EniNetwork object	Details: Configuration of the Cloud Native 2.0 network model. You need to specify this field when creating a CCE Turbo cluster. Constraints: None
serviceNetwork	ServiceNetwork object	Details: Service CIDR block, including IPv4 CIDR block Constraints: None
publicAccess	PublicAccess object	Details: Cluster API access control Constraints: None
authentication	Authentication object	Details: Configurations of the cluster authentication mode Constraints: None
billingMode	Integer	Details: Cluster billing mode Constraints: None Options: • 0: pay-per-use • 1: yearly/monthly Default value: 0

Parameter	Type	Description
masters	Array of MasterSpec objects	Details: Advanced settings of the control plane nodes in a cluster. You can specify AZs for them. Constraints: If this parameter is not specified, no value is returned.
kubernetesSvcIpRange	String	Details: Service CIDR blocks which kubernetes clusterIP must fall within. This parameter is available only for clusters v1.11.7 and later. If this parameter is not specified during cluster creation, the default value 10.247.0.0/16 is used. This parameter is being deprecated, so you can use serviceNetwork instead. The new field contains the IPv4 CIDR blocks. Constraints: None Options: N/A Default value: N/A
clusterTags	Array of ResourceTag objects	Details: Cluster resource tags Constraints: None

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: None Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables is used by default.
az	String	Details: AZ (being deprecated). For details about AZs supported by CCE, see Regions and Endpoints . Constraints: This field is only returned by a query API.
extendParam	ClusterExtendParam object	Details: Extended field to decide whether a cluster will span across AZs or belong to a specified enterprise project, or whether a dedicated CCE cluster is to be created Constraints: None

Parameter	Type	Description
supportIstio	Boolean	Details: Whether Istio is supported Constraints: None Options: • true: Istio is supported. • false: Istio is not supported. Default value: true
enableDistMgt	Boolean	Details: Whether to enable support for remote clouds Constraints: Only CCE Turbo clusters support this function. Options: • true: Support for remote clouds is enabled. • false: Support for remote clouds is not enabled. Default value: false
deletionProtection	Boolean	Details: Cluster deletion protection, which prevents the deletion of a cluster when this function is enabled Constraints: None Options: • true: Cluster deletion protection is enabled. • false: Cluster deletion protection is disabled. Default value: false

Parameter	Type	Description
configurationsO- verride	Array of PackageConfigur ation objects	Details: Overrides the default component configurations in a cluster. For details about the supported components and their parameters, see Modifying Cluster Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored.
clusterOps	ClusterOps object	Details: Cluster O&M settings Constraints: None
encryptionConfig	EncryptionConfig object	Details: Configuration of secret encryption. Only one encryption mode can be configured. By default, the CCE-managed key is used for encryption. (Users are not aware of the key being used.) Constraints: None Options: N/A Default value: N/A

Table 4-135 HostNetwork

Parameter	Type	Description
vpc	String	Details: ID of the VPC where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and view the VPC ID in the VPC details page. - Method 2: Use the VPC APIs. For details, see Querying VPCs.
subnet	String	Details: Network ID of the subnet where nodes will be running Constraints: None Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the network ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
SecurityGroup	String	Details: ID of the default node security group in a cluster Constraints: None Options: N/A Default value: • If this parameter is not specified, the system automatically creates a default node security group. • If this parameter is specified, the cluster will be bound to the specified security group. NOTE To ensure proper communication, allow traffic from certain ports to the specified node security group. For details, see How Can I Configure a Security Group Rule in a Cluster?
controlPlaneSecurityGroup	String	Details: ID of the master node security group in a cluster Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Table 4-136 ContainerNetwork

Parameter	Type	Description
mode	String	Details: Container network type Constraints: Only one container network type can be selected. Options: • overlay_l2: an overlay_l2 network (container tunnel network) built for containers by using OpenVSwitch (OVS). • vpc-router: an underlay_l2 network built for containers by using IPvlan and custom VPC routes. • eni*: a Cloud Native 2.0 network. This model has integrated cloud native elastic network interfaces (ENIs), uses VPC CIDR blocks to allocate container IP addresses, and supports passthrough networking. You can use this model when creating a CCE Turbo cluster. Default value: N/A

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. If the selected CIDR block conflicts with existing ones, an error will be reported. Constraints: This parameter cannot be modified after the cluster is created. (It has been deprecated. If cidrs has been configured, the specified CIDR blocks will be ignored.) Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster. Options: Values in IPv4 CIDR format Default value: If this parameter is left blank, a unique CIDR block is randomly allocated from 172.(16–31).0.0/16 and 10.(0 16 32 48 64 80 96 112).0.0/12.
cidrs	Array of ContainerCIDR objects	Details: List of container CIDR blocks. In clusters v1.21 or later, the cidrs field is used. When the cluster network type is vpc-router, you can configure a maximum of 20 container CIDR blocks. In clusters earlier than v1.21, if the cidrs field is used, the first CIDR element in the array is used as the container CIDR block. Constraints: Once you have created a cluster that uses a container tunnel network, you cannot modify the CIDR blocks. Once you have created a cluster that uses a VPC network, you can add new CIDR blocks, but you are unable to modify the existing ones. If you need to adjust the CIDR blocks, you will have to create a cluster.

Table 4-137 ContainerCIDR

Parameter	Type	Description
cidr	String	Details: Container CIDR block. 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19 is recommended. Constraints: If a CIDR block conflict occurs, an error will be reported. Options: N/A Default value: N/A

Table 4-138 EniNetwork

Parameter	Type	Description
eniSubnetId	String	Details: IPv4 subnet ID of the subnet where an elastic network interface resides Constraints: IPv6 is not supported. This field will be deprecated. Use subnets instead. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Parameter	Type	Description
eniSubnetCIDR	String	Details: ENI subnet CIDR block Constraints: This parameter is being deprecated. The new field subnets is recommended. Options: N/A Default value: N/A
subnets	Array of NetworkSubnet objects	Details: List of IPv4 subnet IDs Constraints: None

Table 4-139 NetworkSubnet

Parameter	Type	Description
subnetID	String	Details: ID of the IPv4 subnet where the container network interfaces are in Constraints: IPv6 is not supported. Options: N/A Default value: N/A To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets.

Table 4-140 ServiceNetwork

Parameter	Type	Description
IPv4CIDR	String	Details: Value range of the Kubernetes clusterIP IPv4 CIDR block Constraints: None Options: N/A Default value: 10.247.0.0/16
IPv6CIDR	String	Details: Value range of the Kubernetes clusterIP IPv6 CIDR block Constraints: IPv6 Service CIDR blocks can be configured only for CCE Turbo clusters with IPv6 dual stack enabled. Options: N/A Default value: Default value for a CCE Turbo cluster: fc00::/112 Default value for a CCE standard cluster: fd00:1234::/120

Table 4-141 PublicAccess

Parameter	Type	Description
cidrs	Array of strings	Details: Trustlist of network CIDRs that are allowed to access cluster APIs. You are advised to allow the traffic from VPC and container network CIDRs. Constraints: This parameter is valid only when a cluster is created. Options: N/A Default value: By default, no trustlist is configured, and the value is 0.0.0.0/0 .

Table 4-142 Authentication

Parameter	Type	Description
mode	String	Details: Cluster authentication mode Constraints: None Options: - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to x509. - Clusters of Kubernetes 1.13 or later support rbac and authenticating_proxy. The parameter defaults to rbac. - Clusters of Kubernetes 1.11 or earlier support x509, rbac, and authenticating_proxy. The parameter defaults to rbac. Default value: - Clusters of Kubernetes 1.11 or earlier: x509 - Clusters of Kubernetes 1.13 or later: rbac

Parameter	Type	Description
authenticatingProxy	AuthenticatingProxy object	Details: Configuration related to the authenticating_proxy mode Constraints: This field is mandatory when the authentication mode is authenticating_proxy .

Table 4-143 AuthenticatingProxy

Parameter	Type	Description
ca	String	Details: X509 CA certificate (Base64-encoded) configured in authenticating_proxy mode Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A
cert	String	Details: Client certificate issued by the X509 CA certificate configured in authenticating_proxy mode, which is used for authentication from kube-apiserver to the extended API server. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy . Options: Maximum size: 1 MB Default value: N/A

Parameter	Type	Description
privateKey	String	Details: Private key of the client certificate issued by the X509 CA certificate configured in <code>authenticating_proxy</code> mode, which is used for authentication from kube-apiserver to the extended API server. The private key used by the Kubernetes cluster does not support password encryption. Use an unencrypted private key. (The value must be Base64-encoded.) Constraints: This field is mandatory when the cluster authentication mode is authenticating_proxy. Options: Maximum size: 1 MB Default value: N/A

Table 4-144 MasterSpec

Parameter	Type	Description
availabilityZone	String	Details: AZ where a control plane node is located. The AZ name must be specified. For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: N/A Default value: N/A

Table 4-145 ResourceTag

Parameter	Type	Description
key	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: <code>_.:=+@</code> • The value cannot start with <code>_sys_</code>. Default value: N/A
value	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: <code>_.:/=+@</code> Default value: N/A

Table 4-146 ClusterExtendParam

Parameter	Type	Description
clusterAZ	String	Details: AZs of master nodes in a cluster For details about AZs supported by CCE, see Regions and Endpoints. Constraints: None Options: • AZs supported by a specified region • multi_az: (Optional) multiple AZs, which can be configured only when a cluster with multiple master nodes is used • <i>AZs of the dedicated cloud computing pool:</i> Master nodes will be deployed in the DeC AZs. It is mandatory for dedicated CCE clusters. Default value: If no AZ is specified, an AZ is randomly allocated by default.

Parameter	Type	Description
dssMasterVolume s	String	Details: Whether the system and data disks of a master node use DSS storage pools. If this parameter is unspecified or left blank, EVS disks are used by default. Constraints: This parameter is mandatory for dedicated CCE clusters. It is in the following format: <pre><rootVol.dssPoolID>.<rootVol.volType>;<dataVol.dssPoolID>.<dataVol.volType></pre> Specifically: • rootVol specifies the system disk. dataVol specifies a data disk. • dssPoolID specifies the ID of a DSS storage pool. • volType specifies the storage volume type of the DSS storage pool, such as SAS, SSD, SATA, ESSD, GPSSD, ESSD2, and GPSSD2. Options: N/A Default value: N/A Example: c950ee97-587c-4f24-8a74-3367e3da570f.sas;6edbc2f4-1507-44f8-ac0d- eed1d2608d38.ssd NOTE This field cannot be configured for non-dedicated CCE clusters.
enterpriseProjec- tId	String	Details: ID of the enterprise project that a cluster belongs to Constraints: An enterprise project can be configured only after the enterprise project function is enabled. Options: N/A Default value: N/A

Parameter	Type	Description
kubeProxyMode	String	Details: Service forwarding mode Constraints: This parameter has been deprecated. If both this parameter and kubeProxyMode in ClusterSpec are specified, the latter is used. Options: • iptables: traditional kube-proxy mode, which uses iptables rules to implement Service load balancing. In this mode, too many iptables rules will be generated when many Services are deployed. Additionally, non-incremental updates will cause latency and obvious performance issues in the case of heavy service traffic. • ipvs: optimized kube-proxy mode with higher throughput and faster speed. This mode supports incremental updates and can keep connections uninterrupted during Service updates. It is suitable for large-sized clusters. Default value: iptables
clusterExternalIP	String	Details: EIP of a cluster control plane node. After an EIP is bound to the control plane node, you can access the cluster management plane APIs through the EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ fixPoolMask	String	Details: Number of mask bits of the fixed IP address pool of the container networks. This parameter determines the maximum number of IP addresses that can be assigned to containers on a node. This, along with the maxPods setting during node creation, determines the maximum number of pods that can be created on a node. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: Only the vpc-router networks support this function. Options: Integers from 24 to 28 Default value: 24
decMasterFlavor	String	Details: Master node flavor in a dedicated CCE cluster Constraints: None Options: N/A Default value: N/A
dockerUmaskMode	String	Details: Default UmaskMode configuration of Docker in a cluster Constraints: None Options: • secure • normal Default value: normal

Parameter	Type	Description
kubernetes.io/ cpuManagerPolicy	String	Details: Cluster CPU management policy Constraints: None Options: • none or null: disables pods from exclusively occupying CPUs. Select this option if you want a large pool of shareable CPU cores. • static: enables pods to exclusively occupy CPUs. Select this option if your workload is sensitive to CPU cache and scheduling latency. In a CCE Turbo cluster, this setting is valid only for nodes where common containers, not secure containers, run. Default value: none
orderId	String	Details: Order ID Constraints: This parameter is returned in the response when the cluster is billed on a yearly/monthly basis with auto payment enabled (only in creation scenarios). Options: N/A Default value: N/A

Parameter	Type	Description
periodType	String	Details: Subscription term unit Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1, which is, the yearly/monthly billing mode is configured. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • month • year Default value: N/A
periodNum	Integer	Details: Number of subscription periods Constraints: As a request parameter, it is valid and mandatory when billingMode is set to 1. As a response parameter, it is returned only when a yearly/monthly cluster is created. Options: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether to enable auto renewal Constraints: This field is valid only when billingMode is set to 1. Options: • true: Auto renewal is enabled. • false: Auto renewal is not enabled. Default value: false

Parameter	Type	Description
isAutoPay	String	Details: Whether to enable auto payment Constraints: This field is valid only when billingMode is set to 1 . Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: false
upgradefrom	String	Details: Records of how a cluster is upgraded to its current version Constraints: This field is only returned by a query API. Options: N/A Default value: N/A

Table 4-147 PackageConfiguration

Parameter	Type	Description
name	String	Details: Component name Constraints: None Options: N/A Default value: N/A
configurations	Array of ConfigurationItem objects	Details: Component configuration items Constraints: None

Table 4-148 ConfigurationItem

Parameter	Type	Description
name	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A
value	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Table 4-149 ClusterOps

Parameter	Type	Description
alarm	AlarmInfo object	Details: Alarm assistant settings. CCE is integrated with AOM, which enables alarm functionality, allowing for easy search and configuration of alarms. The Cloud Native Cluster Monitoring add-on sends metric rule data in CCE alarm center to AOM instances. Constraints: None

Table 4-150 AlarmInfo

Parameter	Type	Description
topics	Array of strings	Details: Contact group list. You can enter SMN topic names and configure contact groups to manage endpoints that have subscribed to alarm messages. Constraints: None
alarmRuleTemplated	String	Details: Alarm rule template ID transferred when the alarm assistant is enabled. By default, the alarm rule template in the container scenario is used. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
promInstanceID	String	Details: ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A
promEnterpriseProjectID	String	Details: Enterprise project ID of the AOM Prometheus instance transferred when the alarm assistant is enabled. If the Prometheus add-on is not installed or the AOM instance is not connected, there is no need to specify this parameter. In this case, the alarm center will not generate metric alarm rules. Constraints: None Options: N/A Default value: N/A

Table 4-151 EncryptionConfig

Parameter	Type	Description
mode	String	Details: Encryption mode. Either encryption using the CCE-managed key or KMS key can be configured. Constraints: None Options: • Default: Encryption is performed using the CCE-managed key. • KMS: KMS encryption is used. Default value: Default
kmsKeyID	String	Details: KMS key ID • In the API for creating a cluster, if the mode field is set to Default, there is no need to specify this field. If the mode field is set to KMS, you can specify a value for this field. If this parameter is left blank, the default KMS key is used by default. If the default key does not exist, DEW will automatically create a default key (cce/default) for the user. To ensure proper cluster functionality, it is important to use a valid KMS key. Refrain from deleting or disabling the key before the cluster lifecycle is complete, as this may cause exceptions. Once a key has been configured for a cluster, it cannot be modified. • In the cluster query API, if the mode field is set to Default, the returned value of this field is empty. If the mode field is set to KMS, the specific key ID will be returned. Constraints: None Options: N/A Default value: N/A

Table 4-152 ClusterStatus

Parameter	Type	Description
phase	String	Details: Cluster status Constraints: None Options: • Available: The cluster is running properly. • Unavailable: The cluster is exhibiting unexpected behavior and needs to be manually deleted. • ScalingUp: Nodes are being added to the cluster. • ScalingDown: The cluster is being downsized to fewer nodes. • Creating: The cluster is being created. • Deleting: The cluster is being deleted. • Upgrading: The cluster is being upgraded. • Resizing: Cluster specifications are being changed. • ResizeFailed: Cluster specification changing is abnormal. • RollingBack: The cluster is being rolled back. • RollbackFailed: The cluster rollback is abnormal. • Hibernating: The cluster is being hibernated. • Hibernation: The cluster is in hibernation. • Freezing: The cluster is being frozen. • Frozen: The cluster has been frozen. • UnFreezing: The cluster is being unfrozen. • Awaking: The cluster is being woken up from hibernation.

Parameter	Type	Description
		• Empty: The cluster does not have any resources. This field has been deprecated. • Error: Resources in the cluster are abnormal. You can manually delete the cluster.
jobID	String	Details: ID of the job associated with the current cluster status. It supports: • ID of the associated job returned during cluster creation. You can use it to obtain the auxiliary job information for creating a cluster. • ID of the associated job returned when a cluster fails to be deleted or is being deleted. If this parameter is not empty, you can use the job ID to obtain the auxiliary job information for deleting a cluster. Constraints: None Options: N/A NOTE Job information is short-lived and is only used for tracking the progress of short-term jobs. Do not use it for cluster status query or other scenarios.
reason	String	Details: Why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A

Parameter	Type	Description
message	String	Details: Detailed information about why a cluster changes to the current state. This parameter is returned if the cluster is not in the Available state. Constraints: None Options: N/A
endpoints	Array of ClusterEndpoints objects	Details: Access address of kube-apiserver in a cluster Constraints: None
isLocked	Boolean	Details: CBC resource locked Constraints: None Options: • true: The resource is locked by CBC. • false: The resource is not locked by CBC.
lockScene	String	Details: CBC resource locking scenario Constraints: None Options: N/A
lockSource	String	Details: Resource locked Constraints: None Options: N/A

Parameter	Type	Description
lockSourceId	String	Details: ID of a locked resource Constraints: None Options: N/A
deleteOption	Object	Details: Status of the configuration to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A
deleteStatus	Object	Details: Information of the status to be deleted. This parameter is contained only in the response to the deletion request. Constraints: None Options: N/A

Table 4-153 ClusterEndpoints

Parameter	Type	Description
url	String	Details: Access address of kube-apiserver in a cluster Constraints: None Options: N/A

Parameter	Type	Description
type	String	Details: Type of the cluster access address Constraints: None Options: • Internal: address for internal network access • External: address for external network access

Example Requests

None

Example Responses

Status code: 200

The job for deleting a cluster is successfully delivered.

```
{
  "kind" : "Cluster",
  "apiVersion" : "v3",
  "metadata" : {
    "alias" : "mycluster",
    "name" : "mycluster",
    "uid" : "fc563b3c-9552-11e8-8beb-0255ac106311",
    "creationTimestamp" : "2018-08-01 06:20:28.81667161 +0000 UTC",
    "updateTimestamp" : "2018-08-01 09:23:38.944333282 +0000 UTC"
  },
  "spec" : {
    "type" : "VirtualMachine",
    "flavor" : "cce.s1.small",
    "version" : "v1.7.3-r13",
    "description" : "new description",
    "hostNetwork" : {
      "vpc" : "cbcd56e8-03e7-4304-a477-b54bef0857c3",
      "subnet" : "5de50062-2be2-4a52-893e-e0906e3e9c9d"
    },
    "containerNetwork" : {
      "mode" : "overlay_l2",
      "cidr" : "172.16.0.0/16"
    },
    "authentication" : {
      "mode" : "x509",
      "authenticatingProxy" : { }
    },
    "billingMode" : 0
  },
  "status" : {
    "phase" : "Available",
    "jobID" : "e8ebf96c-956d-11e8-a949-0255ac10575d",
    "endpoints" : [ {
      "url" : "https://192.168.0.16:5443",
      "type" : "Internal"
    } ]
  }
}
```



```
}  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
public class DeleteClusterSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        CceClient client = CceClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))  
            .build();  
        DeleteClusterRequest request = new DeleteClusterRequest();  
        request.withClusterId("{cluster_id}");  
        try {  
            DeleteClusterResponse response = client.deleteCluster(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials
```

```
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteClusterRequest()
        request.cluster_id = "{cluster_id}"
        response = client.delete_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteClusterRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.DeleteCluster(request)
```

```
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for deleting a cluster is successfully delivered.

Error Codes

See [Error Codes](#).

4.2.6 Hibernating a Cluster

Function

This API is used to hibernate a running cluster. After a cluster is hibernated, master node resources stop being billed.

Constraints

1. After a cluster is hibernated, resources such as workloads cannot be created or managed in the cluster.
2. After a pay-per-use cluster is hibernated, master node resources are not billed. Other resources such as the nodes to which the cluster belongs, bound EIP, and bandwidth are billed based on their billing mode (yearly/monthly or pay-per-use).

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/hibernate

Table 4-154 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-155 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

The cluster hibernation job is successfully delivered. Keep querying the cluster status. When the cluster status changes to **Hibernation**, the cluster is hibernated.

None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class HibernateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        HibernateClusterRequest request = new HibernateClusterRequest();
        request.withClusterId("{cluster_id}");
        try {
            HibernateClusterResponse response = client.hibernateCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = HibernateClusterRequest()
        request.cluster_id = "{cluster_id}"
        response = client.hibernate_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
```

```
        WithCredential(auth).  
        Build())  
  
    request := &model.HibernateClusterRequest{}  
    request.ClusterId = "{cluster_id}"  
    response, err := client.HibernateCluster(request)  
    if err == nil {  
        fmt.Printf("%v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The cluster hibernation job is successfully delivered. Keep querying the cluster status. When the cluster status changes to Hibernation , the cluster is hibernated.

Error Codes

See [Error Codes](#).

4.2.7 Waking Up a Cluster

Function

This API is used to wake up a hibernated cluster. After the cluster is woken up, master node resources continue to be billed.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/awake

Table 4-156 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-157 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

The cluster wakeup job is successfully delivered. Keep querying the cluster status. When the cluster status changes to **Available**, the cluster is woken up successfully.

None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class AwakeClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        AwakeClusterRequest request = new AwakeClusterRequest();
        request.withClusterId("{cluster_id}");
        try {
            AwakeClusterResponse response = client.awakeCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AwakeClusterRequest()
        request.cluster_id = "{cluster_id}"
        response = client.awake_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
```

```
WithCredential(auth).
Build())

request := &model.AwakeClusterRequest{}
request.ClusterId = "{cluster_id}"
response, err := client.AwakeCluster(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The cluster wakeup job is successfully delivered. Keep querying the cluster status. When the cluster status changes to Available , the cluster is woken up successfully.

Error Codes

See [Error Codes](#).

4.2.8 Obtaining a Cluster Certificate

Function

This API is used to obtain a certificate of a specified cluster.

Constraints

This API is applicable to clusters of v1.13 and later.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/clustercert

Table 4-158 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-159 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-160 Request body parameters

Parameter	Mandatory	Type	Description
duration	Yes	Integer	Definition How long a cluster certificate is valid Constraints N/A Range -1 or 1 to 1827 NOTE - The minimum value is 1 day and the maximum value is 1827 days (5 years). For example, if there is one leap year within the five-year period, the maximum value is 1826 days. - If the value is set -1, the maximum value (5 years) will be used. Default Value N/A

Response Parameters

Status code: 200

Table 4-161 Response header parameters

Parameter	Type	Description
Port-ID	String	Definition Port ID of the cluster control plane node Constraints N/A Range N/A Default Value N/A

Table 4-162 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Config
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v1
preferences	Object	Definition This field is not in use. Constraints N/A Range N/A Default Value Empty
clusters	Array of Clusters objects	Definition Cluster list Constraints N/A
users	Array of Users objects	Definition Certificate information and client key information of a specified user Constraints N/A

Parameter	Type	Description
contexts	Array of Contexts objects	Definition Context list Constraints N/A
current-context	String	Definition Current context Constraints N/A Range • external: public network access • internal: private network access Default Value • If publicIp (a VM's EIP) is present, the value is external. • If publicIp is not present, the value is internal.

Table 4-163 Clusters

Parameter	Type	Description
name	String	Definition Cluster name Constraints N/A Range • internalCluster: a cluster that can be accessed through a private network • externalCluster: a cluster that can be accessed through the public network Default Value • If publicIp (a VM's EIP) is not present, there is only one cluster in the cluster list, and the value of this parameter is internalCluster. • If publicIp is present, there are more than one cluster in the cluster list, and the value of name of all extended clusters is externalCluster.
cluster	ClusterCert object	Definition Cluster information Constraints N/A

Table 4-164 ClusterCert

Parameter	Type	Description
server	String	Definition Server address Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
certificate-authority-data	String	Definition Certificate authorization data Constraints N/A Range N/A Default Value N/A
insecure-skip-tls-verify	Boolean	Definition Whether to skip server certificate verification Constraints N/A Range - true - false Default Value If the cluster type is externalCluster , the value is true .

Table 4-165 Users

Parameter	Type	Description
name	String	Definition Name Constraints N/A Range N/A Default Value user
user	User object	Definition Certificate information and client key information of a specified user Constraints N/A

Table 4-166 User

Parameter	Type	Description
client-certificate-data	String	Definition Client certificate Constraints N/A Range N/A Default Value N/A
client-key-data	String	Definition PEM encoding data from the TLS client key file Constraints N/A Range N/A Default Value N/A

Table 4-167 Contexts

Parameter	Type	Description
name	String	Definition Context name Constraints N/A Range • internal: private network access • external: public network access Default Value • If publicIp (a VM's EIP) is not present, there is only one cluster in the cluster list, and the value of this parameter is internal. • If publicIp is present, there are more than one cluster in the cluster list, and the value of name of all extended contexts is external.

Parameter	Type	Description
context	Context object	Definition Context Constraints N/A

Table 4-168 Context

Parameter	Type	Description
cluster	String	Definition Cluster context Constraints N/A Range N/A Default Value N/A
user	String	Definition User context Constraints N/A Range N/A Default Value N/A

Example Requests

Applying for a cluster access certificate valid for 30 days

```
{
  "duration" : 30
}
```

Example Responses

Status code: 200

The certificate of the specified cluster is successfully obtained. For details about the certificate file format, see the Kubernetes v1.Config structure.

```
{
  "kind" : "Config",
  "apiVersion" : "v1",

```

```
"preferences" : { },
"clusters" : [ {
  "name" : "internalCluster",
  "cluster" : {
    "server" : "https://192.168.1.7:5443",
    "certificate-authority-data" : "Q2VydGhmaWNhdGU6*****FTkQgQ0VSVElGSUNBVEUtLS0tLQo="
  }
}],
"users" : [ {
  "name" : "user",
  "user" : {
    "client-certificate-data" : "LS0tLS1CRUdJTiBDR*****QVRFLS0tLS0K",
    "client-key-data" : "LS0tLS1CRUdJTi*****BLRVktLS0tLQo="
  }
}],
"contexts" : [ {
  "name" : "internal",
  "context" : {
    "cluster" : "internalCluster",
    "user" : "user"
  }
}],
"current-context" : "internal"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Applying for a cluster access certificate valid for 30 days

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class CreateKubernetesClusterCertSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        CreateKubernetesClusterCertRequest request = new CreateKubernetesClusterCertRequest();
```

```
request.withClusterId("{cluster_id}");
CertDuration body = new CertDuration();
body.withDuration(30);
request.withBody(body);
try {
    CreateKubernetesClusterCertResponse response = client.createKubernetesClusterCert(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Applying for a cluster access certificate valid for 30 days

coding: utf-8

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateKubernetesClusterCertRequest()
        request.cluster_id = "{cluster_id}"
        request.body = CertDuration(
            duration=30
        )
        response = client.create_kubernetes_cluster_cert(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Applying for a cluster access certificate valid for 30 days


```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateKubernetesClusterCertRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.CertDuration{
        Duration: int32(30),
    }
    response, err := client.CreateKubernetesClusterCert(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The certificate of the specified cluster is successfully obtained. For details about the certificate file format, see the Kubernetes v1.Config structure.

Error Codes

See [Error Codes](#).

4.2.9 Revoking a Cluster Certificate of a User

Function

This API is used to revoke a certificate of a specified cluster.

 NOTE

Once the cluster certificate is revoked, the certificate applicant will no longer be able to use the downloaded certificate and kubectl configuration files to access the cluster. However, the applicant can simply download the files again and use the newly downloaded ones to regain access.

Constraints

Before revoking a cluster certificate, you need to get the user ID, which can be obtained in either of the following ways:

- Method 1: Obtain the certificate downloaded by the applicant. The name (**CN - Common Name**) of the certificate is the required user ID.
- Method 2: If you cannot obtain the certificate downloaded by the applicant, use CTS to obtain the events of deleting a user (**deleteUser**) and deleting an agency (**deleteAgency**). The resource IDs of the events are the IDs of the deleted user and delegated account, respectively.

If the ID still cannot be obtained, submit a service ticket.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/clustercertrevoke

Table 4-169 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-170 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-171 Request body parameters

Parameter	Mandatory	Type	Description
userId	No	String	Definition User ID. For details about how to obtain the value, see Revoking a Cluster Certificate of a User . Constraints This parameter is mutually exclusive with agencyId . Specify either of them. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
agencyId	No	String	Definition Agency ID. For details about how to obtain the value, see Revoking a Cluster Certificate of a User . Constraints This parameter is mutually exclusive with userId . Specify either of them. Range N/A Default Value N/A

Response Parameters

Status code: 200

A user cluster certificate has been revoked.

None

Example Requests

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/clustercertrevoke
{
  "userId" : "537e7a3bd*****6c5657fd908ff"
}
```

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;
```

```
public class RevokeKubernetesClusterCertSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        RevokeKubernetesClusterCertRequest request = new RevokeKubernetesClusterCertRequest();
        request.withClusterId("{cluster_id}");
        CertRevokeConfigRequestBody body = new CertRevokeConfigRequestBody();
        body.withUserId("537e7a3bd*****6c5657fd908ff");
        request.withBody(body);
        try {
            RevokeKubernetesClusterCertResponse response = client.revokeKubernetesClusterCert(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()
```

```
try:
    request = RevokeKubernetesClusterCertRequest()
    request.cluster_id = "{cluster_id}"
    request.body = CertRevokeConfigRequestBody(
        user_id="537e7a3bd*****6c5657fd908ff"
    )
    response = client.revoke_kubernetes_cluster_cert(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RevokeKubernetesClusterCertRequest{}
    request.ClusterId = "{cluster_id}"
    request.UserIdCertRevokeConfigRequestBody := "537e7a3bd*****6c5657fd908ff"
    request.Body = &model.CertRevokeConfigRequestBody{
        UserId: &userIdCertRevokeConfigRequestBody,
    }
    response, err := client.RevokeKubernetesClusterCert(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	A user cluster certificate has been revoked.

Error Codes

See [Error Codes](#).

4.2.10 Modifying Cluster Specifications

Function

This API is used to modify the specifications of a cluster.

NOTE

- The URL for cluster management is in the format of **https://Endpoint/uri**, in which *uri* indicates the resource path, that is, the path for API access.
- For details about constraints, see [Changing Cluster Scale](#).

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/resize

Table 4-172 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-173 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token. Constraints: None Options: N/A Default value: N/A

Table 4-174 Request body parameters

Parameter	Mandatory	Type	Description
flavorResize	Yes	String	Definition The target flavor to be changed. You can only adjust the maximum number of worker nodes in a cluster. The number of control plane nodes is fixed and cannot be altered, and the cluster flavor cannot be downgraded. Suppose the original cluster flavor is cce.s2.medium. In that case, you can only upgrade it to cce.s2.large or higher, but you cannot downgrade it to cce.s2.small or cce.s1.medium. Constraints N/A Range • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with

Parameter	Mandatory	Type	Description
			three control plane nodes and a maximum of 2,000 worker nodes Default Value N/A NOTE The fields in the parameters are described as follows: • s1: specifies a cluster with one control plane node. If the control plane node is faulty, the cluster will become unavailable, but running workloads in the cluster are not affected. • s2: specifies an HA cluster with three control plane nodes. If one of the control plane nodes is faulty, the cluster is still available. • dec: specifies a Dec CCE cluster. For example, cce.dec.s1.small specifies a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes. • small: specifies that a cluster can manage a maximum of 50 worker nodes. • medium: specifies that a cluster can manage a maximum of 200 worker nodes. • large: specifies that a cluster can manage a maximum of 1,000 worker nodes. • xlarge: specifies that a cluster can manage a maximum of 2,000 worker nodes.

Parameter	Mandatory	Type	Description
skippedTasks	No	Array of strings	Definition Whether to skip some tasks during a cluster flavor change Constraints N/A Range • IngressChecker: The check of configuration consistency between the ingress and ELB is skipped during the cluster flavor change. NOTE • Skipping this consistency check may interrupt services. • If the cluster is unavailable or overloaded, this consistency check must be skipped. If it is not skipped, the cluster flavor change will fail. Before changing the cluster flavor, ensure that the consistency problem has been resolved by referring to What Should I Do If the LoadBalancer Ingress Configuration Is Inconsistent with the Load Balancer Configuration During a CCE Cluster Upgrade? Default Value An unavailable cluster includes IngressChecker by default.
extendParam	No	extendParam object	Definition Extended field for changing cluster flavors Constraints N/A

Table 4-175 extendParam

Parameter	Mandatory	Type	Description
decMasterFlavor	No	String	Definition The flavors of the control plane nodes can be specified for a DeC CCE cluster. Constraints N/A Range N/A Default Value N/A
isAutoPay	No	String	Definition Whether to enable auto payment Constraints N/A Range • true: Auto payment is enabled. • false: Auto payment is not enabled. NOTE This parameter is valid only for clusters billed on a yearly/monthly basis. If this parameter is not set, auto payment is disabled by default. Default Value N/A

Response Parameters

Status code: 201

Table 4-176 Response body parameters

Parameter	Type	Description
jobID	String	Definition Job ID Constraints N/A Range N/A Default Value N/A
orderId	String	Definition ID of the order for modifying the flavor of a yearly/monthly cluster Constraints N/A Range N/A Default Value N/A

Example Requests

- Modifying the specifications of the yearly/monthly cluster (automatic payment)

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/resize
```

```
{
  "flavorResize": "cce.s1.medium",
  "extendParam": {
    "isAutoPay": "true"
  }
}
```

- Modifying the specifications of the pay-per-use cluster

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/resize
```

```
{
  "flavorResize": "cce.s1.medium"
}
```

Example Responses

Status code: 201

The job for modifying the specifications of the pay-per-use cluster is delivered.

```
{
  "jobID": "13b8d958-8fcf-11ed-aef3-0255ac1001bd"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Modifying the specifications of the yearly/monthly cluster (automatic payment)

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ResizeClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ResizeClusterRequest request = new ResizeClusterRequest();
        request.withClusterId("{cluster_id}");
        ResizeClusterRequestBody body = new ResizeClusterRequestBody();
        ResizeClusterRequestBodyExtendParam extendParambody = new
        ResizeClusterRequestBodyExtendParam();
        extendParambody.withIsAutoPay("true");
        body.withExtendParam(extendParambody);
        body.withFlavorResize("cce.s1.medium");
        request.withBody(body);
        try {
            ResizeClusterResponse response = client.resizeCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```


- Modifying the specifications of the pay-per-use cluster

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ResizeClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ResizeClusterRequest request = new ResizeClusterRequest();
        request.withClusterId("{cluster_id}");
        ResizeClusterRequestBody body = new ResizeClusterRequestBody();
        body.withFlavorResize("cce.s1.medium");
        request.withBody(body);
        try {
            ResizeClusterResponse response = client.resizeCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

- Modifying the specifications of the yearly/monthly cluster (automatic payment)

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ResizeClusterRequest()
        request.cluster_id = "{cluster_id}"
        extendParambody = ResizeClusterRequestBodyExtendParam(
            is_auto_pay="true"
        )
        request.body = ResizeClusterRequestBody(
            extend_param=extendParambody,
            flavor_resize="cce.s1.medium"
        )
        response = client.resize_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

- **Modifying the specifications of the pay-per-use cluster**

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ResizeClusterRequest()
        request.cluster_id = "{cluster_id}"
        request.body = ResizeClusterRequestBody(
            flavor_resize="cce.s1.medium"
        )
    )
```

```
response = client.resize_cluster(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- Modifying the specifications of the yearly/monthly cluster (automatic payment)

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ResizeClusterRequest{}
    request.ClusterId = "{cluster_id}"
    isAutoPayExtendParam := "true"
    extendParambody := &model.ResizeClusterRequestBodyExtendParam{
        IsAutoPay: &isAutoPayExtendParam,
    }
    request.Body = &model.ResizeClusterRequestBody{
        ExtendParam: extendParambody,
        FlavorResize: "cce.s1.medium",
    }
    response, err := client.ResizeCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- Modifying the specifications of the pay-per-use cluster

```
package main

import (
    "fmt"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ResizeClusterRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.ResizeClusterRequestBody{
        FlavorResize: "cce.s1.medium",
    }
    response, err := client.ResizeCluster(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	The job for modifying the specifications of the pay-per-use cluster is delivered.

Error Codes

See [Error Codes](#).

4.2.11 Querying a Job

Function

This API is used to obtain information about a job via the job ID returned after a query request is delivered.

NOTE

- The URL for cluster management is in the format of **https://Endpoint/uri**. In the URL, **uri** indicates the resource path, that is, the path for API access.
- You can call this API when:
 - Creating or deleting a cluster
 - Creating or deleting a node

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/jobs/{job_id}

Table 4-177 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
job_id	Yes	String	Definition Job ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-178 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-179 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Job
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3

Parameter	Type	Description
metadata	JobMetadata object	Definition Job metadata Constraints N/A
spec	JobSpec object	Definition Detailed job parameters Constraints N/A
status	JobStatus object	Definition Job status Constraints N/A

Table 4-180 JobSpec

Parameter	Type	Description
type	String	Definition Job type Constraints N/A Range N/A Default Value N/A
clusterUID	String	Definition ID of the cluster where the job runs Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
resourceID	String	Definition ID of the resource on which the job is executed Constraints N/A Range N/A Default Value N/A
resourceName	String	Definition Name of the resource on which the job is executed Constraints N/A Range N/A Default Value N/A
extendParam	Map<String,String>	Definition Extended parameters Constraints N/A Range N/A Default Value N/A
subJobs	Array of Job objects	Definition Subjob list. • The list contains details about all subjobs. • In scenarios such as cluster and node creation, a job typically consists of multiple subjobs. The job is complete only after all subjobs are complete. Constraints N/A

Table 4-181 Job

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Job
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	JobMetadata object	Definition Job metadata Constraints N/A
spec	JobSpec object	Definition Detailed job parameters Constraints N/A
status	JobStatus object	Definition Job status Constraints N/A

Table 4-182 JobMetadata

Parameter	Type	Description
uid	String	Definition Job ID Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Time when a job was created Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Time when a job was updated Constraints N/A Range N/A Default Value N/A

Table 4-183 JobStatus

Parameter	Type	Description
phase	String	Definition Job status Constraints N/A Range • Initializing: The job is being initialized. • Running: The job is running. • Failed: The job failed. • Success: The job succeeded. Default Value N/A
reason	String	Definition Why a job is in the current state Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

The progress of the specified job is successfully obtained.

```
{
  "kind": "Job",
  "apiVersion": "v3",
  "metadata": {
    "uid": "354331b2c-229a-11e8-9c75-0255ac100ceb",
    "creationTimestamp": "2018-08-02 08:12:40.672772389 +0000 UTC",
    "updateTimestamp": "2018-08-02 08:21:50.478108569 +0000 UTC"
  },
  "spec": {
    "type": "CreateCluster",
    "clusterUID": "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
    "resourceID": "6f4dcb2c-229a-11e8-9c75-0255ac100ceb",
    "resourceName": "cluster-name",
    "extendParam": {
      "serverID": "bc467e3a-2338-11e8-825b-0255ac100c13"
    }
  }
}
```

```
    },
    "subJobs": [ {
      "kind": "Job",
      "apiVersion": "v3",
      "metadata": {
        "uid": "fd474fab-9606-11e8-baa9-0255ac10215d",
        "creationTimestamp": "2018-08-02 03:52:34.615819618 +0000 UTC",
        "updateTimestamp": "2018-08-02 04:05:29.196243031 +0000 UTC"
      },
      "spec": {
        "type": "InstallMaster",
        "clusterUID": "fcc72de0-9606-11e8-baa8-0255ac10215d",
        "resourceID": "fd3b4ac0-9606-11e8-baa8-0255ac10215d",
        "extendParam": {
          "serverID": "fd3b4ac0-9606-11e8-baa8-0255ac10215d"
        }
      },
      "status": {
        "phase": "Success"
      }
    }, {
      "kind": "Job",
      "apiVersion": "v3",
      "metadata": {
        "uid": "fd474f82-9606-11e8-baa8-0255ac10215d",
        "creationTimestamp": "2018-08-02 03:52:33.859150791 +0000 UTC",
        "updateTimestamp": "2018-08-02 03:52:34.615655429 +0000 UTC"
      },
      "spec": {
        "type": "CreatePSMCert",
        "clusterUID": "fcc72de0-9606-11e8-baa8-0255ac10215d"
      },
      "status": {
        "phase": "Success"
      }
    }
  ],
  "status": {
    "phase": "Running",
    "reason": ""
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
```

this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ShowJobRequest request = new ShowJobRequest();
request.withJobId("{job_id}");
try {
    ShowJobResponse response = client.showJob(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowJobRequest()
        request.job_id = "{job_id}"
        response = client.show_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowJobRequest{}
    request.JobId = "{job_id}"
    response, err := client.ShowJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The progress of the specified job is successfully obtained.

Error Codes

See [Error Codes](#).

4.2.12 Binding/Unbinding Public API Server Address

Function

This API is used to bind or unbind the public API server address of the cluster by cluster ID.

 NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/mastereip

Table 4-184 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-185 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-186 Request body parameters

Parameter	Mandatory	Type	Description
spec	Yes	MasterEIPRequestSpec object	Definition Request parameters for binding or unbinding the public API server address of a cluster Constraints N/A

Table 4-187 MasterEIPRequestSpec

Parameter	Mandatory	Type	Description
action	Yes	String	Definition Binding or unbinding action Constraints N/A Range • bind: specifies the binding action. • unbind: specifies the unbinding action. Default Value N/A
spec	No	spec object	Definition Configuration attributes of the EIP to be bound Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
bandwidth	No	String	Definition Bandwidth (This field has expired and is not recommended.) Constraints N/A Range N/A Default Value N/A
elasticip	No	String	Definition EIP (This field has expired and is not recommended.) Constraints N/A Range N/A Default Value N/A

Table 4-188 spec

Parameter	Mandatory	Type	Description
id	No	String	Definition EIP ID. This parameter is mandatory for binding an EIP and is invalid for unbinding an EIP. NOTE To obtain the ID, log in to the management console, choose Service List > Networking > Elastic IP , click the EIP name to access the details page, and copy the ID on the Summary tab. Constraints N/A Range N/A Default Value N/A

Response Parameters

Status code: 200

Table 4-189 Response body parameters

Parameter	Type	Description
metadata	Metadata object	Definition Basic information. Metadata is a collection of attributes. Constraints N/A
spec	MasterEIPResponseSpec object	Definition Configuration about the public API server address bound to a cluster Constraints N/A
status	status object	Definition Status Constraints N/A

Table 4-190 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-191 MasterEIPResponseSpec

Parameter	Type	Description
action	String	Definition Binding action Constraints N/A Range N/A Default Value N/A
spec	spec object	Definition Configuration attributes of the EIP to be bound Constraints N/A
elasticIp	String	Definition EIP Constraints N/A Range N/A Default Value N/A

Table 4-192 spec

Parameter	Type	Description
id	String	Definition EIP ID Constraints N/A Range N/A Default Value N/A
eip	EipSpec object	Definition EIP details Constraints N/A
IsDynamic	Boolean	Definition Dynamic provisioning or not Constraints N/A Range N/A Default Value N/A

Table 4-193 EipSpec

Parameter	Type	Description
bandwidth	bandwidth object	Definition Bandwidth information Constraints N/A

Table 4-194 bandwidth

Parameter	Type	Description
size	Integer	Definition Bandwidth size Constraints N/A Range N/A Default Value N/A
sharetype	String	Definition Bandwidth type Constraints N/A Range • PER: dedicated bandwidth • WHOLE: shared bandwidth Default Value N/A

Table 4-195 status

Parameter	Type	Description
privateEndpoint	String	Definition Private IP address for accessing a cluster (VIP in the case of an HA cluster) Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
publicEndpoint	String	Definition Public IP address for accessing a cluster Constraints N/A Range N/A Default Value N/A

Example Requests

Bind the public API server address to a cluster.

```
{
  "spec": {
    "action": "bind",
    "spec": {
      "id": "a757a69e-f920-455a-b1ba-d7a22db0fd50"
    }
  }
}
```

Example Responses

Status code: 200

This indicates that the public API server address of the cluster is successfully bound. No response body is returned if the unbinding is successful.

```
{
  "metadata": { },
  "spec": {
    "action": "bind",
    "spec": {
      "id": "a757a69e-f920-455a-b1ba-d7a22db0fd50",
      "eip": {
        "bandwidth": {
          "size": 5,
          "sharetype": "PER"
        }
      },
      "IsDynamic": false
    },
    "elasticIp": "8.8.8.8"
  },
  "status": {
    "privateEndpoint": "https://192.168.3.238:5443",
    "publicEndpoint": "https://8.8.8.8:5443"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Bind the public API server address to a cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpdateClusterEipSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateClusterEipRequest request = new UpdateClusterEipRequest();
        request.withClusterId("{cluster_id}");
        MasterEIPRequest body = new MasterEIPRequest();
        MasterEIPRequestSpecSpec specSpec = new MasterEIPRequestSpecSpec();
        specSpec.withId("a757a69e-f920-455a-b1ba-d7a22db0fd50");
        MasterEIPRequestSpec specbody = new MasterEIPRequestSpec();
        specbody.withAction(MasterEIPRequestSpec.ActionEnum.fromValue("bind"))
            .withSpec(specSpec);
        body.withSpec(specbody);
        request.withBody(body);
        try {
            UpdateClusterEipResponse response = client.updateClusterEip(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Bind the public API server address to a cluster.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateClusterEipRequest()
        request.cluster_id = "{cluster_id}"
        specSpec = MasterEIPRequestSpecSpec(
            id="a757a69e-f920-455a-b1ba-d7a22db0fd50"
        )
        specbody = MasterEIPRequestSpec(
            action="bind",
            spec=specSpec
        )
        request.body = MasterEIPRequest(
            spec=specbody
        )
        response = client.update_cluster_eip(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Bind the public API server address to a cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateClusterEipRequest{}
request.ClusterId = "{cluster_id}"
idSpec := "a757a69e-f920-455a-b1ba-d7a22db0fd50"
specSpec := &model.MasterEipRequestSpecSpec{
    Id: &idSpec,
}
actionSpec := model.GetMasterEipRequestSpecActionEnum().BIND
specbody := &model.MasterEipRequestSpec{
    Action: &actionSpec,
    Spec: specSpec,
}
request.Body = &model.MasterEipRequest{
    Spec: specbody,
}
response, err := client.UpdateClusterEip(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	This indicates that the public API server address of the cluster is successfully bound. No response body is returned if the unbinding is successful.

Error Codes

See [Error Codes](#).

4.2.13 Obtaining Cluster Access Address

Function

This API is used to obtain the cluster access addresses by cluster ID, including the private IP (VIP returned by an HA cluster) and public IP.

 NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/openapi

Table 4-196 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-197 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-198 Response body parameters

Parameter	Type	Description
metadata	Metadata object	Definition Basic information. Metadata is a collection of attributes. Constraints N/A
spec	OpenAPISpec object	Definition Parameters for configuring the cluster access address Constraints N/A
status	status object	Definition Status Constraints N/A

Table 4-199 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-200 OpenAPISpec

Parameter	Type	Description
spec	spec object	Definition Cluster access address Constraints N/A

Table 4-201 spec

Parameter	Type	Description
eip	EipSpec object	Definition EIP details Constraints N/A
IsDynamic	Boolean	Definition Dynamic provisioning or not Constraints N/A Range N/A Default Value N/A

Table 4-202 EipSpec

Parameter	Type	Description
bandwidth	bandwidth object	Definition Bandwidth information Constraints N/A

Table 4-203 bandwidth

Parameter	Type	Description
size	Integer	Definition Bandwidth size Constraints N/A Range N/A Default Value N/A
sharetype	String	Definition Bandwidth type Constraints N/A Range • PER: dedicated bandwidth • WHOLE: shared bandwidth Default Value N/A

Table 4-204 status

Parameter	Type	Description
privateEndpoint	String	Definition Private IP address for accessing a cluster (VIP in the case of an HA cluster) Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
publicEndpoint	String	Definition Public IP address for accessing a cluster Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

The cluster access address is obtained successfully.

```
{
  "metadata" : { },
  "spec" : {
    "spec" : {
      "eip" : {
        "bandwidth" : { }
      },
      "IsDynamic" : false
    }
  },
  "status" : {
    "privateEndpoint" : "https://192.168.3.238:5443",
    "publicEndpoint" : ""
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;
```

```
public class ShowClusterEndpointsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        ShowClusterEndpointsRequest request = new ShowClusterEndpointsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowClusterEndpointsResponse response = client.showClusterEndpoints(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowClusterEndpointsRequest()
```

```
request.cluster_id = "{cluster_id}"
response = client.show_cluster_endpoints(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowClusterEndpointsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowClusterEndpoints(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The cluster access address is obtained successfully.

Error Codes

See [Error Codes](#).

4.2.14 Obtaining a Cluster's Logging Configurations

Function

This API is used to obtain LTS configurations reported by the cluster components.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/cluster/{cluster_id}/log-configs

Table 4-205 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-206 Query Parameters

Parameter	Mandatory	Type	Description
type	No	String	Definition Component type. If this parameter is not specified, all component types will be obtained. Constraints The value can be control, audit, or system-addon. Range • control: specifies the logs of the control plane components. • audit: specifies the audit logs on the control plane. • system-addon: specifies the logs of the system add-ons. Default Value N/A

Request Parameters

Table 4-207 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-208 Response body parameters

Parameter	Type	Description
ttl_in_days	Integer	Definition How long the logs will be stored, in days Range 0 to 30
log_configs	Array of log_configs objects	Definition Detailed information about log configuration items Constraints N/A

Table 4-209 log_configs

Parameter	Type	Description
name	String	Definition Log type Constraints The value must be kube-apiserver , kube-controller-manager , kube-scheduler , audit , or a system add-on name. Range • kube-apiserver: The kube-apiserver logs will be collected. • kube-controller-manager: The kube-controller-manager logs will be collected. • kube-scheduler: The kube-scheduler logs will be collected. • audit: The audit logs will be collected. • <i>System add-on name</i>: The system add-on logs will be collected. Default Value N/A

Parameter	Type	Description
enable	Boolean	Definition Whether to enable log collection Range • true: Log collection is enabled. • false: Log collection is disabled.
type	String	Definition Component type. The value can be control , audit , or system-addon . Constraints • This parameter is returned only in the response of the API for querying cluster log configurations. • This parameter cannot be configured when it is used as an update parameter in the API for configuring cluster logs. Range • control: specifies the logs of the control plane components. • audit: specifies the audit logs on the control plane. • system-addon: specifies the logs of the system add-ons. Default Value N/A

Example Requests

The following example describes how to obtain add-on log configurations.

```
/api/v3/projects/{project_id}/cluster/{cluster_id}/log-configs?type=system-addon
```

Example Responses

Status code: 200

Cluster log configurations are obtained successfully.

```
{  
  "log_configs": [ {  
    "name": "volcano",  
    "enable": true  
  }, {  
    "name": "coredns",  
    "enable": false  
  }, {  
    "name": "everest",  
    "enable": false  
  }  
]
```

```
    }  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
public class ShowClusterConfigSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        CceClient client = CceClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ShowClusterConfigRequest request = new ShowClusterConfigRequest();  
        request.withClusterId("{cluster_id}");  
        try {  
            ShowClusterConfigResponse response = client.showClusterConfig(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials
```

```
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowClusterConfigRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_cluster_config(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowClusterConfigRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowClusterConfig(request)
```

```
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Cluster log configurations are obtained successfully.

Error Codes

See [Error Codes](#).

4.2.15 Configuring Cluster Logs

Function

This API is used to select the master node components whose logs are reported to LTS.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/cluster/{cluster_id}/log-configs

Table 4-210 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-211 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-212 Request body parameters

Parameter	Mandatory	Type	Description
ttl_in_days	No	Integer	Definition How long the logs will be stored, in days Range 0 to 30
log_configs	No	Array of log_configs objects	Definition Detailed information about log configuration items Constraints N/A

Table 4-213 log_configs

Parameter	Mandatory	Type	Description
name	No	String	Definition Log type Constraints The value must be kube-apiserver , kube-controller-manager , kube-scheduler , audit , or a system add-on name. Range • kube-apiserver: The kube-apiserver logs will be collected. • kube-controller-manager: The kube-controller-manager logs will be collected. • kube-scheduler: The kube-scheduler logs will be collected. • audit: The audit logs will be collected. • <i>System add-on name</i>: The system add-on logs will be collected. Default Value N/A

Parameter	Mandatory	Type	Description
enable	No	Boolean	Definition Whether to enable log collection Range • true: Log collection is enabled. • false: Log collection is disabled.
type	No	String	Definition Component type. The value can be control , audit , or system-addon . Constraints • This parameter is returned only in the response of the API for querying cluster log configurations. • This parameter cannot be configured when it is used as an update parameter in the API for configuring cluster logs. Range • control: specifies the logs of the control plane components. • audit: specifies the audit logs on the control plane. • system-addon: specifies the logs of the system add-ons. Default Value N/A

Response Parameters

Status code: 200

Table 4-214 Response body parameters

Parameter	Type	Description
ttl_in_days	Integer	Definition How long the logs will be stored, in days Range 0 to 30
log_configs	Array of log_configs objects	Definition Detailed information about log configuration items Constraints N/A

Table 4-215 log_configs

Parameter	Type	Description
name	String	Definition Log type Constraints The value must be kube-apiserver , kube-controller-manager , kube-scheduler , audit , or a system add-on name. Range • kube-apiserver: The kube-apiserver logs will be collected. • kube-controller-manager: The kube-controller-manager logs will be collected. • kube-scheduler: The kube-scheduler logs will be collected. • audit: The audit logs will be collected. • <i>System add-on name</i>: The system add-on logs will be collected. Default Value N/A

Parameter	Type	Description
enable	Boolean	Definition Whether to enable log collection Range • true: Log collection is enabled. • false: Log collection is disabled.
type	String	Definition Component type. The value can be control , audit , or system-addon . Constraints • This parameter is returned only in the response of the API for querying cluster log configurations. • This parameter cannot be configured when it is used as an update parameter in the API for configuring cluster logs. Range • control: specifies the logs of the control plane components. • audit: specifies the audit logs on the control plane. • system-addon: specifies the logs of the system add-ons. Default Value N/A

Example Requests

The following example describes how to configure cluster logs that are reported to LTS.

```
/api/v3/projects/{project_id}/cluster/{cluster_id}/log-configs
```

```
{
  "log_configs": [ {
    "name": "kube-apiserver",
    "enable": true
  }, {
    "name": "kube-controller-manager",
    "enable": false
  }, {
    "name": "kube-scheduler",
    "enable": false
  }, {
    "name": "volcano",
    "enable": true
  }, {
    "name": "coredns",
    "enable": false
  } ]
}
```

```
}, {  
  "name" : "everest",  
  "enable" : false  
}]  
}
```

Example Responses

Status code: 200

Cluster logs are configured successfully.

```
{  
  "ttl_in_days" : 7,  
  "log_configs" : [ {  
    "name" : "kube-apiserver",  
    "enable" : true  
  }, {  
    "name" : "kube-controller-manager",  
    "enable" : false  
  }, {  
    "name" : "kube-scheduler",  
    "enable" : false  
  }, {  
    "name" : "volcano",  
    "enable" : true  
  }, {  
    "name" : "coredns",  
    "enable" : false  
  }, {  
    "name" : "everest",  
    "enable" : false  
  } ]  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

The following example describes how to configure cluster logs that are reported to LTS.

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
import java.util.List;  
import java.util.ArrayList;  
  
public class UpdateClusterLogConfigSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateClusterLogConfigRequest request = new UpdateClusterLogConfigRequest();
request.withClusterId("{cluster_id}");
ClusterLogConfig body = new ClusterLogConfig();
List<ClusterLogConfigLogConfigs> listbodyLogConfigs = new ArrayList<>();
listbodyLogConfigs.add(
    new ClusterLogConfigLogConfigs()
        .withName("kube-apiserver")
        .withEnable(true)
);
listbodyLogConfigs.add(
    new ClusterLogConfigLogConfigs()
        .withName("kube-controller-manager")
        .withEnable(false)
);
listbodyLogConfigs.add(
    new ClusterLogConfigLogConfigs()
        .withName("kube-scheduler")
        .withEnable(false)
);
listbodyLogConfigs.add(
    new ClusterLogConfigLogConfigs()
        .withName("volcano")
        .withEnable(true)
);
listbodyLogConfigs.add(
    new ClusterLogConfigLogConfigs()
        .withName("coredns")
        .withEnable(false)
);
listbodyLogConfigs.add(
    new ClusterLogConfigLogConfigs()
        .withName("everest")
        .withEnable(false)
);
body.withLogConfigs(listbodyLogConfigs);
request.withBody(body);
try {
    UpdateClusterLogConfigResponse response = client.updateClusterLogConfig(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

The following example describes how to configure cluster logs that are reported to LTS.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateClusterLogConfigRequest()
        request.cluster_id = "{cluster_id}"
        listLogConfigsbody = [
            ClusterLogConfigLogConfigs(
                name="kube-apiserver",
                enable=True
            ),
            ClusterLogConfigLogConfigs(
                name="kube-controller-manager",
                enable=False
            ),
            ClusterLogConfigLogConfigs(
                name="kube-scheduler",
                enable=False
            ),
            ClusterLogConfigLogConfigs(
                name="volcano",
                enable=True
            ),
            ClusterLogConfigLogConfigs(
                name="coredns",
                enable=False
            ),
            ClusterLogConfigLogConfigs(
                name="everest",
                enable=False
            )
        ]
        request.body = ClusterLogConfig(
            log_configs=listLogConfigsbody
        )
        response = client.update_cluster_log_config(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

The following example describes how to configure cluster logs that are reported to LTS.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateClusterLogConfigRequest{}
    request.ClusterId = "{cluster_id}"
    nameLogConfigs := "kube-apiserver"
    enableLogConfigs := true
    nameLogConfigs1 := "kube-controller-manager"
    enableLogConfigs1 := false
    nameLogConfigs2 := "kube-scheduler"
    enableLogConfigs2 := false
    nameLogConfigs3 := "volcano"
    enableLogConfigs3 := true
    nameLogConfigs4 := "coredns"
    enableLogConfigs4 := false
    nameLogConfigs5 := "everest"
    enableLogConfigs5 := false
    var listLogConfigsboby = []model.ClusterLogConfigLogConfigs{
        {
            Name: &nameLogConfigs,
            Enable: &enableLogConfigs,
        },
        {
            Name: &nameLogConfigs1,
            Enable: &enableLogConfigs1,
        },
        {
            Name: &nameLogConfigs2,
            Enable: &enableLogConfigs2,
        },
        {
            Name: &nameLogConfigs3,
            Enable: &enableLogConfigs3,
        },
    }
```

```
        Name: &nameLogConfigs4,
        Enable: &enableLogConfigs4,
    },
    {
        Name: &nameLogConfigs5,
        Enable: &enableLogConfigs5,
    },
}
request.Body = &model.ClusterLogConfig{
    LogConfigs: &listLogConfigsbody,
}
response, err := client.UpdateClusterLogConfig(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Cluster logs are configured successfully.

Error Codes

See [Error Codes](#).

4.2.16 Obtaining the Partition List

Function

This API is used to obtain the partition list.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/partitions

Table 4-216 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-217 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value List
apiVersion	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
items	Array of Partition objects	Definition Cluster partition information Constraints N/A

Table 4-218 Partition

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Partition

Parameter	Type	Description
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	metadata object	Definition Partition metadata Constraints N/A
spec	spec object	Definition Partition configurations Constraints N/A Range N/A Default Value N/A

Table 4-219 metadata

Parameter	Type	Description
name	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-220 spec

Parameter	Type	Description
hostNetwork	hostNetwork object	Definition Partition subnet Constraints N/A Range N/A Default Value N/A
containerNetwork	Array of containerNetwork objects	Definition Container subnet in a partition Constraints N/A Range N/A Default Value N/A
publicBorderGroup	String	Definition Group. In the CloudPond scenario, the value is the AZ ID. In the central region, the value is center . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
category	String	Definition AZ type Constraints N/A Range • Default: central AZ • IES: dedicated cloud AZ • HomeZone: Intelligent EdgeCloud AZ Default Value N/A

Table 4-221 hostNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Table 4-222 containerNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "kind" : "List",
  "apiVersion" : "v3",
  "items" : [ {
    "kind" : "Partition",
    "apiVersion" : "v3",
    "metadata" : {
      "name" : "partitionName",
      "creationTimestamp" : "2000-1-1 00:00:35.451967 +0000 UTC"
    },
    "spec" : {
      "hostNetwork" : {
        "subnetID" : "subnetID"
      },
      "containerNetwork" : [ {
        "subnetID" : "subnetID"
      } ],
      "publicBorderGroup" : "publicBorderGroup",
      "category" : "category"
    }
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListPartitionsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
```

```
.withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ListPartitionsRequest request = new ListPartitionsRequest();
request.withClusterId("{cluster_id}");
try {
    ListPartitionsResponse response = client.listPartitions(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListPartitionsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_partitions(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListPartitionsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListPartitions(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.17 Creating a Partition

Function

This API is used to create a partition.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/partitions

Table 4-223 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-224 Request body parameters

Parameter	Mandatory	Type	Description
kind	No	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value Partition
apiVersion	No	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
metadata	No	metadata object	Definition Partition metadata Constraints N/A
spec	No	spec object	Definition Partition configurations Constraints N/A Range N/A Default Value N/A

Table 4-225 metadata

Parameter	Mandatory	Type	Description
name	No	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Table 4-226 spec

Parameter	Mandatory	Type	Description
hostNetwork	No	hostNetwork object	Definition Partition subnet Constraints N/A Range N/A Default Value N/A
containerNet work	No	Array of containerNet work objects	Definition Container subnet in a partition Constraints The maximum list length is 20. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
publicBorderGroup	No	String	Definition Group. In the CloudPond scenario, the value is the AZ ID. In the central region, the value is center . Constraints N/A Range N/A Default Value N/A
category	No	String	Definition AZ type Constraints N/A Range • Default: central AZ • IES: dedicated cloud AZ • HomeZone: Intelligent EdgeCloud AZ Default Value N/A

Table 4-227 hostNetwork

Parameter	Mandatory	Type	Description
subnetID	No	String	Definition Subnet ID. To obtain the value: • Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. • Method 2: Use the VPC APIs. For details, see Querying Subnets. Constraints N/A Range N/A Default Value N/A

Table 4-228 containerNetwork

Parameter	Mandatory	Type	Description
subnetID	No	String	Definition Subnet ID. To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets . Constraints N/A Range N/A Default Value N/A

Response Parameters

Status code: 200

Table 4-229 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Partition

Parameter	Type	Description
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	metadata object	Definition Partition metadata Constraints N/A
spec	spec object	Definition Partition configurations Constraints N/A Range N/A Default Value N/A

Table 4-230 metadata

Parameter	Type	Description
name	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-231 spec

Parameter	Type	Description
hostNetwork	hostNetwork object	Definition Partition subnet Constraints N/A Range N/A Default Value N/A
containerNetwork	Array of containerNetwork objects	Definition Container subnet in a partition Constraints N/A Range N/A Default Value N/A
publicBorderGroup	String	Definition Group. In the CloudPond scenario, the value is the AZ ID. In the central region, the value is center . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
category	String	Definition AZ type Constraints N/A Range • Default: central AZ • IES: dedicated cloud AZ • HomeZone: Intelligent EdgeCloud AZ Default Value N/A

Table 4-232 hostNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Table 4-233 containerNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Example Requests

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/partitions

{
  "kind" : "Partition",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "partitionName"
  },
  "spec" : {
    "hostNetwork" : {
      "subnetID" : "subnetID"
    },
    "containerNetwork" : [ {
      "subnetID" : "subnetID"
    } ],
    "publicBorderGroup" : "publicBorderGroup",
    "category" : "category"
  }
}
```

Example Responses

Status code: 200

OK

```
{
  "kind" : "Partition",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "partitionName",
    "creationTimestamp" : "2000-1-1 00:00:35.451967 +0000 UTC"
  },
  "spec" : {
    "hostNetwork" : {
      "subnetID" : "subnetID"
    },
    "containerNetwork" : [ {
      "subnetID" : "subnetID"
    } ],
    "publicBorderGroup" : "publicBorderGroup",
    "category" : "category"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;
```

```
public class CreatePartitionSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR_REGION>"))
            .build();

        CreatePartitionRequest request = new CreatePartitionRequest();
        request.withClusterId("{cluster_id}");
        PartitionReqBody body = new PartitionReqBody();
        List<PartitionReqBodySpecContainerNetwork> listSpecContainerNetwork = new ArrayList<>();
        listSpecContainerNetwork.add(
            new PartitionReqBodySpecContainerNetwork()
                .withSubnetID("subnetID")
        );
        PartitionReqBodySpecHostNetwork hostNetworkSpec = new PartitionReqBodySpecHostNetwork();
        hostNetworkSpec.withSubnetID("subnetID");
        PartitionReqBodySpec specbody = new PartitionReqBodySpec();
        specbody.withHostNetwork(hostNetworkSpec)
            .withContainerNetwork(listSpecContainerNetwork)
            .withPublicBorderGroup("publicBorderGroup")
            .withCategory("category");
        PartitionReqBodyMetadata metadatabody = new PartitionReqBodyMetadata();
        metadatabody.withName("partitionName");
        body.withSpec(specbody);
        body.withMetadata(metadatabody);
        body.withApiVersion("v3");
        body.withKind("Partition");
        request.withBody(body);
        try {
            CreatePartitionResponse response = client.createPartition(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreatePartitionRequest()
        request.cluster_id = "{cluster_id}"
        listContainerNetworkSpec = [
            PartitionReqBodySpecContainerNetwork(
                subnet_id="subnetID"
            )
        ]
        hostNetworkSpec = PartitionReqBodySpecHostNetwork(
            subnet_id="subnetID"
        )
        specbody = PartitionReqBodySpec(
            host_network=hostNetworkSpec,
            container_network=listContainerNetworkSpec,
            public_border_group="publicBorderGroup",
            category="category"
        )
        metadatabody = PartitionReqBodyMetadata(
            name="partitionName"
        )
        request.body = PartitionReqBody(
            spec=specbody,
            metadata=metadatabody,
            api_version="v3",
            kind="Partition"
        )
        response = client.create_partition(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
```

example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreatePartitionRequest{}
request.ClusterId = "{cluster_id}"
subnetIDContainerNetwork:= "subnetID"
var listContainerNetworkSpec = []model.PartitionReqBodySpecContainerNetwork{
    {
        SubnetID: &subnetIDContainerNetwork,
    },
}
subnetIDHostNetwork:= "subnetID"
hostNetworkSpec := &model.PartitionReqBodySpecHostNetwork{
    SubnetID: &subnetIDHostNetwork,
}
publicBorderGroupSpec:= "publicBorderGroup"
categorySpec:= "category"
specbody := &model.PartitionReqBodySpec{
    HostNetwork: hostNetworkSpec,
    ContainerNetwork: &listContainerNetworkSpec,
    PublicBorderGroup: &publicBorderGroupSpec,
    Category: &categorySpec,
}
nameMetadata:= "partitionName"
metadatabody := &model.PartitionReqBodyMetadata{
    Name: &nameMetadata,
}
apiVersionPartitionReqBody:= "v3"
kindPartitionReqBody:= "Partition"
request.Body = &model.PartitionReqBody{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: &apiVersionPartitionReqBody,
    Kind: &kindPartitionReqBody,
}
response, err := client.CreatePartition(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.18 Obtaining Partition Details

Function

This API is used to obtain partition details.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/partitions/{partition_name}

Table 4-234 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
partition_name	Yes	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-235 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Partition

Parameter	Type	Description
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	metadata object	Definition Partition metadata Constraints N/A
spec	spec object	Definition Partition configurations Constraints N/A Range N/A Default Value N/A

Table 4-236 metadata

Parameter	Type	Description
name	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-237 spec

Parameter	Type	Description
hostNetwork	hostNetwork object	Definition Partition subnet Constraints N/A Range N/A Default Value N/A
containerNetwork	Array of containerNetwork objects	Definition Container subnet in a partition Constraints N/A Range N/A Default Value N/A
publicBorderGroup	String	Definition Group. In the CloudPond scenario, the value is the AZ ID. In the central region, the value is center . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
category	String	Definition AZ type Constraints N/A Range • Default: central AZ • IES: dedicated cloud AZ • HomeZone: Intelligent EdgeCloud AZ Default Value N/A

Table 4-238 hostNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Table 4-239 containerNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "kind" : "Partition",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "partitionName",
    "creationTimestamp" : "2000-1-1 00:00:35.451967 +0000 UTC"
  },
  "spec" : {
    "hostNetwork" : {
      "subnetID" : "subnetID"
    },
    "containerNetwork" : [ {
      "subnetID" : "subnetID"
    } ],
    "publicBorderGroup" : "publicBorderGroup",
    "category" : "category"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowPartitionSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
```

```
        .withRegion(CceRegion.valueOf("<YOUR REGION>"))
        .build();
ShowPartitionRequest request = new ShowPartitionRequest();
request.withPartitionName("{partition_name}");
request.withClusterId("{cluster_id}");
try {
    ShowPartitionResponse response = client.showPartition(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.valueOf("<YOUR REGION>")) \
        .build()

    try:
        request = ShowPartitionRequest()
        request.partition_name = "{partition_name}"
        request.cluster_id = "{cluster_id}"
        response = client.show_partition(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowPartitionRequest{}
    request.PartitionName = "{partition_name}"
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowPartition(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.19 Updating a Partition

Function

This API is used to update a partition.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/partitions/{partition_name}

Table 4-240 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
partition_name	Yes	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-241 Request body parameters

Parameter	Mandatory	Type	Description
kind	No	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value Partition
apiVersion	No	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
metadata	No	metadata object	Definition Partition metadata Constraints N/A
spec	No	spec object	Definition Partition configurations Constraints N/A Range N/A Default Value N/A

Table 4-242 metadata

Parameter	Mandatory	Type	Description
name	No	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Table 4-243 spec

Parameter	Mandatory	Type	Description
hostNetwork	No	hostNetwork object	Definition Partition subnet Constraints N/A Range N/A Default Value N/A
containerNet work	No	Array of containerNet work objects	Definition Container subnet in a partition Constraints The maximum list length is 20. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
publicBorderGroup	No	String	Definition Group. In the CloudPond scenario, the value is the AZ ID. In the central region, the value is center. Constraints N/A Range N/A Default Value N/A
category	No	String	Definition AZ type Constraints N/A Range • Default: central AZ • IES: dedicated cloud AZ • HomeZone: Intelligent EdgeCloud AZ Default Value N/A

Table 4-244 hostNetwork

Parameter	Mandatory	Type	Description
subnetID	No	String	Definition Subnet ID. To obtain the value: • Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. • Method 2: Use the VPC APIs. For details, see Querying Subnets. Constraints N/A Range N/A Default Value N/A

Table 4-245 containerNetwork

Parameter	Mandatory	Type	Description
subnetID	No	String	Definition Subnet ID. To obtain the value: - Method 1: Log in to the VPC console and click the target subnet on the Subnets page. You can view the IPv4 subnet ID on the displayed page. - Method 2: Use the VPC APIs. For details, see Querying Subnets . Constraints N/A Range N/A Default Value N/A

Response Parameters

Status code: 200

Table 4-246 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Partition

Parameter	Type	Description
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	metadata object	Definition Partition metadata Constraints N/A
spec	spec object	Definition Partition configurations Constraints N/A Range N/A Default Value N/A

Table 4-247 metadata

Parameter	Type	Description
name	String	Definition Partition name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-248 spec

Parameter	Type	Description
hostNetwork	hostNetwork object	Definition Partition subnet Constraints N/A Range N/A Default Value N/A
containerNetwork	Array of containerNetwork objects	Definition Container subnet in a partition Constraints N/A Range N/A Default Value N/A
publicBorderGroup	String	Definition Group. In the CloudPond scenario, the value is the AZ ID. In the central region, the value is center . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
category	String	Definition AZ type Constraints N/A Range • Default: central AZ • IES: dedicated cloud AZ • HomeZone: Intelligent EdgeCloud AZ Default Value N/A

Table 4-249 hostNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Table 4-250 containerNetwork

Parameter	Type	Description
subnetID	String	Definition Subnet ID Constraints N/A Range N/A Default Value N/A

Example Requests

```
PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/partitions/{partition_name}

{
  "kind" : "Partition",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "partitionName"
  },
  "spec" : {
    "hostNetwork" : {
      "subnetID" : "subnetID"
    },
    "containerNetwork" : [ {
      "subnetID" : "subnetID"
    } ],
    "publicBorderGroup" : "publicBorderGroup",
    "category" : "category"
  }
}
```

Example Responses

Status code: 200

OK

```
{
  "kind" : "Partition",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "partitionName",
    "creationTimestamp" : "2000-1-1 00:00:35.451967 +0000 UTC"
  },
  "spec" : {
    "hostNetwork" : {
      "subnetID" : "subnetID"
    },
    "containerNetwork" : [ {
      "subnetID" : "subnetID"
    } ],
    "publicBorderGroup" : "publicBorderGroup",
    "category" : "category"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;
```

```
public class UpdatePartitionSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdatePartitionRequest request = new UpdatePartitionRequest();
        request.withPartitionName("{partition_name}");
        request.withClusterId("{cluster_id}");
        PartitionReqBody body = new PartitionReqBody();
        List<PartitionReqBodySpecContainerNetwork> listSpecContainerNetwork = new ArrayList<>();
        listSpecContainerNetwork.add(
            new PartitionReqBodySpecContainerNetwork()
                .withSubnetID("subnetID")
        );
        PartitionReqBodySpecHostNetwork hostNetworkSpec = new PartitionReqBodySpecHostNetwork();
        hostNetworkSpec.withSubnetID("subnetID");
        PartitionReqBodySpec specbody = new PartitionReqBodySpec();
        specbody.withHostNetwork(hostNetworkSpec)
            .withContainerNetwork(listSpecContainerNetwork)
            .withPublicBorderGroup("publicBorderGroup")
            .withCategory("category");
        PartitionReqBodyMetadata metadatabody = new PartitionReqBodyMetadata();
        metadatabody.withName("partitionName");
        body.withSpec(specbody);
        body.withMetadata(metadatabody);
        body.withApiVersion("v3");
        body.withKind("Partition");
        request.withBody(body);
        try {
            UpdatePartitionResponse response = client.updatePartition(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
```



```
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdatePartitionRequest()
        request.partition_name = "{partition_name}"
        request.cluster_id = "{cluster_id}"
        listContainerNetworkSpec = [
            PartitionReqBodySpecContainerNetwork(
                subnet_id="subnetID"
            )
        ]
        hostNetworkSpec = PartitionReqBodySpecHostNetwork(
            subnet_id="subnetID"
        )
        specbody = PartitionReqBodySpec(
            host_network=hostNetworkSpec,
            container_network=listContainerNetworkSpec,
            public_border_group="publicBorderGroup",
            category="category"
        )
        metadatabody = PartitionReqBodyMetadata(
            name="partitionName"
        )
        request.body = PartitionReqBody(
            spec=specbody,
            metadata=metadatabody,
            api_version="v3",
            kind="Partition"
        )
        response = client.update_partition(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdatePartitionRequest{}
request.PartitionName = "{partition_name}"
request.ClusterId = "{cluster_id}"
subnetIDContainerNetwork:= "subnetID"
var listContainerNetworkSpec = []model.PartitionReqBodySpecContainerNetwork{
    {
        SubnetID: &subnetIDContainerNetwork,
    },
}
subnetIDHostNetwork:= "subnetID"
hostNetworkSpec := &model.PartitionReqBodySpecHostNetwork{
    SubnetID: &subnetIDHostNetwork,
}
publicBorderGroupSpec:= "publicBorderGroup"
categorySpec:= "category"
specbody := &model.PartitionReqBodySpec{
    HostNetwork: hostNetworkSpec,
    ContainerNetwork: &listContainerNetworkSpec,
    PublicBorderGroup: &publicBorderGroupSpec,
    Category: &categorySpec,
}
nameMetadata:= "partitionName"
metadatabody := &model.PartitionReqBodyMetadata{
    Name: &nameMetadata,
}
apiVersionPartitionReqBody:= "v3"
kindPartitionReqBody:= "Partition"
request.Body = &model.PartitionReqBody{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: &apiVersionPartitionReqBody,
    Kind: &kindPartitionReqBody,
}
response, err := client.UpdatePartition(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.20 Obtaining the Global OBS Access Secret of a Cluster

Function

This API is used to obtain the global OBS access secret of a cluster.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/longaksk/config

Table 4-251 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-252 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-253 Response body parameters

Parameter	Type	Description
hasUploadedLong AKSK	Boolean	Definition: Whether the global OBS access secret is uploaded to the cluster. Constraints: N/A Options: false: The global OBS access secret is not uploaded. true: The global OBS access secret is uploaded. Default value: N/A

Parameter	Type	Description
enableLongAKSK	Boolean	Definition: Whether to enable the global OBS access secret. If this option is enabled, the paas.longaksk secret will be created in the kube-system namespace of the cluster. If this option is disabled, the secret will be deleted. Constraints: N/A Options: • false: The global OBS access secret is disabled. • true: The global OBS access secret is enabled. Default value: N/A

Example Requests

None

Example Responses

Status code: 200

The global OBS access secret has been obtained.

```
{  
  "hasUploadedLongAKSK" : true,  
  "enableLongAKSK" : true  
}
```

Status Codes

Status Code	Description
200	The global OBS access secret has been obtained.

Error Codes

See [Error Codes](#).

4.2.21 Updating the Global OBS Access Secret of a Cluster

Function

This API is used to update the global OBS access secret of a cluster.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/longaksk/config

Table 4-254 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-255 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-256 Request body parameters

Parameter	Mandatory	Type	Description
enableLongAKSK	Yes	Boolean	Definition: Whether to enable the global OBS access secret. If this option is enabled, the paas.longaksk secret will be created in the kube-system namespace of the cluster. If this option is disabled, the secret will be deleted. Constraints: N/A Options: • false: The global OBS access secret is disabled. • true: The global OBS access secret is enabled. Default value: N/A

Response Parameters

Status code: 204

The global OBS access secret has been updated.

None

Example Requests

```
PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/longaksk/config
{
  "enableLongAKSK" : true
}
```

Example Responses

None

Status Codes

Status Code	Description
204	The global OBS access secret has been updated.

Error Codes

See [Error Codes](#).

4.2.22 Obtaining the Global OBS Access Secret of a Project

Function

This API is used to obtain the global OBS access secret of a project.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/longaksk/config

Table 4-257 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Request Parameters

Table 4-258 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-259 Response body parameters

Parameter	Type	Description
hasUploadedLongAKSK	Boolean	Definition: Whether the global OBS access secret is uploaded for the project. Constraints: N/A Options: • false: The global OBS access secret is not uploaded. • true: The global OBS access secret is uploaded. Default value: N/A
enableLongAKSKInNewCluster	Boolean	Definition: Whether to enable the global OBS access secret for a new cluster. Constraints: N/A Options: • false: The global OBS access secret is not enabled for the new cluster. • true: The global OBS access secret is enabled for the new cluster. Default value: N/A

Example Requests

None

Example Responses

Status code: 200

The global OBS access secret has been obtained.

```
{
  "hasUploadedLongAKSK" : true,
  "enableLongAKSKInNewCluster" : true
}
```

Status Codes

Status Code	Description
200	The global OBS access secret has been obtained.

Error Codes

See [Error Codes](#).

4.2.23 Updating the Global OBS Access Secret of a Project

Function

This API is used to update the global OBS access secret of a project.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/longaksk/config

Table 4-260 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Request Parameters

Table 4-261 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-262 Request body parameters

Parameter	Mandatory	Type	Description
enableLongAKSKInNewCluster	Yes	Boolean	Definition: Whether to enable the global OBS access secret for a new cluster. Constraints: N/A Options: • false: The global OBS access secret is not enabled for the new cluster. • true: The global OBS access secret is enabled for the new cluster. Default value: N/A

Response Parameters

Status code: 204

The global OBS access secret has been updated.

None

Example Requests

```
PUT /api/v3/projects/{project_id}/longaksk/config
{
  "enableLongAKSKInNewCluster" : true
}
```

Example Responses

None

Status Codes

Status Code	Description
204	The global OBS access secret has been updated.

Error Codes

See [Error Codes](#).

4.2.24 Obtaining Available Cluster Flavors

Function

This API is used to obtain the available cluster flavors.

 NOTE

The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** specifies the resource path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v2/flavor/specifications

Table 4-263 Query Parameters

Parameter	Mandatory	Type	Description
clusterType	Yes	String	Definition Used to obtain available flavors by cluster architecture. Range ● VirtualMachine: CCE cluster ● ARM64: Kunpeng cluster

Request Parameters

Table 4-264 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-265 Response body parameters

Parameter	Type	Description
clusterFlavorSpecs	ClusterFlavorSpecification object	Definition Cluster flavors for sale

Table 4-266 ClusterFlavorSpecification

Parameter	Type	Description
name	String	Definition Flavor name Range • cce.s1.small: a small-scale CCE cluster with one control plane node and a maximum of 50 worker nodes • cce.s1.medium: a medium-scale CCE cluster with one control plane node and a maximum of 200 worker nodes • cce.s2.small: a small-scale CCE cluster with three control plane nodes and a maximum of 50 worker nodes • cce.s2.medium: a medium-scale CCE cluster with three control plane nodes and a maximum of 200 worker nodes • cce.s2.large: a large-scale CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes • cce.s2.xlarge: an ultra-large-scale CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes Special cluster flavors in DeC are as follows: - cce.dec.s1.small: a small-scale DeC CCE cluster with one control plane node and a maximum of 50 worker nodes) - cce.dec.s1.medium: a medium-scale DeC CCE cluster with one control plane node and a maximum of 200 worker nodes) - cce.dec.s1.large: a large-scale DeC CCE cluster with one control plane node and a maximum of 1,000 worker nodes) - cce.dec.s1.xlarge: an ultra-large-scale DeC CCE cluster with one control plane node and a maximum of 2,000 worker nodes)

Parameter	Type	Description
		- cce.dec.s2.small: a small-scale DeC CCE cluster with three control plane nodes and a maximum of 50 worker nodes) - cce.dec.s2.medium: a medium-scale DeC CCE cluster with three control plane nodes and a maximum of 200 worker nodes) - cce.dec.s2.large: a large-scale DeC CCE cluster with three control plane nodes and a maximum of 1,000 worker nodes) - cce.dec.s2.xlarge: an ultra-large-scale DeC CCE cluster with three control plane nodes and a maximum of 2,000 worker nodes)
nodeCapacity	Integer	Definition The number of nodes in a cluster Range • 50: A maximum of 50 nodes is supported. • 200: A maximum of 200 nodes is supported. • 1000: A maximum of 1,000 nodes is supported. • 2000: A maximum of 2,000 nodes is supported.
availableMaster-Flavors	Array of MasterFlavorSpec objects	Definition Control plane node details
isSoldOut	Boolean	Definition Whether the cluster flavors are sold out Range N/A
isSupportMultiAZ	Boolean	Definition Whether the control plane nodes in a cluster can be deployed in different AZs Range N/A

Table 4-267 MasterFlavorSpec

Parameter	Type	Description
name	String	Definition Control plane node flavors Range N/A
azs	Array of strings	Definition AZs supported by control plane nodes
azFaultDomains	Map<String,Array<String>>	Definition Fault domains supported by the AZs where control plane nodes reside

Example Requests

None

Example Responses

Status code: 200

The available cluster flavors have been obtained.

```
{
  "clusterFlavorSpecs": [ {
    "name": "cce.s2.large",
    "nodeCapacity": 1000,
    "availableMasterFlavors": [ {
      "name": "ac6.4xlarge.2",
      "azs": [ "az1", "az2" ],
      "azFaultDomains": {
        "az1": [ "FD1", "FD2" ]
      }
    }
  ]
}
```

Status Codes

Status Code	Description
200	The available cluster flavors have been obtained.

Error Codes

See [Error Codes](#).

4.2.25 Obtaining the Feature Gate Status

Function

This API is used to obtain the feature gate status.

NOTE

The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** specifies the resource path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3.1/feature-gates

Request Parameters

Table 4-268 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-269 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range Configuration Default Value N/A
apiVersion	String	Definition API version Constraints The value cannot be changed. Range v3.1 Default Value N/A

Parameter	Type	Description
spec	map<string, object>	Definition Feature gate details

Example Requests

None

Example Responses

Status code: 200

The feature gate status has been obtained.

```
{
  "kind" : "Configuration",
  "apiVersion" : "v3.1",
  "spec" : {
    "enableBilling" : true
  }
}
```

Status Codes

Status Code	Description
200	The feature gate status has been obtained.

Error Codes

See [Error Codes](#).

4.2.26 Obtaining the AZ List

Function

This API is used to obtain the AZ list.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v2/availabilityZones

Table 4-270 Query Parameters

Parameter	Mandatory	Type	Description
locale	No	String	Definition: AZ name display setting by region Range: • zh-cn: displays the Chinese name. • en-us: displays the English name, for example, AZ1.

Request Parameters

Table 4-271 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-272 Response body parameters

Parameter	Type	Description
[items]	Array of GetAvailableZoneResponseBody objects	Definition AZ list information Constraints N/A

Table 4-273 GetAvailableZoneResponseBody

Parameter	Type	Description
id	String	Definition: AZ ID. Options: N/A
name	String	Definition: AZ name. Options: N/A

Parameter	Type	Description
displayName	String	Definition: AZ name displayed in the region where the AZ is located. Options: N/A
azGroupIds	Array of strings	Definition: Groups that an AZ belongs to. An AZ can belong to multiple AZ groups. Options: N/A
PublicBorderGroup	String	Definition: Group that an EIP belongs to. In the CloudPond edge scenario, the value is the AZ ID. In the central region, the value is center . Options: N/A
category	String	Definition: AZ type. Options: • Default: central AZ • IES: dedicated cloud AZ • HomeZone: Intelligent EdgeCloud AZ
alias	String	Definition: AZ alias. Options: N/A

Example Requests

None

Example Responses

Status code: 200

The AZ list has been obtained.

```
[ {
  "id" : "cn-north-4a",
  "name" : "AZ1",
  "displayName" : "AZ1",
```

```
"azGroupIds" : [ "cn-north-4-all" ],  
"publicBorderGroup" : "center",  
"category" : "Default",  
"alias" : "AZ1"  
}]
```

Status Codes

Status Code	Description
200	The AZ list has been obtained.

Error Codes

See [Error Codes](#).

4.2.27 Obtaining Cluster Quotas

Function

This API is used to obtain the cluster quotas.

Calling Method

For details, see [Calling APIs](#).

URI

GET /cce/v1/projects/{project_id}/quota

Table 4-274 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Request Parameters

Table 4-275 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-276 Response body parameters

Parameter	Type	Description
quotas	quotas object	Definition Cluster quota Constraints N/A

Table 4-277 quotas

Parameter	Type	Description
resources	Array of ClusterQuotaResource objects	Definition Cluster quota Constraints N/A

Table 4-278 ClusterQuotaResource

Parameter	Type	Description
type	String	Definition Resource type Constraints N/A Range cluster: a standard or Turbo cluster autopilot_cluster: an Autopilot cluster Default Value N/A
quota	Integer	Definition Total quota Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
used	Integer	Definition Used resource quota Constraints N/A Range N/A Default Value N/A
unit	String	Definition Unit Constraints N/A Range N/A Default Value N/A
min	Integer	Definition Minimum quota Constraints N/A Range N/A Default Value N/A
max	Integer	Definition Maximum quota Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

The cluster quotas have been obtained.

```
{
  "quotas": {
    "resources": [ {
      "type": "type",
      "quota": 50,
      "used": 23,
      "min": 1,
      "max": 1000,
      "unit": "unit"
    } ]
  }
}
```

Status Codes

Status Code	Description
200	The cluster quotas have been obtained.

Error Codes

See [Error Codes](#).

4.3 Node Management

4.3.1 Creating a Node

Function

This API is used to create a node in a specified cluster.

NOTE

- If there is no cluster, [create one](#).
- The URL for cluster management is in the format of **https://Endpoint/uri**. In the URL, **uri** indicates the resource path, that is, the path for API access.

Constraints

Only KVM nodes can be created. Non-KVM nodes cannot be used after being created.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes

Table 4-279 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-280 Query Parameters

Parameter	Mandatory	Type	Description
nodepoolScaleUp	No	String	Definition Whether a node creation request is delivered for node pool scale-out. If the parameter is set to NodepoolScaleUp, CCE automatically adjusts the number of nodes to be created based on the CIDR block of the current cluster subnet. If the subnet supports only one user node, but the requested number of nodes exceeds that, CCE will create only one node. Constraints N/A Range • NodepoolScaleUp: Nodes are created for node pool scale-out. Default Value N/A

Request Parameters

Table 4-281 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-282 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Node
apiVersion	Yes	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	No	NodeMetadata object	Definition Metadata of a node, which is a collection of attributes Constraints N/A
spec	Yes	NodeSpec object	Definition Details about a node. CCE creates or updates objects by defining or updating spec . Constraints N/A

Table 4-283 NodeMetadata

Parameter	Mandatory	Type	Description
name	No	String	Definition Node name Constraints If the number of nodes (count) is greater than 1, some random characters are added to the end of the entered node name according to the default rule, which is <i>entered-name-random-characters</i>. If the length of the entered node name exceeds 50 characters, CCE uses the first 50 characters and adds random characters to the end of the node name. Range The node name must contain 1 to 56 characters that start with a lowercase letter and not end with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Default Value If name is not specified or is set to an empty string, the node name is generated based on the default rule, which is <i>cluster-name-random-characters</i>. If the cluster name is too long, only the first 36 characters are used.

Parameter	Mandatory	Type	Description
uid	No	String	Definition Node ID, which uniquely identifies a node Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
annotations	No	Map<String,String>	Definition CCE node annotations, in key-value pairs (not the native Kubernetes annotations) Example: <pre>"annotations": { "key1" : "value1", "key2" : "value2" }</pre> Constraints This parameter is only used for query and cannot be input through a request. Entered data of this parameter is invalid. Range Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. Default Value N/A

Parameter	Mandatory	Type	Description
creationTimes tamp	No	String	Definition Time when a node was created Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
updateTime stamp	No	String	Definition Time when a node was updated Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
ownerReferen ces	No	ownerRefere nces object	Definition Owner object Constraints • The value is automatically generated after the object is created. A user-defined value will not take effect. • This parameter is not present in the response of the API for creating a node. Range N/A Default Value N/A

Table 4-284 ownerReferences

Parameter	Mandatory	Type	Description
nodepoolName	No	String	Definition Node pool name Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
nodepoolID	No	String	Definition Node pool ID Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Table 4-285 NodeSpec

Parameter	Mandatory	Type	Description
flavor	Yes	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
az	Yes	String	Details: AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to a node created using APIs. For details about AZs supported by CCE, see Regions and Endpoints. Constraints: When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
os	No	String	Details: OS type of a node. For details about the supported OSs, see Node OSs . Constraints: • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Options: N/A Default value: N/A
login	No	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Yes	Volume object	Details: Disk information of a node Constraints: None

Parameter	Mandatory	Type	Description
dataVolumes	No	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume . Constraints: • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left blank only when a system disk is selected for the system components.

Parameter	Mandatory	Type	Description
storage	No	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	No	NodePublicIP object	Details: EIP of a node Constraints: This parameter is not supported when you add a node during node pool creation.

Parameter	Mandatory	Type	Description
nodeNicSpec	No	NodeNicSpec object	Details: ENI of a node Constraints: None
count	No	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	No	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Mandatory	Type	Description
taints	No	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Mandatory	Type	Description
waitPostInstallFinish	No	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Mandatory	Type	Description
k8sTags	No	Map<String,String>	Details: Defined in key-value pairs. • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints: A maximum of 20 key-value pairs are allowed.

Parameter	Mandatory	Type	Description
ecsGroupId	No	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A
dedicatedHostId	No	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
userTags	No	Array of UserTag objects	Details: Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints: • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	No	Runtime object	Details: Container runtime • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Mandatory	Type	Description
initializedConditions	No	Array of strings	Details: Custom initialization flag, which is left blank by default. Before CCE nodes are initialized, they are tainted with node.cloudprovider.kubernetes.io/uninitialized to prevent pods from being scheduled to them. When creating a node, you can specify the initializedConditions parameter to control the taint removal time. By default, the timeout interval is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two labels of types CCEInitial and CustomedInitial and setting the status to True, as shown in the following: 3. CCE polls status.Conditions of nodes to check whether there are conditions of types CCEInitial and CustomedInitial. If such conditions exist and the status is True, the node initialization is complete and the initialization taint is removed. 4. initializedConditions allows you to configure a timeout period during node creation. For example,

Parameter	Mandatory	Type	Description
			"initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout period is 15 minutes. After the timeout period is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 seconds. • Each value has 1 to 20 characters, which must consist of letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout period is minute (m).
extendParam	No	NodeExtendParam object	Details: Extended parameters for creating a node Constraints: None
hostnameConfig	No	HostnameConfig object	Details: Kubernetes node name configuration parameter Constraints: The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Mandatory	Type	Description
serverEnterpriseProjectID	No	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
partition	No	String	Details: Zone that a node belongs to. It can be the central cloud or an edge site. Constraints: This field can be specified only for CCE Turbo clusters that support distributed clouds. Options: • center: central cloud • AZ ID of the edge site Default value: N/A

Parameter	Mandatory	Type	Description
nodeNameTemplate	No	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product . Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Options: N/A Default value: N/A

Table 4-286 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-287 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-288 Volume

Parameter	Mandatory	Type	Description
size	Yes	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A
volumetype	Yes	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance . Default value: N/A

Parameter	Mandatory	Type	Description
extendParam	No	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	No	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A
cluster_type	No	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A

Parameter	Mandatory	Type	Description
hw:passthrough	No	Boolean	Details: • Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. • If the node pool type is ElasticBMS, this field must be set to true. • If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	No	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-289 VolumeMetadata

Parameter	Mandatory	Type	Description
<code>__system__encrypted</code>	No	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	No	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-290 Storage

Parameter	Mandatory	Type	Description
<code>storageSelectors</code>	Yes	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Mandatory	Type	Description
storageGroups	Yes	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-291 StorageSelectors

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
storageType	Yes	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	No	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-292 matchLabels

Parameter	Mandatory	Type	Description
size	No	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	No	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	No	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
metadataCmkid	No	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	No	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-293 StorageGroups

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: • If cceManaged is set to true, the name field must be set to vgpaas. • If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. • If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	No	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
selectorNames	Yes	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.
virtualSpaces	Yes	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-294 VirtualSpace

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	Yes	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Mandatory	Type	Description
lvmConfig	No	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	No	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-295 LVMConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
path	No	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-296 RuntimeConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-297 NodePublicIP

Parameter	Mandatory	Type	Description
ids	No	Array of strings	Details: IDs of existing EIPs Constraints: • The quantity cannot be greater than the number of nodes to be created. • If ids has been specified, you do not need to configure count and eip.
count	No	Integer	Details: Number of EIPs to be dynamically created Constraints: count and eip must be configured simultaneously. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
eip	No	NodeEIPSpec object	Details: EIP configuration Constraints: This parameter is not supported when you create a node pool.

Table 4-298 NodeEIPSpec

Parameter	Mandatory	Type	Description
iptype	Yes	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	No	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-299 NodeBandwidth

Parameter	Mandatory	Type	Description
chargemode	No	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Mandatory	Type	Description
size	No	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	No	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-300 NodeNicSpec

Parameter	Mandatory	Type	Description
primaryNic	No	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	No	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-301 NicSpec

Parameter	Mandatory	Type	Description
subnetId	No	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	No	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps . The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Mandatory	Type	Description
ipBlock	No	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	No	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-302 Taint

Parameter	Mandatory	Type	Description
key	Yes	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
value	No	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	Yes	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-303 UserTag

Parameter	Mandatory	Type	Description
key	No	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A
value	No	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, Default value: N/A

Table 4-304 Runtime

Parameter	Mandatory	Type	Description
name	No	String	Details: Container runtime Constraints: None Options: N/A Default value: - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-305 NodeExtendParam

Parameter	Mandatory	Type	Description
ecs:performance-type	No	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
orderId	No	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	No	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
maxPods	No	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: None Options: 16 to 256 Default value: N/A
periodType	No	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Mandatory	Type	Description
periodNum	No	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	No	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Mandatory	Type	Description
isAutoPay	No	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Mandatory	Type	Description
DockerLVMConfigOverride	No	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride":{"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Mandatory	Type	Description
<code>dockerBaseSize</code>	No	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Mandatory	Type	Description
containerBaseSize	No	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single

Parameter	Mandatory	Type	Description
			container is not limited by default.
publicKey	No	String	Definition Public key of a node, which is used for SSH key login Constraints • This parameter is invalid if you use a key pair to log in to the node. • This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/preInstall	No	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "*Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
alpha.cce/postInstall	No	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "*Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	No	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	No	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
marketType	No	String	Details: When creating a spot ECS, this parameter should be set to spot . Constraints: This parameter is valid only when billingMode is set to 0 . Options: N/A Default value: N/A
spotPrice	No	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
agency_name	No	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	No	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	No	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
init-node-password	No	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-306 HostnameConfig

Parameter	Mandatory	Type	Description
type	Yes	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-307 nodeNameTemplate

Parameter	Mandatory	Type	Description
nodeNamePrefix	No	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	No	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Response Parameters

Status code: 201

Table 4-308 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Node
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	NodeMetadata object	Definition Metadata of a node, which is a collection of attributes Constraints N/A
spec	NodeSpec object	Definition Details about a node. CCE creates or updates objects by defining or updating spec . Constraints N/A
status	NodeStatus object	Definition Node status, which is dynamically recorded. Specifying node status during creation or modification is unnecessary and has no effect. Constraints N/A

Table 4-309 NodeMetadata

Parameter	Type	Description
name	String	Definition Node name Constraints If the number of nodes (count) is greater than 1 , some random characters are added to the end of the entered node name according to the default rule, which is <i>entered-name-random-characters</i> . If the length of the entered node name exceeds 50 characters, CCE uses the first 50 characters and adds random characters to the end of the node name. Range The node name must contain 1 to 56 characters that start with a lowercase letter and not end with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Default Value If name is not specified or is set to an empty string, the node name is generated based on the default rule, which is <i>cluster-name-random-characters</i> . If the cluster name is too long, only the first 36 characters are used.
uid	String	Definition Node ID, which uniquely identifies a node Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
annotations	Map<String,String>	Definition CCE node annotations, in key-value pairs (not the native Kubernetes annotations) Example: <pre>"annotations": { "key1": "value1", "key2": "value2" }</pre> Constraints This parameter is only used for query and cannot be input through a request. Entered data of this parameter is invalid. Range Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. Default Value N/A
creationTimestamp	String	Definition Time when a node was created Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
updateTimestamp	String	Definition Time when a node was updated Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
ownerReferences	ownerReferences object	Definition Owner object Constraints - The value is automatically generated after the object is created. A user-defined value will not take effect. - This parameter is not present in the response of the API for creating a node. Range N/A Default Value N/A

Table 4-310 ownerReferences

Parameter	Type	Description
nodepoolName	String	Definition Node pool name Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
nodepoolID	String	Definition Node pool ID Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Table 4-311 NodeSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A
az	String	Details: AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to a node created using APIs. For details about AZs supported by CCE, see Regions and Endpoints . Constraints: When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random . Options: N/A Default value: N/A

Parameter	Type	Description
os	String	Details: OS type of a node. For details about the supported OSs, see Node OSs. Constraints: • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Options: N/A Default value: N/A
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Details: Disk information of a node Constraints: None

Parameter	Type	Description
dataVolumes	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume. Constraints: • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left blank only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodePublicIP object	Details: EIP of a node Constraints: This parameter is not supported when you add a node during node pool creation.
nodeNicSpec	NodeNicSpec object	Details: ENI of a node Constraints: None

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Details: Defined in key-value pairs. • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints: A maximum of 20 key-value pairs are allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Details: Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints: • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Details: Container runtime • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Type	Description
initializedConditions	Array of strings	Details: Custom initialization flag, which is left blank by default. Before CCE nodes are initialized, they are tainted with node.cloudprovider.kubernetes.io/uninitialized to prevent pods from being scheduled to them. When creating a node, you can specify the initializedConditions parameter to control the taint removal time. By default, the timeout interval is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two labels of types CCEInitial and CustomedInitial and setting the status to True, as shown in the following: 3. CCE polls status.Conditions of nodes to check whether there are conditions of types CCEInitial and CustomedInitial. If such conditions exist and the status is True, the node initialization is complete and the initialization taint is removed. 4. initializedConditions allows you to configure a timeout period during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout period is 15 minutes. After the timeout period is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial

Parameter	Type	Description
		status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 seconds. • Each value has 1 to 20 characters, which must consist of letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout period is minute (m).
extendParam	NodeExtendParam object	Details: Extended parameters for creating a node Constraints: None
hostnameConfig	HostnameConfig object	Details: Kubernetes node name configuration parameter Constraints: The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
partition	String	Details: Zone that a node belongs to. It can be the central cloud or an edge site. Constraints: This field can be specified only for CCE Turbo clusters that support distributed clouds. Options: • center: central cloud • AZ ID of the edge site Default value: N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product . Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Options: N/A Default value: N/A

Table 4-312 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-313 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }],./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-314 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-315 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with __system__encrypted . Constraints: None Options: N/A Default value: N/A

Table 4-316 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-317 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-318 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-319 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-320 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-321 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-322 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-323 NodePublicIP

Parameter	Type	Description
ids	Array of strings	Details: IDs of existing EIPs Constraints: - The quantity cannot be greater than the number of nodes to be created. - If ids has been specified, you do not need to configure count and eip.
count	Integer	Details: Number of EIPs to be dynamically created Constraints: count and eip must be configured simultaneously. Options: N/A Default value: N/A
eip	NodeEIPSpec object	Details: EIP configuration Constraints: This parameter is not supported when you create a node pool.

Table 4-324 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-325 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Type	Description
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-326 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-327 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-328 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-329 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-330 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-331 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <code>echo -n "**Content to be encoded*" base64</code> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/ NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-332 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-333 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A

Table 4-334 NodeStatus

Parameter	Type	Description
phase	String	Definition Node status Constraints This parameter is not present in the response of the API for creating a node. Range Node resource lifecycle management (such as installation and uninstallation) status and Kubernetes node status in a cluster. The options include: • Build: The node is being created. • Installing: The node is being managed. • Upgrading: The node is being upgraded. • Active: The node is running properly. • Abnormal: The node is abnormal. • Deleting: The node is being deleted. • Error: The node is faulty. Default Value N/A
lastProbeTime	String	Definition Last time when the node status was checked. If the cluster is in the abnormal, frozen, or intermediate state (for example, creating), the node status check may be affected. If checking a node's status takes longer than 5 minutes, the result is considered meaningless. Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
jobID	String	Definition Job ID during creation or deletion Range N/A
serverId	String	Definition ID of the underlying ECS or BMS node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIP	String	Definition Private IP address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIPv6IP	String	Definition Private IPv6 address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
publicIP	String	Definition Node EIP. If the ECS data is not synchronized in real time, you can synchronize the node data through the console. Constraints This parameter is not present in the response of the API for creating a node. Range N/A
deleteStatus	DeleteStatus object	Definition Resource status during resource deletion Constraints This parameter is returned only when a node is deleted. Range N/A
configurationUp- ToDate	Boolean	Definition Whether the node configuration is consistent with the latest configuration of the node template in the node pool to which the node belongs. When a node pool's OS or runtime is updated, the existing nodes may have a different OS or runtime, causing configurationUpToDate to be false . However, resetting the nodes will synchronize their OSs and runtimes with the node pool, setting configurationUpToDate to true . Constraints This parameter is not present in the response of the API for creating or updating a node. Range N/A

Table 4-335 DeleteStatus

Parameter	Type	Description
previous_total	Integer	Definition The total number of existing cluster resource records during node deletion Range N/A
current_total	Integer	Definition The latest number of resource records. This number is generated based on the current cluster resource records. Range N/A
updated	Integer	Definition The total number of to-be-deleted resource records updated during node deletion Range N/A
added	Integer	Definition The total number of to-be-deleted resource records added during node deletion Range N/A
deleted	Integer	Definition The total number of resource records deleted during node deletion Range N/A

Example Requests

- Create a yearly/monthly node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes
```

```
{
  "kind": "Node",
  "apiVersion": "v3",
  "metadata": {
    "name": "test-67235"
```

```
},
"spec": {
  "flavor": "c7.large.2",
  "az": "*****",
  "os": "EulerOS 2.5",
  "dataVolumes": [ {
    "size": 100,
    "volumetype": "SAS"
  } ],
  "billingMode": 1,
  "extendParam": {
    "maxPods": 110,
    "periodType": "month",
    "periodNum": 1,
    "isAutoPay": "false",
    "isAutoRenew": "false"
  },
  "nodeNicSpec": {
    "primaryNic": {
      "subnetId": "ca964acf-8468-4735-8229-97940ef6c881"
    }
  },
  "rootVolume": {
    "size": 50,
    "volumetype": "SAS"
  },
  "runtime": {
    "name": "docker"
  },
  "login": {
    "sshKey": "KeyPair-001"
  },
  "storage": {
    "storageSelectors": [ {
      "name": "cceUse",
      "storageType": "evs",
      "matchLabels": {
        "size": "100",
        "volumeType": "SAS",
        "count": "1"
      }
    } ],
    "storageGroups": [ {
      "name": "vgpaas",
      "selectorNames": [ "cceUse" ],
      "cceManaged": true,
      "virtualSpaces": [ {
        "name": "runtime",
        "size": "90%"
      }, {
        "name": "kubernetes",
        "size": "10%"
      } ]
    } ]
  },
  "count": 1
}
```

- Create a pay-per-use node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes

```
{
  "kind": "Node",
  "apiVersion": "v3",
  "metadata": {
```

```
"name" : "test-83790"
},
"spec" : {
  "flavor" : "c7.large.2",
  "az" : "*****",
  "os" : "EulerOS 2.5",
  "dataVolumes" : [ {
    "size" : 100,
    "volumetype" : "SAS"
  } ],
  "billingMode" : 0,
  "extendParam" : {
    "maxPods" : 110
  },
  "nodeNicSpec" : {
    "primaryNic" : {
      "subnetId" : "ca964acf-8468-4735-8229-97940ef6c881"
    }
  },
  "rootVolume" : {
    "size" : 50,
    "volumetype" : "SAS"
  },
  "runtime" : {
    "name" : "docker"
  },
  "login" : {
    "sshKey" : "KeyPair-001"
  },
  "storage" : {
    "storageSelectors" : [ {
      "name" : "cceUse",
      "storageType" : "evs",
      "matchLabels" : {
        "size" : "100",
        "volumeType" : "SAS",
        "count" : "1"
      }
    } ],
    "storageGroups" : [ {
      "name" : "vgpaas",
      "selectorNames" : [ "cceUse" ],
      "cceManaged" : true,
      "virtualSpaces" : [ {
        "name" : "runtime",
        "size" : "90%"
      }, {
        "name" : "kubernetes",
        "size" : "10%"
      } ]
    } ]
  },
  "count" : 1
}
}
```

- Create a pay-per-usenode with the flavor of 2 vCPUs and 4 GiB of memory. The node runs Huawei Cloud EulerOS 2.0 and uses Docker. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. The data disk uses the shared disk space mode.

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes

```
{
  "kind" : "Node",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "test-66909"
  },
  "spec" : {
```

```
"flavor" : "c7.large.2",
"az" : "*****",
"os" : "Huawei Cloud EulerOS 2.0",
"dataVolumes" : [ {
  "size" : 100,
  "volumetype" : "SAS"
} ],
"billingMode" : 0,
"extendParam" : {
  "maxPods" : 110
},
"nodeNicSpec" : {
  "primaryNic" : {
    "subnetId" : "ca964acf-8468-4735-8229-97940ef6c881"
  }
},
"rootVolume" : {
  "size" : 50,
  "volumetype" : "SAS"
},
"runtime" : {
  "name" : "docker"
},
"login" : {
  "sshKey" : "KeyPair-001"
},
"storage" : {
  "storageSelectors" : [ {
    "name" : "cceUse",
    "storageType" : "evs",
    "matchLabels" : {
      "size" : "100",
      "volumeType" : "SAS",
      "count" : "1"
    }
  }
],
"storageGroups" : [ {
  "name" : "vgpaas",
  "selectorNames" : [ "cceUse" ],
  "cceManaged" : true,
  "virtualSpaces" : [ {
    "name" : "share",
    "size" : "100%"
  } ]
} ]
},
"count" : 1
}
```

Example Responses

Status code: 201

The job for creating a node in a specified cluster is successfully delivered.

```
{
  "kind" : "Node",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "test-83790",
    "uid" : "5ecfddfe-87db-11ec-b5e5-0255ac101514",
    "annotations" : {
      "jobid" : "5ec1518c-87db-11ec-b5e5-0255ac101514",
      "resourceJobId" : "5ed0d692-87db-11ec-b5e5-0255ac101514"
    }
  },
  "spec" : {
    "flavor" : "c7.large.2",
```

```
"az" : "*****",
"os" : "EulerOS 2.5",
"login" : {
  "sshKey" : "KeyPair-001"
},
"rootVolume" : {
  "volumetype" : "SAS",
  "size" : 50
},
"dataVolumes" : [ {
  "volumetype" : "SAS",
  "size" : 100
} ],
"storage" : {
  "storageSelectors" : [ {
    "name" : "cceUse",
    "storageType" : "evs",
    "matchLabels" : {
      "count" : "1",
      "size" : "100",
      "volumeType" : "SAS"
    }
  } ],
  "storageGroups" : [ {
    "name" : "vgpaas",
    "cceManaged" : true,
    "selectorNames" : [ "cceUse" ],
    "virtualSpaces" : [ {
      "name" : "runtime",
      "size" : "90%"
    }, {
      "name" : "kubernetes",
      "size" : "10%"
    } ]
  } ]
},
"publicIP" : {
  "eip" : {
    "bandwidth" : { }
  }
},
"nodeNicSpec" : {
  "primaryNic" : {
    "subnetId" : "ca964acf-8468-4735-8229-97940ef6c881"
  }
},
"count" : 1,
"billingMode" : 0,
"runtime" : {
  "name" : "docker"
},
"extendParam" : {
  "chargingMode" : 0,
  "ecs:performancetype" : "computingv3",
  "init-node-password" : "*****",
  "maxPods" : 110,
  "publicKey" : ""
},
"status" : {
  "jobID" : "5ec1518c-87db-11ec-b5e5-0255ac101514"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Create a yearly/monthly node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;
import com.huaweicloud.sdk.cce.v3.model.Runtime;

import java.util.List;
import java.util.ArrayList;

public class CreateNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateNodeRequest request = new CreateNodeRequest();
        request.withClusterId("{cluster_id}");
        NodeCreateRequest body = new NodeCreateRequest();
        NodeExtendParam extendParamSpec = new NodeExtendParam();
        extendParamSpec.withMaxPods(110)
            .withPeriodType("month")
            .withPeriodNum(1)
            .withIsAutoRenew("false")
            .withIsAutoPay("false");
        Runtime runtimeSpec = new Runtime();
        runtimeSpec.withName(Runtime.NameEnum.fromValue("docker"));
        NicSpec primaryNicNodeNicSpec = new NicSpec();
        primaryNicNodeNicSpec.withSubnetId("ca964acf-8468-4735-8229-97940ef6c881");
        NodeNicSpec nodeNicSpecSpec = new NodeNicSpec();
        nodeNicSpecSpec.withPrimaryNic(primaryNicNodeNicSpec);
        List<VirtualSpace> listStorageGroupsVirtualSpaces = new ArrayList<>();
        listStorageGroupsVirtualSpaces.add(
            new VirtualSpace()
                .withName("runtime")
                .withSize("90%")
        );
        listStorageGroupsVirtualSpaces.add(
            new VirtualSpace()
                .withName("kubernetes")
                .withSize("10%")
        );
    }
}
```



```
);
List<String> listStorageGroupsSelectorNames = new ArrayList<>();
listStorageGroupsSelectorNames.add("cceUse");
List<StorageGroups> listStorageStorageGroups = new ArrayList<>();
listStorageStorageGroups.add(
    new StorageGroups()
        .WithName("vgpaas")
        .withCceManaged(true)
        .withSelectorNames(listStorageGroupsSelectorNames)
        .withVirtualSpaces(listStorageGroupsVirtualSpaces)
);
StorageSelectorsMatchLabels matchLabelsStorageSelectors = new
StorageSelectorsMatchLabels();
matchLabelsStorageSelectors.withSize("100")
    .withVolumeType("SAS")
    .withCount("1");
List<StorageSelectors> listStorageStorageSelectors = new ArrayList<>();
listStorageStorageSelectors.add(
    new StorageSelectors()
        .WithName("cceUse")
        .withStorageType("evs")
        .withMatchLabels(matchLabelsStorageSelectors)
);
Storage storageSpec = new Storage();
storageSpec.withStorageSelectors(listStorageStorageSelectors)
    .withStorageGroups(listStorageStorageGroups);
List<Volume> listSpecDataVolumes = new ArrayList<>();
listSpecDataVolumes.add(
    new Volume()
        .withSize(100)
        .withVolumetype("SAS")
);
Volume rootVolumeSpec = new Volume();
rootVolumeSpec.withSize(50)
    .withVolumetype("SAS");
Login loginSpec = new Login();
loginSpec.withSshKey("KeyPair-001");
NodeSpec specbody = new NodeSpec();
specbody.withFlavor("c7.large.2")
    .withAz("*****")
    .withOs("EulerOS 2.5")
    .withLogin(loginSpec)
    .withRootVolume(rootVolumeSpec)
    .withDataVolumes(listSpecDataVolumes)
    .withStorage(storageSpec)
    .withNodeNicSpec(nodeNicSpecSpec)
    .withCount(1)
    .withBillingMode(1)
    .withRuntime(runtimeSpec)
    .withExtendParam(extendParamSpec);
NodeMetadata metadatabody = new NodeMetadata();
metadatabody.withName("test-67235");
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("Node");
request.withBody(body);
try {
    CreateNodeResponse response = client.createNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
}
```

```
        System.out.println(e.getErrMsg());
    }
}
}
```

- Create a pay-per-use node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;
import com.huaweicloud.sdk.cce.v3.model.Runtime;

import java.util.List;
import java.util.ArrayList;

public class CreateNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateNodeRequest request = new CreateNodeRequest();
        request.withClusterId("{cluster_id}");
        NodeCreateRequest body = new NodeCreateRequest();
        NodeExtendParam extendParamSpec = new NodeExtendParam();
        extendParamSpec.withMaxPods(110);
        Runtime runtimeSpec = new Runtime();
        runtimeSpec.withName(Runtime.NameEnum.fromValue("docker"));
        NicSpec primaryNicNodeNicSpec = new NicSpec();
        primaryNicNodeNicSpec.withSubnetId("ca964acf-8468-4735-8229-97940ef6c881");
        NodeNicSpec nodeNicSpecSpec = new NodeNicSpec();
        nodeNicSpecSpec.withPrimaryNic(primaryNicNodeNicSpec);
        List<VirtualSpace> listStorageGroupsVirtualSpaces = new ArrayList<>();
        listStorageGroupsVirtualSpaces.add(
            new VirtualSpace()
                .withName("runtime")
                .withSize("90%")
        );
        listStorageGroupsVirtualSpaces.add(
            new VirtualSpace()
                .withName("kubernetes")
                .withSize("10%")
        );
        List<String> listStorageGroupsSelectorNames = new ArrayList<>();
```

```
listStorageGroupsSelectorNames.add("cceUse");
List<StorageGroups> listStorageStorageGroups = new ArrayList<>();
listStorageStorageGroups.add(
    new StorageGroups()
        .WithName("vgpaas")
        .withCceManaged(true)
        .withSelectorNames(listStorageGroupsSelectorNames)
        .withVirtualSpaces(listStorageGroupsVirtualSpaces)
);
StorageSelectorsMatchLabels matchLabelsStorageSelectors = new
StorageSelectorsMatchLabels();
matchLabelsStorageSelectors.withSize("100")
    .withVolumeType("SAS")
    .withCount("1");
List<StorageSelectors> listStorageStorageSelectors = new ArrayList<>();
listStorageStorageSelectors.add(
    new StorageSelectors()
        .WithName("cceUse")
        .withStorageType("evs")
        .withMatchLabels(matchLabelsStorageSelectors)
);
Storage storageSpec = new Storage();
storageSpec.withStorageSelectors(listStorageStorageSelectors)
    .withStorageGroups(listStorageStorageGroups);
List<Volume> listSpecDataVolumes = new ArrayList<>();
listSpecDataVolumes.add(
    new Volume()
        .withSize(100)
        .withVolumetype("SAS")
);
Volume rootVolumeSpec = new Volume();
rootVolumeSpec.withSize(50)
    .withVolumetype("SAS");
Login loginSpec = new Login();
loginSpec.withSshKey("KeyPair-001");
NodeSpec specbody = new NodeSpec();
specbody.withFlavor("c7.large.2")
    .withAz("*****")
    .withOs("EulerOS 2.5")
    .withLogin(loginSpec)
    .withRootVolume(rootVolumeSpec)
    .withDataVolumes(listSpecDataVolumes)
    .withStorage(storageSpec)
    .withNodeNicSpec(nodeNicSpecSpec)
    .withCount(1)
    .withBillingMode(0)
    .withRuntime(runtimeSpec)
    .withExtendParam(extendParamSpec);
NodeMetadata metadatabody = new NodeMetadata();
metadatabody.withName("test-83790");
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("Node");
request.withBody(body);
try {
    CreateNodeResponse response = client.createNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
```

- Create a pay-per-usenode with the flavor of 2 vCPUs and 4 GiB of memory. The node runs Huawei Cloud EulerOS 2.0 and uses Docker. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. The data disk uses the shared disk space mode.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;
import com.huaweicloud.sdk.cce.v3.model.Runtime;

import java.util.List;
import java.util.ArrayList;

public class CreateNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateNodeRequest request = new CreateNodeRequest();
        request.withClusterId("{cluster_id}");
        NodeCreateRequest body = new NodeCreateRequest();
        NodeExtendParam extendParamSpec = new NodeExtendParam();
        extendParamSpec.withMaxPods(110);
        Runtime runtimeSpec = new Runtime();
        runtimeSpec.withName(Runtime.NameEnum.fromValue("docker"));
        NicSpec primaryNicNodeNicSpec = new NicSpec();
        primaryNicNodeNicSpec.withSubnetId("ca964acf-8468-4735-8229-97940ef6c881");
        NodeNicSpec nodeNicSpecSpec = new NodeNicSpec();
        nodeNicSpecSpec.withPrimaryNic(primaryNicNodeNicSpec);
        List<VirtualSpace> listStorageGroupsVirtualSpaces = new ArrayList<>();
        listStorageGroupsVirtualSpaces.add(
            new VirtualSpace()
                .withName("share")
                .withSize("100%")
        );
        List<String> listStorageGroupsSelectorNames = new ArrayList<>();
        listStorageGroupsSelectorNames.add("cceUse");
        List<StorageGroups> listStorageStorageGroups = new ArrayList<>();
        listStorageStorageGroups.add(
            new StorageGroups()
                .withName("vgpaas")
                .withCceManaged(true)
                .withSelectorNames(listStorageGroupsSelectorNames)
        );
    }
}
```

```
        .withVirtualSpaces(listStorageGroupsVirtualSpaces)
    );
    StorageSelectorsMatchLabels matchLabelsStorageSelectors = new
StorageSelectorsMatchLabels();
    matchLabelsStorageSelectors.withSize("100")
        .withVolumeType("SAS")
        .withCount("1");
    List<StorageSelectors> listStorageStorageSelectors = new ArrayList<>();
    listStorageStorageSelectors.add(
        new StorageSelectors()
            .withName("cceUse")
            .withStorageType("evs")
            .withMatchLabels(matchLabelsStorageSelectors)
    );
    Storage storageSpec = new Storage();
    storageSpec.withStorageSelectors(listStorageStorageSelectors)
        .withStorageGroups(listStorageStorageGroups);
    List<Volume> listSpecDataVolumes = new ArrayList<>();
    listSpecDataVolumes.add(
        new Volume()
            .withSize(100)
            .withVolumetype("SAS")
    );
    Volume rootVolumeSpec = new Volume();
    rootVolumeSpec.withSize(50)
        .withVolumetype("SAS");
    Login loginSpec = new Login();
    loginSpec.withSshKey("KeyPair-001");
    NodeSpec specbody = new NodeSpec();
    specbody.withFlavor("c7.large.2")
        .withAz("*****")
        .withOs("Huawei Cloud EulerOS 2.0")
        .withLogin(loginSpec)
        .withRootVolume(rootVolumeSpec)
        .withDataVolumes(listSpecDataVolumes)
        .withStorage(storageSpec)
        .withNodeNicSpec(nodeNicSpecSpec)
        .withCount(1)
        .withBillingMode(0)
        .withRuntime(runtimeSpec)
        .withExtendParam(extendParamSpec);
    NodeMetadata metadatabody = new NodeMetadata();
    metadatabody.withName("test-66909");
    body.withSpec(specbody);
    body.withMetadata(metadatabody);
    body.withApiVersion("v3");
    body.withKind("Node");
    request.withBody(body);
    try {
        CreateNodeResponse response = client.createNode(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
    }
}
```

Python

- Create a yearly/monthly node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the

system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateNodeRequest()
        request.cluster_id = "{cluster_id}"
        extendParamSpec = NodeExtendParam(
            max_pods=110,
            period_type="month",
            period_num=1,
            is_auto_renew="false",
            is_auto_pay="false"
        )
        runtimeSpec = Runtime(
            name="docker"
        )
        primaryNicNodeNicSpec = NicSpec(
            subnet_id="ca964acf-8468-4735-8229-97940ef6c881"
        )
        nodeNicSpecSpec = NodeNicSpec(
            primary_nic=primaryNicNodeNicSpec
        )
        listVirtualSpacesStorageGroups = [
            VirtualSpace(
                name="runtime",
                size="90%"
            ),
            VirtualSpace(
                name="kubernetes",
                size="10%"
            )
        ]
        listSelectorNamesStorageGroups = [
            "cceUse"
        ]
        listStorageGroupsStorage = [
            StorageGroups(
                name="vgpaas",
                cce_managed=True,
                selector_names=listSelectorNamesStorageGroups,
                virtual_spaces=listVirtualSpacesStorageGroups
            )
        ]
        matchLabelsStorageSelectors = StorageSelectorsMatchLabels(
```

```
        size="100",
        volume_type="SAS",
        count="1"
    )
    listStorageSelectorsStorage = [
        StorageSelectors(
            name="cceUse",
            storage_type="evs",
            match_labels=matchLabelsStorageSelectors
        )
    ]
    storageSpec = Storage(
        storage_selectors=listStorageSelectorsStorage,
        storage_groups=listStorageGroupsStorage
    )
    listDataVolumesSpec = [
        Volume(
            size=100,
            volumetype="SAS"
        )
    ]
    rootVolumeSpec = Volume(
        size=50,
        volumetype="SAS"
    )
    loginSpec = Login(
        ssh_key="KeyPair-001"
    )
    specbody = NodeSpec(
        flavor="c7.large.2",
        az="*****",
        os="EulerOS 2.5",
        login=loginSpec,
        root_volume=rootVolumeSpec,
        data_volumes=listDataVolumesSpec,
        storage=storageSpec,
        node_nic_spec=nodeNicSpecSpec,
        count=1,
        billing_mode=1,
        runtime=runtimeSpec,
        extend_param=extendParamSpec
    )
    metadatabody = NodeMetadata(
        name="test-67235"
    )
    request.body = NodeCreateRequest(
        spec=specbody,
        metadata=metadatabody,
        api_version="v3",
        kind="Node"
    )
    response = client.create_node(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Create a pay-per-use node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

coding: utf-8

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
```

```
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateNodeRequest()
        request.cluster_id = "{cluster_id}"
        extendParamSpec = NodeExtendParam(
            max_pods=110
        )
        runtimeSpec = Runtime(
            name="docker"
        )
        primaryNicNodeNicSpec = NicSpec(
            subnet_id="ca964acf-8468-4735-8229-97940ef6c881"
        )
        nodeNicSpecSpec = NodeNicSpec(
            primary_nic=primaryNicNodeNicSpec
        )
        listVirtualSpacesStorageGroups = [
            VirtualSpace(
                name="runtime",
                size="90%"
            ),
            VirtualSpace(
                name="kubernetes",
                size="10%"
            )
        ]
        listSelectorNamesStorageGroups = [
            "cceUse"
        ]
        listStorageGroupsStorage = [
            StorageGroups(
                name="vgpaas",
                cce_managed=True,
                selector_names=listSelectorNamesStorageGroups,
                virtual_spaces=listVirtualSpacesStorageGroups
            )
        ]
        matchLabelsStorageSelectors = StorageSelectorsMatchLabels(
            size="100",
            volume_type="SAS",
            count="1"
        )
        listStorageSelectorsStorage = [
            StorageSelectors(
                name="cceUse",
                storage_type="evs",
                match_labels=matchLabelsStorageSelectors
            )
        ]
        storageSpec = Storage(
```



```
        storage_selectors=listStorageSelectorsStorage,
        storage_groups=listStorageGroupsStorage
    )
    listDataVolumesSpec = [
        Volume(
            size=100,
            volumetype="SAS"
        )
    ]
    rootVolumeSpec = Volume(
        size=50,
        volumetype="SAS"
    )
    loginSpec = Login(
        ssh_key="KeyPair-001"
    )
    specbody = NodeSpec(
        flavor="c7.large.2",
        az="*****",
        os="EulerOS 2.5",
        login=loginSpec,
        root_volume=rootVolumeSpec,
        data_volumes=listDataVolumesSpec,
        storage=storageSpec,
        node_nic_spec=nodeNicSpecSpec,
        count=1,
        billing_mode=0,
        runtime=runtimeSpec,
        extend_param=extendParamSpec
    )
    metadatabody = NodeMetadata(
        name="test-83790"
    )
    request.body = NodeCreateRequest(
        spec=specbody,
        metadata=metadatabody,
        api_version="v3",
        kind="Node"
    )
    response = client.create_node(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Create a pay-per-usenode with the flavor of 2 vCPUs and 4 GiB of memory. The node runs Huawei Cloud EulerOS 2.0 and uses Docker. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. The data disk uses the shared disk space mode.

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```

```
if __name__ == "__main__":
```

```
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```

```
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
```

```
credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateNodeRequest()
    request.cluster_id = "{cluster_id}"
    extendParamSpec = NodeExtendParam(
        max_pods=110
    )
    runtimeSpec = Runtime(
        name="docker"
    )
    primaryNicNodeNicSpec = NicSpec(
        subnet_id="ca964acf-8468-4735-8229-97940ef6c881"
    )
    nodeNicSpecSpec = NodeNicSpec(
        primary_nic=primaryNicNodeNicSpec
    )
    listVirtualSpacesStorageGroups = [
        VirtualSpace(
            name="share",
            size="100%"
        )
    ]
    listSelectorNamesStorageGroups = [
        "cceUse"
    ]
    listStorageGroupsStorage = [
        StorageGroups(
            name="vgpaas",
            cce_managed=True,
            selector_names=listSelectorNamesStorageGroups,
            virtual_spaces=listVirtualSpacesStorageGroups
        )
    ]
    matchLabelsStorageSelectors = StorageSelectorsMatchLabels(
        size="100",
        volume_type="SAS",
        count="1"
    )
    listStorageSelectorsStorage = [
        StorageSelectors(
            name="cceUse",
            storage_type="evs",
            match_labels=matchLabelsStorageSelectors
        )
    ]
    storageSpec = Storage(
        storage_selectors=listStorageSelectorsStorage,
        storage_groups=listStorageGroupsStorage
    )
    listDataVolumesSpec = [
        Volume(
            size=100,
            volumetype="SAS"
        )
    ]
    rootVolumeSpec = Volume(
        size=50,
        volumetype="SAS"
    )
    loginSpec = Login(
        ssh_key="KeyPair-001"
    )
}
```

```
specbody = NodeSpec(  
    flavor="c7.large.2",  
    az="*****",  
    os="Huawei Cloud EulerOS 2.0",  
    login=loginSpec,  
    root_volume=rootVolumeSpec,  
    data_volumes=listDataVolumesSpec,  
    storage=storageSpec,  
    node_nic_spec=nodeNicSpecSpec,  
    count=1,  
    billing_mode=0,  
    runtime=runtimeSpec,  
    extend_param=extendParamSpec  
)  
metadatabody = NodeMetadata(  
    name="test-66909"  
)  
request.body = NodeCreateRequest(  
    spec=specbody,  
    metadata=metadatabody,  
    api_version="v3",  
    kind="Node"  
)  
response = client.create_node(request)  
print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

- Create a yearly/monthly node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    // environment variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before  
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    // environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := cce.NewCceClient(  
        cce.CceClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).
```

```
Build()

request := &model.CreateNodeRequest{
    request.ClusterId = "{cluster_id}"
    maxPodsExtendParam:= int32(110)
    periodTypeExtendParam:= "month"
    periodNumExtendParam:= int32(1)
    isAutoRenewExtendParam:= "false"
    isAutoPayExtendParam:= "false"
    extendParamSpec := &model.NodeExtendParam{
        MaxPods: &maxPodsExtendParam,
        PeriodType: &periodTypeExtendParam,
        PeriodNum: &periodNumExtendParam,
        IsAutoRenew: &isAutoRenewExtendParam,
        IsAutoPay: &isAutoPayExtendParam,
    }
    nameRuntime:= model.GetRuntimeNameEnum().DOCKER
    runtimeSpec := &model.Runtime{
        Name: &nameRuntime,
    }
    subnetIdPrimaryNic:= "ca964acf-8468-4735-8229-97940ef6c881"
    primaryNicNodeNicSpec := &model.NicSpec{
        SubnetId: &subnetIdPrimaryNic,
    }
    nodeNicSpecSpec := &model.NodeNicSpec{
        PrimaryNic: primaryNicNodeNicSpec,
    }
}
var listVirtualSpacesStorageGroups = []model.VirtualSpace{
    {
        Name: "runtime",
        Size: "90%",
    },
    {
        Name: "kubernetes",
        Size: "10%",
    },
}
var listSelectorNamesStorageGroups = List<String>{
    "cceUse",
}
cceManagedStorageGroups:= true
var listStorageGroupsStorage = []model.StorageGroups{
    {
        Name: "vgpaas",
        CceManaged: &cceManagedStorageGroups,
        SelectorNames: listSelectorNamesStorageGroups,
        VirtualSpaces: listVirtualSpacesStorageGroups,
    },
}
sizeMatchLabels:= "100"
volumeTypeMatchLabels:= "SAS"
countMatchLabels:= "1"
matchLabelsStorageSelectors := &model.StorageSelectorsMatchLabels{
    Size: &sizeMatchLabels,
    VolumeType: &volumeTypeMatchLabels,
    Count: &countMatchLabels,
}
var listStorageSelectorsStorage = []model.StorageSelectors{
    {
        Name: "cceUse",
        StorageType: "evs",
        MatchLabels: matchLabelsStorageSelectors,
    },
}
storageSpec := &model.Storage{
    StorageSelectors: listStorageSelectorsStorage,
    StorageGroups: listStorageGroupsStorage,
}
var listDataVolumesSpec = []model.Volume{
```

```
{
    Size: int32(100),
    Volumetype: "SAS",
},
}
rootVolumeSpec := &model.Volume{
    Size: int32(50),
    Volumetype: "SAS",
}
sshKeyLogin:= "KeyPair-001"
loginSpec := &model.Login{
    SshKey: &sshKeyLogin,
}
osSpec:= "EulerOS 2.5"
countSpec:= int32(1)
billingModeSpec:= int32(1)
specbody := &model.NodeSpec{
    Flavor: "c7.large.2",
    Az: "*****",
    Os: &osSpec,
    Login: loginSpec,
    RootVolume: rootVolumeSpec,
    DataVolumes: listDataVolumesSpec,
    Storage: storageSpec,
    NodeNicSpec: nodeNicSpecSpec,
    Count: &countSpec,
    BillingMode: &billingModeSpec,
    Runtime: runtimeSpec,
    ExtendParam: extendParamSpec,
}
nameMetadata:= "test-67235"
metadatabody := &model.NodeMetadata{
    Name: &nameMetadata,
}
request.Body = &model.NodeCreateRequest{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Node",
}
response, err := client.CreateNode(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Create a pay-per-use node with the flavor of 2 vCPUs and 4 GiB of memory. The node runs EulerOS 2.5 for Docker containers. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    environment
```

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateNodeRequest{}
request.ClusterId = "{cluster_id}"
maxPodsExtendParam := int32(110)
extendParamSpec := &model.NodeExtendParam{
    MaxPods: &maxPodsExtendParam,
}
nameRuntime := model.GetRuntimeNameEnum().DOCKER
runtimeSpec := &model.Runtime{
    Name: &nameRuntime,
}
subnetIdPrimaryNic := "ca964acf-8468-4735-8229-97940ef6c881"
primaryNicNodeNicSpec := &model.NicSpec{
    SubnetId: &subnetIdPrimaryNic,
}
nodeNicSpecSpec := &model.NodeNicSpec{
    PrimaryNic: primaryNicNodeNicSpec,
}
var listVirtualSpacesStorageGroups = []model.VirtualSpace{
    {
        Name: "runtime",
        Size: "90%",
    },
    {
        Name: "kubernetes",
        Size: "10%",
    },
}
var listSelectorNamesStorageGroups = []string{
    "cceUse",
}
cceManagedStorageGroups := true
var listStorageGroupsStorage = []model.StorageGroups{
    {
        Name: "vgpaas",
        CceManaged: &cceManagedStorageGroups,
        SelectorNames: listSelectorNamesStorageGroups,
        VirtualSpaces: listVirtualSpacesStorageGroups,
    },
}
sizeMatchLabels := "100"
volumeTypeMatchLabels := "SAS"
countMatchLabels := "1"
matchLabelsStorageSelectors := &model.StorageSelectorsMatchLabels{
    Size: &sizeMatchLabels,
    VolumeType: &volumeTypeMatchLabels,
    Count: &countMatchLabels,
}
var listStorageSelectorsStorage = []model.StorageSelectors{
    {
        Name: "cceUse",
        StorageType: "evs",
        MatchLabels: matchLabelsStorageSelectors,
    },
}
```

```
}
storageSpec := &model.Storage{
    StorageSelectors: listStorageSelectorsStorage,
    StorageGroups: listStorageGroupsStorage,
}
var listDataVolumesSpec = []model.Volume{
    {
        Size: int32(100),
        Volumetype: "SAS",
    },
}
rootVolumeSpec := &model.Volume{
    Size: int32(50),
    Volumetype: "SAS",
}
sshKeyLogin:= "KeyPair-001"
loginSpec := &model.Login{
    SshKey: &sshKeyLogin,
}
osSpec:= "EulerOS 2.5"
countSpec:= int32(1)
billingModeSpec:= int32(0)
specbody := &model.NodeSpec{
    Flavor: "c7.large.2",
    Az: "*****",
    Os: &osSpec,
    Login: loginSpec,
    RootVolume: rootVolumeSpec,
    DataVolumes: listDataVolumesSpec,
    Storage: storageSpec,
    NodeNicSpec: nodeNicSpecSpec,
    Count: &countSpec,
    BillingMode: &billingModeSpec,
    Runtime: runtimeSpec,
    ExtendParam: extendParamSpec,
}
nameMetadata:= "test-83790"
metadatabody := &model.NodeMetadata{
    Name: &nameMetadata,
}
request.Body = &model.NodeCreateRequest{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Node",
}
response, err := client.CreateNode(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Create a pay-per-usenode with the flavor of 2 vCPUs and 4 GiB of memory. The node runs Huawei Cloud EulerOS 2.0 and uses Docker. The sizes of the system disk and data disk of the node are 50 GB and 100 GB, respectively. The data disk uses the shared disk space mode.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
```

// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateNodeRequest{
    request.ClusterId = "{cluster_id}"
    maxPodsExtendParam:= int32(110)
    extendParamSpec := &model.NodeExtendParam{
        MaxPods: &maxPodsExtendParam,
    }
    nameRuntime:= model.GetRuntimeNameEnum().DOCKER
    runtimeSpec := &model.Runtime{
        Name: &nameRuntime,
    }
    subnetIdPrimaryNic:= "ca964acf-8468-4735-8229-97940ef6c881"
    primaryNicNodeNicSpec := &model.NicSpec{
        SubnetId: &subnetIdPrimaryNic,
    }
    nodeNicSpecSpec := &model.NodeNicSpec{
        PrimaryNic: primaryNicNodeNicSpec,
    }
}
var listVirtualSpacesStorageGroups = []model.VirtualSpace{
    {
        Name: "share",
        Size: "100%",
    },
}
var listSelectorNamesStorageGroups = []string{
    "cceUse",
}
cceManagedStorageGroups:= true
var listStorageGroupsStorage = []model.StorageGroups{
    {
        Name: "vgpaas",
        CceManaged: &cceManagedStorageGroups,
        SelectorNames: listSelectorNamesStorageGroups,
        VirtualSpaces: listVirtualSpacesStorageGroups,
    },
}
sizeMatchLabels:= "100"
volumeTypeMatchLabels:= "SAS"
countMatchLabels:= "1"
matchLabelsStorageSelectors := &model.StorageSelectorsMatchLabels{
    Size: &sizeMatchLabels,
    VolumeType: &volumeTypeMatchLabels,
    Count: &countMatchLabels,
}
var listStorageSelectorsStorage = []model.StorageSelectors{
    {
        Name: "cceUse",
        StorageType: "evs",
    },
}
```



```
        MatchLabels: matchLabelsStorageSelectors,
    },
}
storageSpec := &model.Storage{
    StorageSelectors: listStorageSelectorsStorage,
    StorageGroups: listStorageGroupsStorage,
}
var listDataVolumesSpec = []model.Volume{
    {
        Size: int32(100),
        Volumetype: "SAS",
    },
}
rootVolumeSpec := &model.Volume{
    Size: int32(50),
    Volumetype: "SAS",
}
sshKeyLogin:= "KeyPair-001"
loginSpec := &model.Login{
    SshKey: &sshKeyLogin,
}
osSpec:= "Huawei Cloud EulerOS 2.0"
countSpec:= int32(1)
billingModeSpec:= int32(0)
specbody := &model.NodeSpec{
    Flavor: "c7.large.2",
    Az: "*****",
    Os: &osSpec,
    Login: loginSpec,
    RootVolume: rootVolumeSpec,
    DataVolumes: listDataVolumesSpec,
    Storage: storageSpec,
    NodeNicSpec: nodeNicSpecSpec,
    Count: &countSpec,
    BillingMode: &billingModeSpec,
    Runtime: runtimeSpec,
    ExtendParam: extendParamSpec,
}
nameMetadata:= "test-66909"
metadatabody := &model.NodeMetadata{
    Name: &nameMetadata,
}
request.Body = &model.NodeCreateRequest{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Node",
}
response, err := client.CreateNode(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	The job for creating a node in a specified cluster is successfully delivered.

Error Codes

See [Error Codes](#).

4.3.2 Reading a Specified Node

Function

This API is used to obtain details about a specified node via the node ID.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/{node_id}

Table 4-336 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
node_id	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-337 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-338 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Node
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	NodeMetadata object	Definition Metadata of a node, which is a collection of attributes Constraints N/A
spec	NodeSpec object	Definition Details about a node. CCE creates or updates objects by defining or updating spec . Constraints N/A
status	NodeStatus object	Definition Node status, which is dynamically recorded. Specifying node status during creation or modification is unnecessary and has no effect. Constraints N/A

Table 4-339 NodeMetadata

Parameter	Type	Description
name	String	Definition Node name Constraints If the number of nodes (count) is greater than 1 , some random characters are added to the end of the entered node name according to the default rule, which is <i>entered-name-random-characters</i> . If the length of the entered node name exceeds 50 characters, CCE uses the first 50 characters and adds random characters to the end of the node name. Range The node name must contain 1 to 56 characters that start with a lowercase letter and not end with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Default Value If name is not specified or is set to an empty string, the node name is generated based on the default rule, which is <i>cluster-name-random-characters</i> . If the cluster name is too long, only the first 36 characters are used.
uid	String	Definition Node ID, which uniquely identifies a node Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
annotations	Map<String,String>	Definition CCE node annotations, in key-value pairs (not the native Kubernetes annotations) Example: <pre>"annotations": { "key1": "value1", "key2": "value2" }</pre> Constraints This parameter is only used for query and cannot be input through a request. Entered data of this parameter is invalid. Range Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. Default Value N/A
creationTimestamp	String	Definition Time when a node was created Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
updateTimestamp	String	Definition Time when a node was updated Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
ownerReferences	ownerReferences object	Definition Owner object Constraints - The value is automatically generated after the object is created. A user-defined value will not take effect. - This parameter is not present in the response of the API for creating a node. Range N/A Default Value N/A

Table 4-340 ownerReferences

Parameter	Type	Description
nodepoolName	String	Definition Node pool name Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
nodepoolID	String	Definition Node pool ID Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Table 4-341 NodeSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A
az	String	Details: AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to a node created using APIs. For details about AZs supported by CCE, see Regions and Endpoints . Constraints: When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random . Options: N/A Default value: N/A

Parameter	Type	Description
os	String	Details: OS type of a node. For details about the supported OSs, see Node OSs . Constraints: • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Options: N/A Default value: N/A
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Details: Disk information of a node Constraints: None

Parameter	Type	Description
dataVolumes	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume. Constraints: • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left blank only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodePublicIP object	Details: EIP of a node Constraints: This parameter is not supported when you add a node during node pool creation.
nodeNicSpec	NodeNicSpec object	Details: ENI of a node Constraints: None

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Details: Defined in key-value pairs. • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints: A maximum of 20 key-value pairs are allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Details: Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints: The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Details: Container runtime - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Type	Description
initializedConditions	Array of strings	Details: Custom initialization flag, which is left blank by default. Before CCE nodes are initialized, they are tainted with node.cloudprovider.kubernetes.io/uninitialized to prevent pods from being scheduled to them. When creating a node, you can specify the initializedConditions parameter to control the taint removal time. By default, the timeout interval is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two labels of types CCEInitial and CustomedInitial and setting the status to True, as shown in the following: 3. CCE polls status.Conditions of nodes to check whether there are conditions of types CCEInitial and CustomedInitial. If such conditions exist and the status is True, the node initialization is complete and the initialization taint is removed. 4. initializedConditions allows you to configure a timeout period during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout period is 15 minutes. After the timeout period is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial

Parameter	Type	Description
		status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 seconds. • Each value has 1 to 20 characters, which must consist of letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout period is minute (m).
extendParam	NodeExtendParam object	Details: Extended parameters for creating a node Constraints: None
hostnameConfig	HostnameConfig object	Details: Kubernetes node name configuration parameter Constraints: The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
partition	String	Details: Zone that a node belongs to. It can be the central cloud or an edge site. Constraints: This field can be specified only for CCE Turbo clusters that support distributed clouds. Options: • center: central cloud • AZ ID of the edge site Default value: N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product . Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Options: N/A Default value: N/A

Table 4-342 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-343 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-344 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-345 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-346 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-347 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-348 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-349 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-350 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-351 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-352 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-353 NodePublicIP

Parameter	Type	Description
ids	Array of strings	Details: IDs of existing EIPs Constraints: - The quantity cannot be greater than the number of nodes to be created. - If ids has been specified, you do not need to configure count and eip.
count	Integer	Details: Number of EIPs to be dynamically created Constraints: count and eip must be configured simultaneously. Options: N/A Default value: N/A
eip	NodeEIPSpec object	Details: EIP configuration Constraints: This parameter is not supported when you create a node pool.

Table 4-354 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-355 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Type	Description
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-356 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-357 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-358 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-359 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-360 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-361 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <code>echo -n "**Content to be encoded*" base64</code> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/ NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-362 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-363 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Table 4-364 NodeStatus

Parameter	Type	Description
phase	String	Definition Node status Constraints This parameter is not present in the response of the API for creating a node. Range Node resource lifecycle management (such as installation and uninstallation) status and Kubernetes node status in a cluster. The options include: • Build: The node is being created. • Installing: The node is being managed. • Upgrading: The node is being upgraded. • Active: The node is running properly. • Abnormal: The node is abnormal. • Deleting: The node is being deleted. • Error: The node is faulty. Default Value N/A
lastProbeTime	String	Definition Last time when the node status was checked. If the cluster is in the abnormal, frozen, or intermediate state (for example, creating), the node status check may be affected. If checking a node's status takes longer than 5 minutes, the result is considered meaningless. Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
jobID	String	Definition Job ID during creation or deletion Range N/A
serverId	String	Definition ID of the underlying ECS or BMS node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIP	String	Definition Private IP address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIPv6IP	String	Definition Private IPv6 address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
publicIP	String	Definition Node EIP. If the ECS data is not synchronized in real time, you can synchronize the node data through the console. Constraints This parameter is not present in the response of the API for creating a node. Range N/A
deleteStatus	DeleteStatus object	Definition Resource status during resource deletion Constraints This parameter is returned only when a node is deleted. Range N/A
configurationUp- ToDate	Boolean	Definition Whether the node configuration is consistent with the latest configuration of the node template in the node pool to which the node belongs. When a node pool's OS or runtime is updated, the existing nodes may have a different OS or runtime, causing configurationUpToDate to be false . However, resetting the nodes will synchronize their OSs and runtimes with the node pool, setting configurationUpToDate to true . Constraints This parameter is not present in the response of the API for creating or updating a node. Range N/A

Table 4-365 DeleteStatus

Parameter	Type	Description
previous_total	Integer	Definition The total number of existing cluster resource records during node deletion Range N/A
current_total	Integer	Definition The latest number of resource records. This number is generated based on the current cluster resource records. Range N/A
updated	Integer	Definition The total number of to-be-deleted resource records updated during node deletion Range N/A
added	Integer	Definition The total number of to-be-deleted resource records added during node deletion Range N/A
deleted	Integer	Definition The total number of resource records deleted during node deletion Range N/A

Example Requests

None

Example Responses

Status code: 200

Information about the specified node is successfully obtained.

```
{
  "kind" : "Node",
  "apiVersion" : "v3",
```

```
"metadata" : {
  "name" : "myhost",
  "uid" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
  "creationTimestamp" : "2018-08-02 08:12:40.124294439 +0000 UTC",
  "updateTimestamp" : "2018-08-02 08:18:20.221871842 +0000 UTC",
  "annotations" : {
    "kubernetes.io/node-pool.id" : "az1.dc1#s1.medium#EulerOS 2.2"
  }
},
"spec" : {
  "flavor" : "s1.medium",
  "az" : "*****1",
  "os" : "EulerOS 2.2",
  "login" : {
    "sshKey" : "KeyPair-001"
  },
  "rootVolume" : {
    "volumetype" : "SAS",
    "size" : 40
  },
  "dataVolumes" : [ {
    "volumetype" : "SAS",
    "size" : 100
  } ],
  "publicIP" : {
    "eip" : {
      "bandwidth" : { }
    }
  },
  "billingMode" : 0
},
"status" : {
  "phase" : "Active",
  "serverId" : "456789abc-9368-46f3-8f29-d1a95622a568",
  "publicIP" : "10.34.56.78",
  "privateIP" : "192.168.1.23"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
```

```
ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ShowNodeRequest request = new ShowNodeRequest();
request.withClusterId("{cluster_id}");
request.withNodeId("{node_id}");
try {
    ShowNodeResponse response = client.showNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowNodeRequest()
        request.cluster_id = "{cluster_id}"
        request.node_id = "{node_id}"
        response = client.show_node(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowNodeRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodeId = "{node_id}"
    response, err := client.ShowNode(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Information about the specified node is successfully obtained.

Error Codes

See [Error Codes](#).

4.3.3 Listing All Nodes in a Cluster

Function

This API is used to obtain details about all nodes in a specified cluster with the cluster ID.

NOTE

The URL for cluster management is in the format of `https://Endpoint/uri`. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

GET `/api/v3/projects/{project_id}/clusters/{cluster_id}/nodes`

Table 4-366 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-367 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-368 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value List
apiVersion	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
items	Array of Node objects	Definition List of nodes, which include all nodes in the current cluster. You can find the corresponding nodes based on the value of items.metadata.name . Constraints N/A Range N/A Default Value N/A

Table 4-369 Node

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Node
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	NodeMetadata object	Definition Metadata of a node, which is a collection of attributes Constraints N/A
spec	NodeSpec object	Definition Details about a node. CCE creates or updates objects by defining or updating spec . Constraints N/A
status	NodeStatus object	Definition Node status, which is dynamically recorded. Specifying node status during creation or modification is unnecessary and has no effect. Constraints N/A

Table 4-370 NodeMetadata

Parameter	Type	Description
name	String	Definition Node name Constraints If the number of nodes (count) is greater than 1, some random characters are added to the end of the entered node name according to the default rule, which is <i>entered-name-random-characters</i>. If the length of the entered node name exceeds 50 characters, CCE uses the first 50 characters and adds random characters to the end of the node name. Range The node name must contain 1 to 56 characters that start with a lowercase letter and not end with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Default Value If name is not specified or is set to an empty string, the node name is generated based on the default rule, which is <i>cluster-name-random-characters</i>. If the cluster name is too long, only the first 36 characters are used.
uid	String	Definition Node ID, which uniquely identifies a node Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
annotations	Map<String,String>	Definition CCE node annotations, in key-value pairs (not the native Kubernetes annotations) Example: <pre>"annotations": { "key1": "value1", "key2": "value2" }</pre> Constraints This parameter is only used for query and cannot be input through a request. Entered data of this parameter is invalid. Range Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. Default Value N/A
creationTimestamp	String	Definition Time when a node was created Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
updateTimestamp	String	Definition Time when a node was updated Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
ownerReferences	ownerReferences object	Definition Owner object Constraints - The value is automatically generated after the object is created. A user-defined value will not take effect. - This parameter is not present in the response of the API for creating a node. Range N/A Default Value N/A

Table 4-371 ownerReferences

Parameter	Type	Description
nodepoolName	String	Definition Node pool name Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
nodepoolID	String	Definition Node pool ID Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Table 4-372 NodeSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A
az	String	Details: AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to a node created using APIs. For details about AZs supported by CCE, see Regions and Endpoints . Constraints: When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random . Options: N/A Default value: N/A

Parameter	Type	Description
os	String	Details: OS type of a node. For details about the supported OSs, see Node OSs . Constraints: • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Options: N/A Default value: N/A
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Details: Disk information of a node Constraints: None

Parameter	Type	Description
dataVolumes	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume. Constraints: • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left blank only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodePublicIP object	Details: EIP of a node Constraints: This parameter is not supported when you add a node during node pool creation.
nodeNicSpec	NodeNicSpec object	Details: ENI of a node Constraints: None

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Details: Defined in key-value pairs. • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints: A maximum of 20 key-value pairs are allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Details: Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints: • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Details: Container runtime • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Type	Description
initializedConditions	Array of strings	Details: Custom initialization flag, which is left blank by default. Before CCE nodes are initialized, they are tainted with node.cloudprovider.kubernetes.io/uninitialized to prevent pods from being scheduled to them. When creating a node, you can specify the initializedConditions parameter to control the taint removal time. By default, the timeout interval is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two labels of types CCEInitial and CustomedInitial and setting the status to True, as shown in the following: 3. CCE polls status.Conditions of nodes to check whether there are conditions of types CCEInitial and CustomedInitial. If such conditions exist and the status is True, the node initialization is complete and the initialization taint is removed. 4. initializedConditions allows you to configure a timeout period during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout period is 15 minutes. After the timeout period is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial

Parameter	Type	Description
		status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 seconds. • Each value has 1 to 20 characters, which must consist of letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout period is minute (m).
extendParam	NodeExtendParam object	Details: Extended parameters for creating a node Constraints: None
hostnameConfig	HostnameConfig object	Details: Kubernetes node name configuration parameter Constraints: The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
partition	String	Details: Zone that a node belongs to. It can be the central cloud or an edge site. Constraints: This field can be specified only for CCE Turbo clusters that support distributed clouds. Options: • center: central cloud • AZ ID of the edge site Default value: N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product . Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Options: N/A Default value: N/A

Table 4-373 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-374 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }],./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-375 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-376 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-377 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-378 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-379 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-380 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-381 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-382 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-383 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-384 NodePublicIP

Parameter	Type	Description
ids	Array of strings	Details: IDs of existing EIPs Constraints: - The quantity cannot be greater than the number of nodes to be created. - If ids has been specified, you do not need to configure count and eip.
count	Integer	Details: Number of EIPs to be dynamically created Constraints: count and eip must be configured simultaneously. Options: N/A Default value: N/A
eip	NodeEIPSpec object	Details: EIP configuration Constraints: This parameter is not supported when you create a node pool.

Table 4-385 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-386 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE - Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. - Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: - If this field is not specified, the billing is based on bandwidth. - If the field value is empty, the billing is based on bandwidth. - If the field value is traffic, the billing is based on traffic. - If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Type	Description
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-387 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-388 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-389 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-390 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-391 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-392 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-393 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-394 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A

Table 4-395 NodeStatus

Parameter	Type	Description
phase	String	Definition Node status Constraints This parameter is not present in the response of the API for creating a node. Range Node resource lifecycle management (such as installation and uninstallation) status and Kubernetes node status in a cluster. The options include: • Build: The node is being created. • Installing: The node is being managed. • Upgrading: The node is being upgraded. • Active: The node is running properly. • Abnormal: The node is abnormal. • Deleting: The node is being deleted. • Error: The node is faulty. Default Value N/A
lastProbeTime	String	Definition Last time when the node status was checked. If the cluster is in the abnormal, frozen, or intermediate state (for example, creating), the node status check may be affected. If checking a node's status takes longer than 5 minutes, the result is considered meaningless. Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
jobID	String	Definition Job ID during creation or deletion Range N/A
serverId	String	Definition ID of the underlying ECS or BMS node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIP	String	Definition Private IP address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIPv6IP	String	Definition Private IPv6 address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
publicIP	String	Definition Node EIP. If the ECS data is not synchronized in real time, you can synchronize the node data through the console. Constraints This parameter is not present in the response of the API for creating a node. Range N/A
deleteStatus	DeleteStatus object	Definition Resource status during resource deletion Constraints This parameter is returned only when a node is deleted. Range N/A
configurationUp- ToDate	Boolean	Definition Whether the node configuration is consistent with the latest configuration of the node template in the node pool to which the node belongs. When a node pool's OS or runtime is updated, the existing nodes may have a different OS or runtime, causing configurationUpToDate to be false . However, resetting the nodes will synchronize their OSs and runtimes with the node pool, setting configurationUpToDate to true . Constraints This parameter is not present in the response of the API for creating or updating a node. Range N/A

Table 4-396 DeleteStatus

Parameter	Type	Description
previous_total	Integer	Definition The total number of existing cluster resource records during node deletion Range N/A
current_total	Integer	Definition The latest number of resource records. This number is generated based on the current cluster resource records. Range N/A
updated	Integer	Definition The total number of to-be-deleted resource records updated during node deletion Range N/A
added	Integer	Definition The total number of to-be-deleted resource records added during node deletion Range N/A
deleted	Integer	Definition The total number of resource records deleted during node deletion Range N/A

Example Requests

None

Example Responses

Status code: 200

Information about all the nodes in the specified cluster is successfully obtained.

```
{  
  "kind" : "List",  
  "apiVersion" : "v3",  
}
```



```
"items" : [ {
  "kind" : "Node",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "myhost",
    "uid" : "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
    "creationTimestamp" : "2018-08-02 07:37:24.005071325 +0000 UTC",
    "updateTimestamp" : "2018-08-02 07:44:04.965500815 +0000 UTC",
    "annotations" : {
      "kubernetes.io/node-pool.id" : "az1.dc1#s1.medium#EulerOS 2.2"
    }
  },
  "spec" : {
    "flavor" : "s1.medium",
    "az" : "az1.dc1",
    "os" : "EulerOS 2.2",
    "login" : {
      "sshKey" : "KeyPair-001"
    },
    "rootVolume" : {
      "volumetype" : "SAS",
      "size" : 40
    },
    "dataVolumes" : [ {
      "volumetype" : "SAS",
      "size" : 100
    } ],
    "publicIP" : {
      "eip" : {
        "bandwidth" : { }
      }
    },
    "billingMode" : 0
  },
  "status" : {
    "phase" : "Active",
    "serverId" : "456789abc-9368-46f3-8f29-d1a95622a568",
    "publicIP" : "10.34.56.78",
    "privateIP" : "192.168.1.23"
  }
} ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListNodesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
```

this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ListNodesRequest request = new ListNodesRequest();
request.withClusterId("{cluster_id}");
try {
    ListNodesResponse response = client.listNodes(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListNodesRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_nodes(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListNodesRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListNodes(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Information about all the nodes in the specified cluster is successfully obtained.

Error Codes

See [Error Codes](#).

4.3.4 Updating a Specified Node

Function

This API is used to update information about a specified node.

NOTE

- Currently, only the **name** field in **metadata** can be updated. This field indicates the node name.
- The URL for cluster management is in the format of **https://Endpoint/uri**. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/{node_id}

Table 4-397 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
node_id	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-398 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-399 Request body parameters

Parameter	Mandatory	Type	Description
metadata	Yes	ClusterNodeInformationMetadata object	Definition Metadata of a node, which is a collection of attributes Constraints N/A

Table 4-400 ClusterNodeInformationMetadata

Parameter	Mandatory	Type	Description
name	Yes	String	Definition Node name Constraints The node name must contain 1 to 56 characters that start with a lowercase letter and not end with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range N/A Default Value N/A NOTE After the node name is changed, the ECS name (VM name) will be changed accordingly.

Response Parameters

Status code: 200

Table 4-401 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Node
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
metadata	NodeMetadata object	Definition Metadata of a node, which is a collection of attributes Constraints N/A
spec	NodeSpec object	Definition Details about a node. CCE creates or updates objects by defining or updating spec . Constraints N/A
status	NodeStatus object	Definition Node status, which is dynamically recorded. Specifying node status during creation or modification is unnecessary and has no effect. Constraints N/A

Table 4-402 NodeMetadata

Parameter	Type	Description
name	String	Definition Node name Constraints If the number of nodes (count) is greater than 1, some random characters are added to the end of the entered node name according to the default rule, which is <i>entered-name-random-characters</i>. If the length of the entered node name exceeds 50 characters, CCE uses the first 50 characters and adds random characters to the end of the node name. Range The node name must contain 1 to 56 characters that start with a lowercase letter and not end with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Default Value If name is not specified or is set to an empty string, the node name is generated based on the default rule, which is <i>cluster-name-random-characters</i>. If the cluster name is too long, only the first 36 characters are used.
uid	String	Definition Node ID, which uniquely identifies a node Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
annotations	Map<String,String>	Definition CCE node annotations, in key-value pairs (not the native Kubernetes annotations) Example: <pre>"annotations": { "key1": "value1", "key2": "value2" }</pre> Constraints This parameter is only used for query and cannot be input through a request. Entered data of this parameter is invalid. Range Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. Default Value N/A
creationTimestamp	String	Definition Time when a node was created Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
updateTimestamp	String	Definition Time when a node was updated Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
ownerReferences	ownerReferences object	Definition Owner object Constraints - The value is automatically generated after the object is created. A user-defined value will not take effect. - This parameter is not present in the response of the API for creating a node. Range N/A Default Value N/A

Table 4-403 ownerReferences

Parameter	Type	Description
nodepoolName	String	Definition Node pool name Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
nodepoolID	String	Definition Node pool ID Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Table 4-404 NodeSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A
az	String	Details: AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to a node created using APIs. For details about AZs supported by CCE, see Regions and Endpoints . Constraints: When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random . Options: N/A Default value: N/A

Parameter	Type	Description
os	String	Details: OS type of a node. For details about the supported OSs, see Node OSs . Constraints: • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Options: N/A Default value: N/A
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Details: Disk information of a node Constraints: None

Parameter	Type	Description
dataVolumes	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume. Constraints: • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left blank only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodePublicIP object	Details: EIP of a node Constraints: This parameter is not supported when you add a node during node pool creation.
nodeNicSpec	NodeNicSpec object	Details: ENI of a node Constraints: None

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Details: Defined in key-value pairs. • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints: A maximum of 20 key-value pairs are allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Details: Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints: The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Details: Container runtime - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Type	Description
initializedConditions	Array of strings	Details: Custom initialization flag, which is left blank by default. Before CCE nodes are initialized, they are tainted with node.cloudprovider.kubernetes.io/uninitialized to prevent pods from being scheduled to them. When creating a node, you can specify the initializedConditions parameter to control the taint removal time. By default, the timeout interval is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two labels of types CCEInitial and CustomedInitial and setting the status to True, as shown in the following: 3. CCE polls status.Conditions of nodes to check whether there are conditions of types CCEInitial and CustomedInitial. If such conditions exist and the status is True, the node initialization is complete and the initialization taint is removed. 4. initializedConditions allows you to configure a timeout period during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout period is 15 minutes. After the timeout period is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial

Parameter	Type	Description
		status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 seconds. • Each value has 1 to 20 characters, which must consist of letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout period is minute (m).
extendParam	NodeExtendParam object	Details: Extended parameters for creating a node Constraints: None
hostnameConfig	HostnameConfig object	Details: Kubernetes node name configuration parameter Constraints: The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
partition	String	Details: Zone that a node belongs to. It can be the central cloud or an edge site. Constraints: This field can be specified only for CCE Turbo clusters that support distributed clouds. Options: • center: central cloud • AZ ID of the edge site Default value: N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product . Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Options: N/A Default value: N/A

Table 4-405 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-406 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }],./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-407 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-408 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-409 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-410 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-411 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-412 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-413 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-414 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-415 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-416 NodePublicIP

Parameter	Type	Description
ids	Array of strings	Details: IDs of existing EIPs Constraints: - The quantity cannot be greater than the number of nodes to be created. - If ids has been specified, you do not need to configure count and eip.
count	Integer	Details: Number of EIPs to be dynamically created Constraints: count and eip must be configured simultaneously. Options: N/A Default value: N/A
eip	NodeEIPSpec object	Details: EIP configuration Constraints: This parameter is not supported when you create a node pool.

Table 4-417 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-418 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Type	Description
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-419 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-420 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-421 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-422 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-423 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-424 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot . Constraints: This parameter is valid only when billingMode is set to 0 . Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-425 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-426 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A

Table 4-427 NodeStatus

Parameter	Type	Description
phase	String	Definition Node status Constraints This parameter is not present in the response of the API for creating a node. Range Node resource lifecycle management (such as installation and uninstallation) status and Kubernetes node status in a cluster. The options include: • Build: The node is being created. • Installing: The node is being managed. • Upgrading: The node is being upgraded. • Active: The node is running properly. • Abnormal: The node is abnormal. • Deleting: The node is being deleted. • Error: The node is faulty. Default Value N/A
lastProbeTime	String	Definition Last time when the node status was checked. If the cluster is in the abnormal, frozen, or intermediate state (for example, creating), the node status check may be affected. If checking a node's status takes longer than 5 minutes, the result is considered meaningless. Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
jobID	String	Definition Job ID during creation or deletion Range N/A
serverId	String	Definition ID of the underlying ECS or BMS node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIP	String	Definition Private IP address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIPv6IP	String	Definition Private IPv6 address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
publicIP	String	Definition Node EIP. If the ECS data is not synchronized in real time, you can synchronize the node data through the console. Constraints This parameter is not present in the response of the API for creating a node. Range N/A
deleteStatus	DeleteStatus object	Definition Resource status during resource deletion Constraints This parameter is returned only when a node is deleted. Range N/A
configurationUp- ToDate	Boolean	Definition Whether the node configuration is consistent with the latest configuration of the node template in the node pool to which the node belongs. When a node pool's OS or runtime is updated, the existing nodes may have a different OS or runtime, causing configurationUpToDate to be false . However, resetting the nodes will synchronize their OSs and runtimes with the node pool, setting configurationUpToDate to true . Constraints This parameter is not present in the response of the API for creating or updating a node. Range N/A

Table 4-428 DeleteStatus

Parameter	Type	Description
previous_total	Integer	Definition The total number of existing cluster resource records during node deletion Range N/A
current_total	Integer	Definition The latest number of resource records. This number is generated based on the current cluster resource records. Range N/A
updated	Integer	Definition The total number of to-be-deleted resource records updated during node deletion Range N/A
added	Integer	Definition The total number of to-be-deleted resource records added during node deletion Range N/A
deleted	Integer	Definition The total number of resource records deleted during node deletion Range N/A

Example Requests

Update the name of a specified node.

```
{
  "metadata": {
    "name": "new-hostname"
  }
}
```

Example Responses

Status code: 200

Information about the specified node is successfully updated.

```
{
  "kind": "Node",
  "apiVersion": "v3",
  "metadata": {
    "name": "new-hostname",
    "uid": "4d1ecb2c-229a-11e8-9c75-0255ac100ceb",
    "creationTimestamp": "2017-08-20T21:11:09Z",
    "updateTimestamp": "2017-08-20T21:11:09Z",
    "annotations": {
      "kubernetes.io/node-pool.id": "az1.dc1#s1.medium#EulerOS 2.2"
    }
  },
  "spec": {
    "flavor": "s1.medium",
    "az": "az1.dc1",
    "os": "EulerOS 2.2",
    "login": {
      "sshKey": "KeyPair-001"
    }
  },
  "rootVolume": {
    "volumetype": "SAS",
    "size": 40
  },
  "dataVolumes": [ {
    "volumetype": "SAS",
    "size": 100
  } ],
  "publicIP": {
    "eip": { }
  },
  "billingMode": 0
},
"status": {
  "phase": "Active",
  "serverId": "456789abc-9368-46f3-8f29-d1a95622a568",
  "publicIP": "10.34.56.78",
  "privateIP": "192.168.1.23"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Update the name of a specified node.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpdateNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
```

```
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();

UpdateNodeRequest request = new UpdateNodeRequest();
request.withClusterId("{cluster_id}");
request.withNodeId("{node_id}");
ClusterNodeInformation body = new ClusterNodeInformation();
ClusterNodeInformationMetadata metadatabody = new ClusterNodeInformationMetadata();
metadatabody.withName("new-hostname");
body.withMetadata(metadatabody);
request.withBody(body);
try {
    UpdateNodeResponse response = client.updateNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Update the name of a specified node.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()
```



```
try:
    request = UpdateNodeRequest()
    request.cluster_id = "{cluster_id}"
    request.node_id = "{node_id}"
    metadatabody = ClusterNodeInformationMetadata(
        name="new-hostname"
    )
    request.body = ClusterNodeInformation(
        metadata=metadatabody
    )
    response = client.update_node(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Update the name of a specified node.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateNodeRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodeId = "{node_id}"
    metadatabody := &model.ClusterNodeInformationMetadata{
        Name: "new-hostname",
    }
    request.Body = &model.ClusterNodeInformation{
        Metadata: metadatabody,
    }
    response, err := client.UpdateNode(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Information about the specified node is successfully updated.

Error Codes

See [Error Codes](#).

4.3.5 Deleting a Node

Function

This API is used to delete a specified node.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/{node_id}

Table 4-429 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
node_id	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Table 4-430 Query Parameters

Parameter	Mandatory	Type	Description
nodepoolScaleDown	No	String	Definition Whether the request is delivered by a node pool. If the parameter is not set to NoScaleDown , the number of nodes in the corresponding node pool will be automatically updated. Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-431 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-432 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value Node
apiVersion	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3

Parameter	Type	Description
metadata	NodeMetadata object	Definition Metadata of a node, which is a collection of attributes Constraints N/A
spec	NodeSpec object	Definition Details about a node. CCE creates or updates objects by defining or updating spec . Constraints N/A
status	NodeStatus object	Definition Node status, which is dynamically recorded. Specifying node status during creation or modification is unnecessary and has no effect. Constraints N/A

Table 4-433 NodeMetadata

Parameter	Type	Description
name	String	Definition Node name Constraints If the number of nodes (count) is greater than 1 , some random characters are added to the end of the entered node name according to the default rule, which is <i>entered-name-random-characters</i> . If the length of the entered node name exceeds 50 characters, CCE uses the first 50 characters and adds random characters to the end of the node name. Range The node name must contain 1 to 56 characters that start with a lowercase letter and not end with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Default Value If name is not specified or is set to an empty string, the node name is generated based on the default rule, which is <i>cluster-name-random-characters</i> . If the cluster name is too long, only the first 36 characters are used.
uid	String	Definition Node ID, which uniquely identifies a node Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
annotations	Map<String,String>	Definition CCE node annotations, in key-value pairs (not the native Kubernetes annotations) Example: <pre>"annotations": { "key1": "value1", "key2": "value2" }</pre> Constraints This parameter is only used for query and cannot be input through a request. Entered data of this parameter is invalid. Range Annotations are not used to identify or select objects. The metadata in annotations may be small or large, structured or unstructured, and may include characters that are not allowed in labels. Default Value N/A
creationTimestamp	String	Definition Time when a node was created Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
updateTimestamp	String	Definition Time when a node was updated Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Parameter	Type	Description
ownerReferences	ownerReferences object	Definition Owner object Constraints - The value is automatically generated after the object is created. A user-defined value will not take effect. - This parameter is not present in the response of the API for creating a node. Range N/A Default Value N/A

Table 4-434 ownerReferences

Parameter	Type	Description
nodepoolName	String	Definition Node pool name Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A
nodepoolID	String	Definition Node pool ID Constraints The value is automatically generated after the object is created. A user-defined value will not take effect. Range N/A Default Value N/A

Table 4-435 NodeSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A
az	String	Details: AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to a node created using APIs. For details about AZs supported by CCE, see Regions and Endpoints . Constraints: When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random . Options: N/A Default value: N/A

Parameter	Type	Description
os	String	Details: OS type of a node. For details about the supported OSs, see Node OSs . Constraints: • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Options: N/A Default value: N/A
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Details: Disk information of a node Constraints: None

Parameter	Type	Description
dataVolumes	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume. Constraints: • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left blank only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodePublicIP object	Details: EIP of a node Constraints: This parameter is not supported when you add a node during node pool creation.
nodeNicSpec	NodeNicSpec object	Details: ENI of a node Constraints: None

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Details: Defined in key-value pairs. • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify the initial value for this parameter and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints: A maximum of 20 key-value pairs are allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Details: Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints: • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Details: Container runtime • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Type	Description
initializedConditions	Array of strings	Details: Custom initialization flag, which is left blank by default. Before CCE nodes are initialized, they are tainted with node.cloudprovider.kubernetes.io/uninitialized to prevent pods from being scheduled to them. When creating a node, you can specify the initializedConditions parameter to control the taint removal time. By default, the timeout interval is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two labels of types CCEInitial and CustomedInitial and setting the status to True, as shown in the following: 3. CCE polls status.Conditions of nodes to check whether there are conditions of types CCEInitial and CustomedInitial. If such conditions exist and the status is True, the node initialization is complete and the initialization taint is removed. 4. initializedConditions allows you to configure a timeout period during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout period is 15 minutes. After the timeout period is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial

Parameter	Type	Description
		status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 seconds. • Each value has 1 to 20 characters, which must consist of letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout period is minute (m).
extendParam	NodeExtendParam object	Details: Extended parameters for creating a node Constraints: None
hostnameConfig	HostnameConfig object	Details: Kubernetes node name configuration parameter Constraints: The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
partition	String	Details: Zone that a node belongs to. It can be the central cloud or an edge site. Constraints: This field can be specified only for CCE Turbo clusters that support distributed clouds. Options: • center: central cloud • AZ ID of the edge site Default value: N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product . Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Options: N/A Default value: N/A

Table 4-436 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-437 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-438 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-439 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-440 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-441 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-442 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-443 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-444 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-445 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-446 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-447 NodePublicIP

Parameter	Type	Description
ids	Array of strings	Details: IDs of existing EIPs Constraints: - The quantity cannot be greater than the number of nodes to be created. - If ids has been specified, you do not need to configure count and eip.
count	Integer	Details: Number of EIPs to be dynamically created Constraints: count and eip must be configured simultaneously. Options: N/A Default value: N/A
eip	NodeEIPSpec object	Details: EIP configuration Constraints: This parameter is not supported when you create a node pool.

Table 4-448 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-449 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Type	Description
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-450 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-451 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-452 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-453 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , _type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-454 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-455 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <code>echo -n "**Content to be encoded*" base64</code> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot . Constraints: This parameter is valid only when billingMode is set to 0 . Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-456 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-457 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Table 4-458 NodeStatus

Parameter	Type	Description
phase	String	Definition Node status Constraints This parameter is not present in the response of the API for creating a node. Range Node resource lifecycle management (such as installation and uninstallation) status and Kubernetes node status in a cluster. The options include: • Build: The node is being created. • Installing: The node is being managed. • Upgrading: The node is being upgraded. • Active: The node is running properly. • Abnormal: The node is abnormal. • Deleting: The node is being deleted. • Error: The node is faulty. Default Value N/A
lastProbeTime	String	Definition Last time when the node status was checked. If the cluster is in the abnormal, frozen, or intermediate state (for example, creating), the node status check may be affected. If checking a node's status takes longer than 5 minutes, the result is considered meaningless. Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
jobID	String	Definition Job ID during creation or deletion Range N/A
serverId	String	Definition ID of the underlying ECS or BMS node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIP	String	Definition Private IP address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A
privateIPv6IP	String	Definition Private IPv6 address of the primary network interface on a node Constraints This parameter is not present in the response of the API for creating a node. Range N/A

Parameter	Type	Description
publicIP	String	Definition Node EIP. If the ECS data is not synchronized in real time, you can synchronize the node data through the console. Constraints This parameter is not present in the response of the API for creating a node. Range N/A
deleteStatus	DeleteStatus object	Definition Resource status during resource deletion Constraints This parameter is returned only when a node is deleted. Range N/A
configurationUp- ToDate	Boolean	Definition Whether the node configuration is consistent with the latest configuration of the node template in the node pool to which the node belongs. When a node pool's OS or runtime is updated, the existing nodes may have a different OS or runtime, causing configurationUpToDate to be false . However, resetting the nodes will synchronize their OSs and runtimes with the node pool, setting configurationUpToDate to true . Constraints This parameter is not present in the response of the API for creating or updating a node. Range N/A

Table 4-459 DeleteStatus

Parameter	Type	Description
previous_total	Integer	Definition The total number of existing cluster resource records during node deletion Range N/A
current_total	Integer	Definition The latest number of resource records. This number is generated based on the current cluster resource records. Range N/A
updated	Integer	Definition The total number of to-be-deleted resource records updated during node deletion Range N/A
added	Integer	Definition The total number of to-be-deleted resource records added during node deletion Range N/A
deleted	Integer	Definition The total number of resource records deleted during node deletion Range N/A

Example Requests

None

Example Responses

Status code: 200

The job for deleting a node is successfully delivered.

```
{  
  "kind" : "Node",  
  "apiVersion" : "v3",  
}
```

```
"metadata" : {
  "name" : "new-hostname",
  "uid" : "cc697ad9-9563-11e8-8ea7-0255ac106311",
  "creationTimestamp" : "2018-08-01 08:20:49.944664515 +0000 UTC",
  "updateTimestamp" : "2018-08-01 09:20:05.644032347 +0000 UTC",
  "annotations" : {
    "kubernetes.io/node-pool.id" : "az1.dc1#s1.medium#EulerOS 2.2"
  }
},
"spec" : {
  "flavor" : "s1.medium",
  "az" : "az1.dc1",
  "os" : "EulerOS 2.2",
  "login" : {
    "sshKey" : "KeyPair-001"
  },
  "rootVolume" : {
    "volumetype" : "SAS",
    "size" : 40
  },
  "dataVolumes" : [ {
    "volumetype" : "SAS",
    "size" : 100
  } ],
  "publicIP" : {
    "eip" : {
      "bandwidth" : { }
    }
  },
  "billingMode" : 0
},
"status" : {
  "phase" : "Deleting",
  "jobID" : "661f6f7d-956c-11e8-a916-0255ac10575d",
  "serverId" : "5b504f8d-33f1-4ab7-a600-b62dac967d72",
  "privateIP" : "192.168.0.69",
  "publicIP" : "10.154.194.59"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class DeleteNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```

```
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();

DeleteNodeRequest request = new DeleteNodeRequest();
request.withClusterId("{cluster_id}");
request.withNodeId("{node_id}");
try {
    DeleteNodeResponse response = client.deleteNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteNodeRequest()
        request.cluster_id = "{cluster_id}"
        request.node_id = "{node_id}"
        response = client.delete_node(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteNodeRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodeId = "{node_id}"
    response, err := client.DeleteNode(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for deleting a node is successfully delivered.

Error Codes

See [Error Codes](#).

4.3.6 Enabling Scale-In Protection for a Node

Function

This API is used to enable scale-in protection for a node. A node with scale-in protection enabled cannot be removed by simply changing the number of nodes in the node pool.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/locknodescaledown

Table 4-460 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-461 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-462 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	Yes	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value List
nodeList	No	Array of strings	Definition ID list of nodes for which scale-in protection needs to be enabled. For details about how to obtain the IDs, see How to Obtain Parameters in the API URI . Constraints N/A

Response Parameters

Status code: 200

The scale-in protection is enabled for the node.

None

Example Requests

Lock a node so that it will not be scaled in.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/locknodescaledown
{
```

```
"kind" : "List",  
"apiVersion" : "v3",  
"nodeList" : [ "xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx" ]  
}
```

Example Responses

None

Status Codes

Status Code	Description
200	The scale-in protection is enabled for the node.

Error Codes

See [Error Codes](#).

4.3.7 Disabling Scale-In Protection for a Node

Function

This API is used to disable scale-in protection for a node. A node with scale-in protection disabled can be removed by changing the number of nodes in the node pool. Scale-in protection can be disabled only for pay-per-use nodes.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/
unlocknodescaledown

Table 4-463 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-464 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-465 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Definition API version. The value is fixed at v3 . Constraints N/A Range v3 Default Value N/A
kind	Yes	String	Definition API type. The value is fixed at List . Constraints N/A Range List Default Value N/A
nodeList	No	Array of strings	Definition ID list of nodes for which scale-in protection needs to be disabled. For details about how to obtain the IDs, see How to Obtain Parameters in the API URI . Constraints N/A

Response Parameters

Status code: 200

The scale-in protection is disabled for the node.

None

Example Requests

Unlock a node which cannot be scaled in.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/unlocknodescaledown
{
```

```
"kind" : "List",  
"apiVersion" : "v3",  
"nodeList" : [ "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" ]  
}
```

Example Responses

None

Status Codes

Status Code	Description
200	The scale-in protection is disabled for the node.

Error Codes

See [Error Codes](#).

4.3.8 Synchronizing Nodes

Function

This API is used to synchronize nodes.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v2/projects/{project_id}/clusters/{cluster_id}/nodes/{node_id}/sync

Table 4-466 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
node_id	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-467 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-468 Response body parameters

Parameter	Type	Description
-	String	Definition The value is fixed at Sync node success , which indicates that the node has been synchronized. Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class SyncNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
```

```
        .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
SyncNodeRequest request = new SyncNodeRequest();
request.withClusterId("{cluster_id}");
request.withNodeId("{node_id}");
try {
    SyncNodeResponse response = client.syncNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = SyncNodeRequest()
        request.cluster_id = "{cluster_id}"
        request.node_id = "{node_id}"
        response = client.sync_node(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main
```

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.SyncNodeRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodeId = "{node_id}"
    response, err := client.SyncNode(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Node configuration synchronized successfully.

Error Codes

See [Error Codes](#).

4.3.9 Synchronizing Nodes in Batches

Function

This API is used to synchronize nodes in batches.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/sync

Table 4-469 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-470 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-471 Response body parameters

Parameter	Type	Description
-	String	Definition The value is fixed at Start to batch sync nodes , which indicates that multiple nodes have been synchronized at a time. Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

Batch node configuration synchronized.

Start to batch sync nodes

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class BatchSyncNodesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```

```
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();

BatchSyncNodesRequest request = new BatchSyncNodesRequest();
request.withClusterId("{cluster_id}");
try {
    BatchSyncNodesResponse response = client.batchSyncNodes(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchSyncNodesRequest()
        request.cluster_id = "{cluster_id}"
        response = client.batch_sync_nodes(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchSyncNodesRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.BatchSyncNodes(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Batch node configuration synchronized.

Error Codes

See [Error Codes](#).

4.3.10 Accepting a Node

Function

This API is used to accept a node into a specified cluster.

NOTE

The URL for cluster management is in the format of `https://Endpoint/uri`. In the URL, **uri** indicates the resource path, that is, the path for API access.

Constraints

- ECSs, DeHs, and BMSs can be managed.
- The nodes to be managed must be running and do not belong to any other clusters. Additionally, the nodes cannot be labeled with **CCE-Dynamic-Provisioning-Node**.
- The nodes to be managed must be in the same VPC as the target cluster. If the cluster version is earlier than v1.13.10, the nodes and the cluster must also be in the same subnet.
- The nodes to be managed must have data disks attached, which can be local disks (disk-intensive disks). Each node should have a data disk of at least 20 GiB attached, and any existing data disks cannot be smaller than 10 GiB.
- Each of the nodes to be managed has at least 2 vCPUs and 4 GiB of memory and only one ENI bound.
- If an enterprise project is used, a node to be managed must be in the same enterprise project as the target cluster. Otherwise, resources cannot be identified during node management, ultimately rendering the node unmanageable. In clusters of v1.21.15-r0, v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later versions, the nodes to be managed can be in different enterprise projects from that of the target cluster, and the enterprise projects of these nodes remain unchanged after the nodes are accepted.
- If IPv6 is enabled for a cluster, only nodes in a subnet with IPv6 enabled can be accepted and managed. If IPv6 is not enabled for the cluster, only nodes in a subnet without IPv6 enabled can be accepted.
- Nodes in a CCE Turbo cluster must support sub-ENIs, or each of them can have at least 16 ENIs bound. For details about the node flavors, see the node flavors that can be selected on the console when you create a node.
- Data disks that have been partitioned will be ignored during node management. Ensure that there is at least one unpartitioned data disk meeting the specifications attached to the node.

Calling Method

For details, see [Calling APIs](#).

URI

POST `/api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/add`

Table 4-472 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-473 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-474 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	Yes	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value List
nodeList	Yes	Array of AddNode objects	Definition List of the nodes to be managed Constraints A maximum of 200 nodes can be managed at a time.

Table 4-475 AddNode

Parameter	Mandatory	Type	Description
serverID	Yes	String	Definition Server ID, which can be obtained from the ECS or BMS console. NOTE To obtain the server ID, log in to the management console, choose Service List > Compute > Elastic Cloud Server or Bare Metal Server , click the server name, and copy the server ID on the server details page. Constraints N/A Range N/A Default Value N/A
spec	Yes	ReinstallNodeSpec object	Definition Node reinstallation parameter settings Constraints N/A

Table 4-476 ReinstallNodeSpec

Parameter	Mandatory	Type	Description
os	Yes	String	Definition OS. If you specify a private image, the actual OS version of the IMS image is used. Select an OS version supported by the current cluster, for example, Huawei Cloud EulerOS 2.0, Ubuntu 22.04, EulerOS 2.9, CentOS 7.6, and EulerOS 2.8. Constraints N/A Range N/A Default Value N/A
login	No	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A

Parameter	Mandatory	Type	Description
name	No	String	Definition Node name. Specifying this field during reinstallation will change the node name, and the server name will change accordingly. By default, the current server name is used as the node name. Constraints The node name must contain 1 to 56 characters that start with a lowercase letter. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range N/A Default Value N/A
serverConfig	No	ReinstallServerConfig object	Definition Server settings Constraints N/A
volumeConfig	No	ReinstallVolumeConfig object	Definition Volume management settings Constraints N/A
runtimeConfig	No	ReinstallRuntimeConfig object	Definition Container runtime settings Constraints N/A
k8sOptions	No	ReinstallK8sOptionsConfig object	Definition Kubernetes node settings Constraints N/A
lifecycle	No	NodeLifecycleConfig object	Definition Customized node lifecycle settings Constraints N/A

Parameter	Mandatory	Type	Description
initializedConditions	No	Array of strings	Definition Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When managing or resetting a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. To manage or reset a node, add "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions and add two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint will be removed. 4. initializedConditions allows you to configure a timeout interval during

Parameter	Mandatory	Type	Description
			node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. The following is an example of conditions settings: <pre>status: conditions: - type: CCEInitial status: 'True' - type: CustomedInitial status: 'True'</pre> Constraints • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	No	ReinstallExtendParam object	Definition (Deprecated) Extended parameters for reinstallation Constraints N/A

Parameter	Mandatory	Type	Description
hostnameConfig	No	HostnameConfig object	Definition Parameter for configuring Kubernetes node name. The supported cluster version is v1.23.6-r0 to v1.25, or v1.25.2-r0 or later. Constraints This parameter is not supported during node resetting.

Table 4-477 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Mandatory	Type	Description
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true, no user password can be specified. Range N/A Default Value false
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true, no SSH key can be specified. Range N/A Default Value false

Table 4-478 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-479 ReinstallServerConfig

Parameter	Mandatory	Type	Description
userTags	No	Array of UserTag objects	Definition Cloud server resource tags Constraints The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
rootVolume	No	ReinstallVolumeSpec object	Definition System disk reinstallation settings Constraints N/A

Table 4-480 UserTag

Parameter	Mandatory	Type	Description
key	No	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Mandatory	Type	Description
value	No	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, Default value: N/A

Table 4-481 ReinstallVolumeSpec

Parameter	Mandatory	Type	Description
imageID	No	String	Definition Private image ID. NOTE To obtain the value ID, choose Service List > Compute > Image Management Service , click the Private Image tab, click the image name, and copy the image ID on the page displayed. Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
cmkID	No	String	Definition CMK ID. If this parameter is left empty, the disk is not encrypted. NOTE For details about how to obtain the key ID, see Querying the Key List. Constraints N/A Range N/A Default Value N/A

Table 4-482 ReinstallVolumeConfig

Parameter	Mandatory	Type	Description
lvmConfig	No	String	Definition (Deprecated) Configuration item of a Docker data disk. Example of the default configuration: <pre>"lvmConfig":{"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> The following fields are included: • userLV: size of the user space, for example, vgpaas/20%VG. • userPath: mount path of the user space, for example, /home/wqt-test. • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
storage	No	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node. Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.

Table 4-483 Storage

Parameter	Mandatory	Type	Description
storageSelectors	Yes	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None
storageGroups	Yes	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-484 StorageSelectors

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
storageType	Yes	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	No	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-485 matchLabels

Parameter	Mandatory	Type	Description
size	No	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	No	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	No	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
metadataCmkid	No	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	No	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-486 StorageGroups

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: • If cceManaged is set to true, the name field must be set to vgpaas. • If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. • If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	No	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
selectorNames	Yes	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.
virtualSpaces	Yes	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-487 VirtualSpace

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	Yes	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Mandatory	Type	Description
lvmConfig	No	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	No	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-488 LVMConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
path	No	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-489 RuntimeConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-490 ReinstallRuntimeConfig

Parameter	Mandatory	Type	Description
dockerBaseSize	No	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. If this parameter is left empty or is set to 0, the default value is used. In Device Mapper mode, the default value is 10. In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container, and the dockerBaseSize setting takes effect only on nodes running EulerOS/Huawei Cloud EulerOS 2.0 in clusters of the new versions. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
containerBaseSize	No	Integer	Definition Available disk space of a single container on a node (unit: G). If this parameter is left empty or is set to 0, the default value is used. In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container. In Device Mapper mode, the default value is 10, and the containerBaseSize setting takes effect only on nodes running EulerOS/Huawei Cloud EulerOS 2.0 in clusters of v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. Constraints NOTE This parameter cannot be updated when a node pool is being updated. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
runtime	No	Runtime object	Details: Container runtime • Clusters earlier than v1.25: The default value is docker. • Clusters v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Table 4-491 Runtime

Parameter	Mandatory	Type	Description
name	No	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-492 ReinstallK8sOptionsConfig

Parameter	Mandatory	Type	Description
labels	No	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints A maximum of 20 key-value pairs is allowed.

Parameter	Mandatory	Type	Description
taints	No	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Mandatory	Type	Description
maxPods	No	Integer	Definition Maximum number of pods that can be created on a node, including the default pods. This parameter is used to prevent the node from being overloaded due to excessive pods. Set this parameter based on your service requirements. Constraints N/A Range 16 to 256 Default Value N/A

Table 4-493 Taint

Parameter	Mandatory	Type	Description
key	Yes	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
value	No	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	Yes	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-494 NodeLifecycleConfig

Parameter	Mandatory	Type	Description
preInstall	No	String	Definition Pre-installation script. The characters of both the pre-installation and post-installation scripts are centrally calculated. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Constraints The length cannot exceed 10,240 bytes. Range N/A Default Value N/A
postInstall	No	String	Definition Pre-installation script. The characters of both the pre-installation and post-installation scripts are centrally calculated. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Constraints The length cannot exceed 10,240 bytes. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
waitPostInstallFinish	No	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during operations such as node reset, management, and batch reset. If the node belongs to a node pool, the node pool settings apply. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Table 4-495 ReinstallExtendParam

Parameter	Mandatory	Type	Description
alpha.cce/ NodeImageID	No	String	Definition (Deprecated) ID of the user image to run the target OS. Specifying this parameter is equivalent to specifying imageID in ReinstallVolumeSpec . The original value will be overwritten. Constraints N/A Range N/A Default Value N/A

Table 4-496 HostnameConfig

Parameter	Mandatory	Type	Description
type	Yes	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Response Parameters

Status code: 200

Table 4-497 Response body parameters

Parameter	Type	Description
jobid	String	Definition ID returned after a job is submitted. This ID can be used to query the execution of the job. Range N/A

Example Requests

Add a node running EulerOS 2.5 to a cluster.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/add
{
  "kind" : "List",
  "apiVersion" : "v3",
  "nodeList" : [ {
    "serverID" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "spec" : {
      "name" : "my-ecs-0001",
      "os" : "EulerOS 2.5",
      "login" : {
        "sshKey" : "KeyPair-001"
      }
    }
  }
]
```

Example Responses

Status code: 200

The job for accepting a node into a specified cluster is successfully delivered.

```
{
  "jobid" : "2ec9b78d-9368-46f3-8f29-d1a95622a568"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Add a node running EulerOS 2.5 to a cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
```



```
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.UUID;
import java.util.List;
import java.util.ArrayList;

public class AddNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        AddNodeRequest request = new AddNodeRequest();
        request.withClusterId("{cluster_id}");
        AddNodeList body = new AddNodeList();
        Login loginSpec = new Login();
        loginSpec.withSshKey("KeyPair-001");
        ReinstallNodeSpec specNodeList = new ReinstallNodeSpec();
        specNodeList.withOs("EulerOS 2.5")
            .withLogin(loginSpec)
            .withName("my-ecs-0001");
        List<AddNode> listbodyNodeList = new ArrayList<>();
        listbodyNodeList.add(
            new AddNode()
                .withServerID(UUID.fromString("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"))
                .withSpec(specNodeList)
        );
        body.withNodeList(listbodyNodeList);
        body.withKind("List");
        body.withApiVersion("v3");
        request.withBody(body);
        try {
            AddNodeResponse response = client.addNode(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Add a node running EulerOS 2.5 to a cluster.

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AddNodeRequest()
        request.cluster_id = "{cluster_id}"
        loginSpec = Login(
            ssh_key="KeyPair-001"
        )
        specNodeList = ReinstallNodeSpec(
            os="EulerOS 2.5",
            login=loginSpec,
            name="my-ecs-0001"
        )
        listNodeListbody = [
            AddNode(
                server_id="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
                spec=specNodeList
            )
        ]
        request.body = AddNodeList(
            node_list=listNodeListbody,
            kind="List",
            api_version="v3"
        )
        response = client.add_node(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Add a node running EulerOS 2.5 to a cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

```
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.AddNodeRequest{}
request.ClusterId = "{cluster_id}"
sshKeyLogin := "KeyPair-001"
loginSpec := &model.Login{
    SshKey: &sshKeyLogin,
}
nameSpec := "my-ecs-0001"
specNodeList := &model.ReinstallNodeSpec{
    Os: "EulerOS 2.5",
    Login: loginSpec,
    Name: &nameSpec,
}
var listNodeListbody = []model.AddNode{
    {
        ServerID: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
        Spec: specNodeList,
    },
}
request.Body = &model.AddNodeList{
    NodeList: listNodeListbody,
    Kind: "List",
    ApiVersion: "v3",
}
response, err := client.AddNode(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for accepting a node into a specified cluster is successfully delivered.

Error Codes

See [Error Codes](#).

4.3.11 Managing a Node in a Customized Node Pool

Function

This API is used to manage a node in a customized node pool of a cluster. Spot instances are not supported.

NOTE

- ECSs, DeHs, and BMSs can be managed.
- The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** specifies the resource path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/nodes/add

Table 4-498 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-499 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-500 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	Yes	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value List
nodeList	Yes	Array of AddNodesToNodePool objects	Definition List of nodes to be managed. A maximum of 200 nodes can be managed at a time. Constraints N/A Range N/A Default Value N/A

Table 4-501 AddNodesToNodePool

Parameter	Mandatory	Type	Description
serverID	Yes	String	Definition Server ID, which can be obtained from the ECS or BMS console. NOTE To obtain the server ID, log in to the management console, choose Service List > Compute > Elastic Cloud Server or Bare Metal Server , click the server name, and copy the server ID on the server details page. Constraints N/A Range N/A Default Value N/A

Response Parameters

Status code: 200

Table 4-502 Response body parameters

Parameter	Type	Description
jobid	String	Definition ID returned after a job is submitted. This ID can be used to query the execution of the job. Range N/A

Example Requests

Manage a node in a customized node pool.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepool/{nodepool_id}/nodes/add
{
  "kind": "List",
  "apiVersion": "v3",
  "nodeList": [ {
    "serverID": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx1"
  }, {
```



```
"serverID" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx2"
} ]
}
```

Example Responses

Status code: 200

The job for managing a node in the customized node pool in the cluster delivered.

```
{
  "jobid" : "2ec9b78d-9368-46f3-8f29-d1a95622a568"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Manage a node in a customized node pool.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.UUID;
import java.util.List;
import java.util.ArrayList;

public class AddNodesToNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        AddNodesToNodePoolRequest request = new AddNodesToNodePoolRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        AddNodesToNodePoolList body = new AddNodesToNodePoolList();
        List<AddNodesToNodePool> listbodyNodeList = new ArrayList<>();
        listbodyNodeList.add(
            new AddNodesToNodePool()
                .withServerID(UUID.fromString("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx1"))
        );
    }
}
```

```
);  
listbodyNodeList.add(  
    new AddNodesToNodePool()  
        .withServerID(UUID.fromString("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx2"))  
);  
body.withNodeList(listbodyNodeList);  
body.withKind("List");  
body.withApiVersion("v3");  
request.withBody(body);  
try {  
    AddNodesToNodePoolResponse response = client.addNodesToNodePool(request);  
    System.out.println(response.toString());  
} catch (ConnectionException e) {  
    e.printStackTrace();  
} catch (RequestTimeoutException e) {  
    e.printStackTrace();  
} catch (ServiceResponseException e) {  
    e.printStackTrace();  
    System.out.println(e.getStatusCode());  
    System.out.println(e.getRequestId());  
    System.out.println(e.getErrorCode());  
    System.out.println(e.getErrorMsg());  
}  
}  
}
```

Python

Manage a node in a customized node pool.

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = CceClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(CceRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = AddNodesToNodePoolRequest()  
        request.cluster_id = "{cluster_id}"  
        request.nodepool_id = "{nodepool_id}"  
        listNodeListbody = [  
            AddNodesToNodePool(  
                server_id="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx1"  
            ),  
            AddNodesToNodePool(  
                server_id="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx2"  
            )  
        ]  
        request.body = AddNodesToNodePoolList(  
            node_list=listNodeListbody,  

```

```
        kind="List",
        api_version="v3"
    )
    response = client.add_nodes_to_node_pool(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Manage a node in a customized node pool.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.AddNodesToNodePoolRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    var listNodeListbody = []model.AddNodesToNodePool{
        {
            ServerID: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx1",
        },
        {
            ServerID: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx2",
        },
    }
    request.Body = &model.AddNodesToNodePoolList{
        NodeList: listNodeListbody,
        Kind: "List",
        ApiVersion: "v3",
    }
    response, err := client.AddNodesToNodePool(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for managing a node in the customized node pool in the cluster delivered.

Error Codes

See [Error Codes](#).

4.3.12 Resetting a Node

Function

This API is used to reset a node in a specified cluster.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/reset

Table 4-503 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-504 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-505 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Definition API version Constraints The value cannot be changed. Range N/A Default Value v3
kind	Yes	String	Definition API type Constraints The value cannot be changed. Range N/A Default Value List
nodeList	Yes	Array of ResetNode objects	Definition List of nodes to be reset. A maximum of 200 nodes can be reset at a time. Constraints N/A

Table 4-506 ResetNode

Parameter	Mandatory	Type	Description
nodeID	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
spec	Yes	ReinstallNodeSpec object	Definition Node reinstallation parameter settings. Nodes in a node pool cannot be specified externally. The nodes will be reinstalled based on the node pool settings. By default, this parameter is mandatory for the nodes in the default node pool. Constraints N/A Range N/A Default Value N/A

Table 4-507 ReinstallNodeSpec

Parameter	Mandatory	Type	Description
os	Yes	String	Definition OS. If you specify a private image, the actual OS version of the IMS image is used. Select an OS version supported by the current cluster, for example, Huawei Cloud EulerOS 2.0, Ubuntu 22.04, EulerOS 2.9, CentOS 7.6, and EulerOS 2.8. Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
login	No	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
name	No	String	Definition Node name. Specifying this field during reinstallation will change the node name, and the server name will change accordingly. By default, the current server name is used as the node name. Constraints The node name must contain 1 to 56 characters that start with a lowercase letter. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range N/A Default Value N/A
serverConfig	No	ReinstallServerConfig object	Definition Server settings Constraints N/A
volumeConfig	No	ReinstallVolumeConfig object	Definition Volume management settings Constraints N/A
runtimeConfig	No	ReinstallRuntimeConfig object	Definition Container runtime settings Constraints N/A

Parameter	Mandatory	Type	Description
k8sOptions	No	ReinstallK8sOptionsConfig object	Definition Kubernetes node settings Constraints N/A
lifecycle	No	NodeLifecycleConfig object	Definition Customized node lifecycle settings Constraints N/A

Parameter	Mandatory	Type	Description
initializedConditions	No	Array of strings	Definition Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When managing or resetting a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. To manage or reset a node, add "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions and add two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint will be removed. 4. initializedConditions allows you to configure a timeout interval during

Parameter	Mandatory	Type	Description
			node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. The following is an example of conditions settings: <pre>status: conditions: - type: CCEInitial status: 'True' - type: CustomedInitial status: 'True'</pre> Constraints • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	No	ReinstallExtendParam object	Definition (Deprecated) Extended parameters for reinstallation Constraints N/A

Parameter	Mandatory	Type	Description
hostnameConfig	No	HostnameConfig object	Definition Parameter for configuring Kubernetes node name. The supported cluster version is v1.23.6-r0 to v1.25, or v1.25.2-r0 or later. Constraints This parameter is not supported during node resetting.

Table 4-508 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Mandatory	Type	Description
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-509 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-510 ReinstallServerConfig

Parameter	Mandatory	Type	Description
userTags	No	Array of UserTag objects	Definition Cloud server resource tags Constraints The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
rootVolume	No	ReinstallVolumeSpec object	Definition System disk reinstallation settings Constraints N/A

Table 4-511 UserTag

Parameter	Mandatory	Type	Description
key	No	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Mandatory	Type	Description
value	No	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<> Default value: N/A

Table 4-512 ReinstallVolumeSpec

Parameter	Mandatory	Type	Description
imageID	No	String	Definition Private image ID. NOTE To obtain the value ID, choose Service List > Compute > Image Management Service , click the Private Image tab, click the image name, and copy the image ID on the page displayed. Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
cmkID	No	String	Definition CMK ID. If this parameter is left empty, the disk is not encrypted. NOTE For details about how to obtain the key ID, see Querying the Key List. Constraints N/A Range N/A Default Value N/A

Table 4-513 ReinstallVolumeConfig

Parameter	Mandatory	Type	Description
lvmConfig	No	String	Definition (Deprecated) Configuration item of a Docker data disk. Example of the default configuration: <pre>"lvmConfig":{"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> The following fields are included: • userLV: size of the user space, for example, vgpaas/20%VG. • userPath: mount path of the user space, for example, /home/wqt-test. • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
storage	No	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node. Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.

Table 4-514 Storage

Parameter	Mandatory	Type	Description
storageSelectors	Yes	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None
storageGroups	Yes	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-515 StorageSelectors

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
storageType	Yes	String	Details: Storage type Constraints: • The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. • The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	No	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size , volumeType , metadataEncrypted , metadataCmkid , and count fields in DataVolume are supported. Default value: N/A

Table 4-516 matchLabels

Parameter	Mandatory	Type	Description
size	No	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	No	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	No	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
metadataCmkid	No	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	No	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-517 StorageGroups

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: • If cceManaged is set to true, the name field must be set to vgpaas. • If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. • If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	No	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
selectorNames	Yes	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.
virtualSpaces	Yes	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-518 VirtualSpace

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	Yes	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Mandatory	Type	Description
lvmConfig	No	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	No	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-519 LVMConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
path	No	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-520 RuntimeConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-521 ReinstallRuntimeConfig

Parameter	Mandatory	Type	Description
dockerBaseSize	No	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. If this parameter is left empty or is set to 0, the default value is used. In Device Mapper mode, the default value is 10. In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container, and the dockerBaseSize setting takes effect only on nodes running EulerOS/Huawei Cloud EulerOS 2.0 in clusters of the new versions. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
containerBaseSize	No	Integer	Definition Available disk space of a single container on a node (unit: G). If this parameter is left empty or is set to 0, the default value is used. In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container. In Device Mapper mode, the default value is 10, and the containerBaseSize setting takes effect only on nodes running EulerOS/Huawei Cloud EulerOS 2.0 in clusters of v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. Constraints NOTE This parameter cannot be updated when a node pool is being updated. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
runtime	No	Runtime object	Details: Container runtime - Clusters earlier than v1.25: The default value is docker. - Clusters v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Table 4-522 Runtime

Parameter	Mandatory	Type	Description
name	No	String	Details: Container runtime Constraints: None Options: N/A Default value: - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-523 ReinstallK8sOptionsConfig

Parameter	Mandatory	Type	Description
labels	No	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints A maximum of 20 key-value pairs is allowed.

Parameter	Mandatory	Type	Description
taints	No	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Mandatory	Type	Description
maxPods	No	Integer	Definition Maximum number of pods that can be created on a node, including the default pods. This parameter is used to prevent the node from being overloaded due to excessive pods. Set this parameter based on your service requirements. Constraints N/A Range 16 to 256 Default Value N/A

Table 4-524 Taint

Parameter	Mandatory	Type	Description
key	Yes	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
value	No	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	Yes	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-525 NodeLifecycleConfig

Parameter	Mandatory	Type	Description
preInstall	No	String	Definition Pre-installation script. The characters of both the pre-installation and post-installation scripts are centrally calculated. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Constraints The length cannot exceed 10,240 bytes. Range N/A Default Value N/A
postInstall	No	String	Definition Pre-installation script. The characters of both the pre-installation and post-installation scripts are centrally calculated. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Constraints The length cannot exceed 10,240 bytes. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
waitPostInstallFinish	No	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during operations such as node reset, management, and batch reset. If the node belongs to a node pool, the node pool settings apply. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Table 4-526 ReinstallExtendParam

Parameter	Mandatory	Type	Description
alpha.cce/ NodeImageID	No	String	Definition (Deprecated) ID of the user image to run the target OS. Specifying this parameter is equivalent to specifying imageID in ReinstallVolumeSpec . The original value will be overwritten. Constraints N/A Range N/A Default Value N/A

Table 4-527 HostnameConfig

Parameter	Mandatory	Type	Description
type	Yes	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Response Parameters

Status code: 200

Table 4-528 Response body parameters

Parameter	Type	Description
jobid	String	Definition ID returned after a job is submitted. This ID can be used to query the execution of the job. Range N/A

Example Requests

- Reset the nodes running EulerOS 2.5 in the default node pool.

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/reset

```
{
  "kind": "List",
  "apiVersion": "v3",
  "nodeList": [ {
    "nodeID": "yyyyyyyy-yyyy-yyyy-yyyyyyyyyyyy",
    "spec": {
      "name": "my-ecs-0001",
      "os": "EulerOS 2.5",
      "login": {
        "sshKey": "KeyPair-001"
      }
    }
  }
]
```

- Reset a node in a node pool (invalid spec).

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/reset

```
{
  "kind": "List",
  "apiVersion": "v3",
  "nodeList": [ {
    "nodeID": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "spec": {
      "name": "my-ecs-0001",
      "os": "EulerOS 2.5",
      "login": {
        "sshKey": "KeyPair-001"
      }
    }
  }
]
```

Example Responses

Status code: 200

The job for resetting a node in a specified cluster is successfully delivered.

```
{
  "jobid": "2ec9b78d-9368-46f3-8f29-d1a95622a568"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Reset the nodes running EulerOS 2.5 in the default node pool.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.UUID;
import java.util.List;
import java.util.ArrayList;

public class ResetNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        ResetNodeRequest request = new ResetNodeRequest();
        request.withClusterId("{cluster_id}");
        ResetNodeList body = new ResetNodeList();
        Login loginSpec = new Login();
        loginSpec.withSshKey("KeyPair-001");
        ReinstallNodeSpec specNodeList = new ReinstallNodeSpec();
        specNodeList.withOs("EulerOS 2.5")
            .withLogin(loginSpec)
            .withName("my-ecs-0001");
        List<ResetNode> listbodyNodeList = new ArrayList<>();
        listbodyNodeList.add(
            new ResetNode()
                .withNodeID(UUID.fromString("yyyyyyyy-yyy-yyy-yyy-yyyyyyyyyyyy"))
                .withSpec(specNodeList)
        );
        body.withNodeList(listbodyNodeList);
        body.withKind("List");
        body.withApiVersion("v3");
        request.withBody(body);
        try {
            ResetNodeResponse response = client.resetNode(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        }
    }
}
```

```
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

- Reset a node in a node pool (invalid spec).

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.UUID;
import java.util.List;
import java.util.ArrayList;

public class ResetNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ResetNodeRequest request = new ResetNodeRequest();
        request.withClusterId("{cluster_id}");
        ResetNodeList body = new ResetNodeList();
        Login loginSpec = new Login();
        loginSpec.withSshKey("KeyPair-001");
        ReinstallNodeSpec specNodeList = new ReinstallNodeSpec();
        specNodeList.withOs("EulerOS 2.5")
            .withLogin(loginSpec)
            .withName("my-ecs-0001");
        List<ResetNode> listbodyNodeList = new ArrayList<>();
        listbodyNodeList.add(
            new ResetNode()
                .withNodeID(UUID.fromString("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"))
                .withSpec(specNodeList)
        );
        body.withNodeList(listbodyNodeList);
        body.withKind("List");
        body.withApiVersion("v3");
        request.withBody(body);
        try {
```

```
        ResetNodeResponse response = client.resetNode(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

- Reset the nodes running EulerOS 2.5 in the default node pool.

coding: utf-8

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ResetNodeRequest()
        request.cluster_id = "{cluster_id}"
        loginSpec = Login(
            ssh_key="KeyPair-001"
        )
        specNodeList = ReinstallNodeSpec(
            os="EulerOS 2.5",
            login=loginSpec,
            name="my-ecs-0001"
        )
        listNodeListbody = [
            ResetNode(
                node_id="yyyyyyyy-yyy-yyy-yyyy-yyyyyyyyyyyy",
                spec=specNodeList
            )
        ]
        request.body = ResetNodeList(
            node_list=listNodeListbody,
            kind="List",
            api_version="v3"
        )
        response = client.reset_node(request)
        print(response)
```

```
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Reset a node in a node pool (invalid spec).

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ResetNodeRequest()
        request.cluster_id = "{cluster_id}"
        loginSpec = Login(
            ssh_key="KeyPair-001"
        )
        specNodeList = ReinstallNodeSpec(
            os="EulerOS 2.5",
            login=loginSpec,
            name="my-ecs-0001"
        )
        listNodeListbody = [
            ResetNode(
                node_id="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
                spec=specNodeList
            )
        ]
        request.body = ResetNodeList(
            node_list=listNodeListbody,
            kind="List",
            api_version="v3"
        )
        response = client.reset_node(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- Reset the nodes running EulerOS 2.5 in the default node pool.

```
package main

import (
```

```
"fmt"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ResetNodeRequest{}
    request.ClusterId = "{cluster_id}"
    sshKeyLogin := "KeyPair-001"
    loginSpec := &model.Login{
        SshKey: &sshKeyLogin,
    }
    nameSpec := "my-ecs-0001"
    specNodeList := &model.ReinstallNodeSpec{
        Os: "EulerOS 2.5",
        Login: loginSpec,
        Name: &nameSpec,
    }
    var listNodeListbody = []model.ResetNode{
        {
            NodeId: "yyyyyyyy-yyy-yyy-yyyy-yyyyyyyyyyyy",
            Spec: specNodeList,
        },
    }
    request.Body = &model.ResetNodeList{
        NodeList: listNodeListbody,
        Kind: "List",
        ApiVersion: "v3",
    }
    response, err := client.ResetNode(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

- Reset a node in a node pool (invalid spec).

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
```

```
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    // environment variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before  
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    // environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := cce.NewCceClient(  
        cce.CceClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ResetNodeRequest{  
        request.ClusterId = "{cluster_id}"  
        sshKeyLogin:= "KeyPair-001"  
        loginSpec := &model.Login{  
            SshKey: &sshKeyLogin,  
        }  
        nameSpec:= "my-ecs-0001"  
        specNodeList := &model.ReinstallNodeSpec{  
            Os: "EulerOS 2.5",  
            Login: loginSpec,  
            Name: &nameSpec,  
        }  
    }  
    var listNodeListbody = []model.ResetNode{  
        {  
            NodeID: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",  
            Spec: specNodeList,  
        },  
    }  
    request.Body = &model.ResetNodeList{  
        NodeList: listNodeListbody,  
        Kind: "List",  
        ApiVersion: "v3",  
    }  
    response, err := client.ResetNode(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for resetting a node in a specified cluster is successfully delivered.

Error Codes

See [Error Codes](#).

4.3.13 Removing a Node

Function

This API is used to remove a node from a specified cluster.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/operation/remove

Table 4-529 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-530 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-531 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	No	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	No	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value RemoveNodesTask

Parameter	Mandatory	Type	Description
spec	Yes	RemoveNodeSpec object	Definition Configuration information Constraints N/A

Table 4-532 RemoveNodesSpec

Parameter	Mandatory	Type	Description
login	No	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
nodes	Yes	Array of NodeItem objects	Definition List of nodes to be removed. A maximum of 200 nodes can be removed at a time. Constraints N/A

Table 4-533 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-534 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-535 NodeItem

Parameter	Mandatory	Type	Description
uid	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Response Parameters

Status code: 200

Table 4-536 Response body parameters

Parameter	Type	Description
apiVersion	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value RemoveNodesTask

Parameter	Type	Description
spec	RemoveNodesSpec object	Definition Configuration information Constraints N/A
status	TaskStatus object	Definition Task status Constraints N/A

Table 4-537 RemoveNodesSpec

Parameter	Type	Description
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
nodes	Array of NodeItem objects	Definition List of nodes to be removed. A maximum of 200 nodes can be removed at a time. Constraints N/A

Table 4-538 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
userPassword	UserPassword object	Details: Password used for node login Constraints: None
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-539 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^_-=+[{ }],./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-540 NodeItem

Parameter	Type	Description
uid	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Table 4-541 TaskStatus

Parameter	Type	Description
jobID	String	Definition Job ID, which is used by the caller to query the job progress. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Example Requests

Removing a Node

```
PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/operation/remove
{
  "spec": {
    "login": {
      "sshKey": "KeyPair-001"
    },
    "nodes": [ {
      "uid": "xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx"
    }, {
      "uid": "yyyyyyyy-yyyy-yyyy-yyyyyyyyyyyy"
    } ]
  }
}
```

Example Responses

Status code: 200

The job for removing a node in a specified cluster is successfully delivered.

```
{
  "spec" : {
    "login" : {
      "sshKey" : "KeyPair-001"
    },
    "nodes" : [ {
      "uid" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
    }, {
      "uid" : "yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy"
    } ]
  },
  "status" : {
    "jobID" : "2ec9b78d-9368-46f3-8f29-d1a95622a568"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Removing a Node

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class RemoveNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        RemoveNodeRequest request = new RemoveNodeRequest();
        request.withClusterId("{cluster_id}");
```

```
RemoveNodesTask body = new RemoveNodesTask();
List<NodeItem> listSpecNodes = new ArrayList<>();
listSpecNodes.add(
    new NodeItem()
        .withUid("xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx")
);
listSpecNodes.add(
    new NodeItem()
        .withUid("yyyyyyyy-yyyy-yyyy-yyyyyyyyyyyy")
);
Login loginSpec = new Login();
loginSpec.withSshKey("KeyPair-001");
RemoveNodesSpec specbody = new RemoveNodesSpec();
specbody.withLogin(loginSpec)
        .withNodes(listSpecNodes);
body.withSpec(specbody);
request.withBody(body);
try {
    RemoveNodeResponse response = client.removeNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Removing a Node

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RemoveNodeRequest()
        request.cluster_id = "{cluster_id}"
        listNodesSpec = [
            NodeItem(
                uid="xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx"
            )
        ]
        request.spec = listNodesSpec
        response = client.remove_node(request)
        print(response.to_dict())
    except exceptions.ApiException as e:
        print(e.to_dict())
    except exceptions.ApiException as e:
        print(e.to_dict())
```

```
    ),  
    NodeItem(  
        uid="yyyyyyyy-yyy-yyy-yyy-yyyyyyyyyyyy"  
    )  
]  
loginSpec = Login(  
    ssh_key="KeyPair-001"  
)  
specbody = RemoveNodesSpec(  
    login=loginSpec,  
    nodes=listNodesSpec  
)  
request.body = RemoveNodesTask(  
    spec=specbody  
)  
response = client.remove_node(request)  
print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

Removing a Node

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := cce.NewCceClient(  
        cce.CceClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.RemoveNodeRequest{  
        request.ClusterId = "{cluster_id}"  
    }  
    var listNodesSpec = []model.NodeItem{  
        {  
            Uid: "xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx",  
        },  
        {  
            Uid: "yyyyyyyy-yyy-yyy-yyy-yyyyyyyyyyyy",  
        },  
    }  
}
```

```
sshKeyLogin:= "KeyPair-001"
loginSpec := &model.Login{
    SshKey: &sshKeyLogin,
}
specbody := &model.RemoveNodesSpec{
    Login: loginSpec,
    Nodes: listNodesSpec,
}
request.Body = &model.RemoveNodesTask{
    Spec: specbody,
}
response, err := client.RemoveNode(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for removing a node in a specified cluster is successfully delivered.

Error Codes

See [Error Codes](#).

4.3.14 Migrating a Node

Function

This API is used to migrate a node from a specified cluster to another cluster. The node can be migrated only between different clusters in the same VPC and project, and the cluster types before and after the migration must be the same.

BMS nodes in a CCE Turbo cluster cannot be migrated.

NOTE

The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** indicates the resource path for API access.

Constraints

- Nodes can be migrated only between clusters in the same VPC and project.
- Nodes cannot be migrated between CCE Turbo clusters and CCE standard clusters.

- Nodes cannot be migrated between DeC and non-DeC clusters.
- Nodes cannot be migrated between clusters with IPv6 enabled and clusters without IPv6 enabled.
- Nodes can be migrated only to the default node pool of the target cluster.
- Nodes that have only one system disk attached cannot be migrated.
- Physical ECS nodes in a CCE Turbo cluster cannot be migrated.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/operation/migrateto/{target_cluster_id}

Table 4-542 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Parameter	Mandatory	Type	Description
target_cluster_id	Yes	String	Definition Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-543 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-544 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	No	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	No	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value MigrateNodesTask

Parameter	Mandatory	Type	Description
spec	Yes	MigrateNodeSpec object	Definition Configuration information Constraints N/A

Table 4-545 MigrateNodesSpec

Parameter	Mandatory	Type	Description
os	Yes	String	Definition OS type, which must be accurate to the version number, for example, Huawei Cloud EulerOS 2.0 . For details about the supported OSs, see Node OSs . Constraints If alpha.cce/NodeImageID is specified, the value of os must be the same as the OS of the private image. Range N/A Default Value N/A
extendParam	No	MigrateNodeExtendParam object	Definition Extended parameters for node migration Constraints N/A
login	No	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A

Parameter	Mandatory	Type	Description
runtime	No	Runtime object	Definition Container runtime Constraints N/A
serverConfig	No	MigrateServerConfig object	Definition Server settings Constraints N/A
nodes	Yes	Array of NodeItem objects	Definition List of nodes to be migrated. A maximum of 200 nodes can be migrated at a time. Constraints N/A

Table 4-546 MigrateNodeExtendParam

Parameter	Mandatory	Type	Description
maxPods	No	Integer	Definition The maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints N/A Range 16 to 256 Default Value N/A

Parameter	Mandatory	Type	Description
DockerLVMConfigOverride	No	String	Definition Docker data disk configuration items. The disk type of the node to be migrated must match the disk type specified at node creation (that is, the value of diskType in DockerLVMConfigOverride must be the same as that specified during node creation). Ensure that the disk types of the nodes selected for a single API call are the same as those specified during the node creation. Example of the default configuration: <pre>"DockerLVMConfigOverride": "dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"</pre> Constraints N/A Range The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default Value

Parameter	Mandatory	Type	Description
			N/A
alpha.cce/ preInstall	No	String	Definition Pre-installation script Constraints The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Range N/A Default Value N/A
alpha.cce/ postInstall	No	String	Definition Post-installation script Constraints The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
alpha.cce/ NodeImageID	No	String	Definition ID of the user image used by the target OS. NOTE If alpha.cce/NodeImageID is specified, the value of os must be the same as the OS of the private image. Constraints N/A Range N/A Default Value N/A

Table 4-547 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Mandatory	Type	Description
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-548 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-549 Runtime

Parameter	Mandatory	Type	Description
name	No	String	Details: Container runtime Constraints: None Options: N/A Default value: - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-550 MigrateServerConfig

Parameter	Mandatory	Type	Description
rootVolume	No	MigrateVolumeSpec object	Definition System disk reinstallation settings Constraints N/A

Table 4-551 MigrateVolumeSpec

Parameter	Mandatory	Type	Description
cmkID	No	String	Details: User master key ID. NOTE For details about how to obtain the key ID, see Querying the Key List . Constraints: None Options: N/A Default value: It is left empty by default, which indicates that the EVS disks are not encrypted.

Table 4-552 NodeItem

Parameter	Mandatory	Type	Description
uid	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Response Parameters

Status code: 200

Table 4-553 Response body parameters

Parameter	Type	Description
apiVersion	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value MigrateNodesTask
spec	MigrateNodesSpec object	Definition Configuration information Constraints N/A
status	TaskStatus object	Definition Task status Constraints N/A

Table 4-554 MigrateNodesSpec

Parameter	Type	Description
os	String	Definition OS type, which must be accurate to the version number, for example, Huawei Cloud EulerOS 2.0 . For details about the supported OSs, see Node OSs . Constraints If alpha.cce/NodeImageID is specified, the value of os must be the same as the OS of the private image. Range N/A Default Value N/A
extendParam	MigrateNodeExtendParam object	Definition Extended parameters for node migration Constraints N/A
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
runtime	Runtime object	Definition Container runtime Constraints N/A
serverConfig	MigrateServerConfig object	Definition Server settings Constraints N/A

Parameter	Type	Description
nodes	Array of NodeItem objects	Definition List of nodes to be migrated. A maximum of 200 nodes can be migrated at a time. Constraints N/A

Table 4-555 MigrateNodeExtendParam

Parameter	Type	Description
maxPods	Integer	Definition The maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints N/A Range 16 to 256 Default Value N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Definition Docker data disk configuration items. The disk type of the node to be migrated must match the disk type specified at node creation (that is, the value of diskType in DockerLVMConfigOverride must be the same as that specified during node creation). Ensure that the disk types of the nodes selected for a single API call are the same as those specified during the node creation. Example of the default configuration: <pre>"DockerLVMConfigOverride": "dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"</pre> Constraints N/A Range The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default Value N/A

Parameter	Type	Description
alpha.cce/ preInstall	String	Definition Pre-installation script Constraints The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. The input value must be encoded using Base64. The method is as follows: <code>echo -n "*"<content-to-be-encoded>" base64</code> Range N/A Default Value N/A
alpha.cce/ postInstall	String	Definition Post-installation script Constraints The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. The input value must be encoded using Base64. The method is as follows: <code>echo -n "*"<content-to-be-encoded>" base64</code> Range N/A Default Value N/A

Parameter	Type	Description
alpha.cce/ NodeImageID	String	Definition ID of the user image used by the target OS. NOTE If alpha.cce/NodeImageID is specified, the value of os must be the same as the OS of the private image. Constraints N/A Range N/A Default Value N/A

Table 4-556 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-557 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-558 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-559 MigrateServerConfig

Parameter	Type	Description
rootVolume	MigrateVolumeSpec object	Definition System disk reinstallation settings Constraints N/A

Table 4-560 MigrateVolumeSpec

Parameter	Type	Description
cmkID	String	Details: User master key ID. NOTE For details about how to obtain the key ID, see Querying the Key List . Constraints: None Options: N/A Default value: It is left empty by default, which indicates that the EVS disks are not encrypted.

Table 4-561 NodeItem

Parameter	Type	Description
uid	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Table 4-562 TaskStatus

Parameter	Type	Description
jobID	String	Definition Job ID, which is used by the caller to query the job progress. For details about how to obtain the value, see How to Obtain Parameters in the API URI. Constraints N/A Range N/A Default Value N/A

Example Requests

- Migrate a node running EulerOS 2.5 to another cluster.

```
PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/operation/migrateto/{target_cluster_id}
{
  "spec": {
    "os": "EulerOS 2.5",
    "login": {
      "sshKey": "KeyPair-001"
    },
    "nodes": [ {
      "uid": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
    }, {
      "uid": "yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy"
    } ]
  }
}
```
- Migrate a node running EulerOS 2.5 and created using a specified image ID to another cluster.

```
PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/operation/migrateto/{target_cluster_id}
{
  "spec": {
    "os": "EulerOS 2.5",
    "extendParam": {
      "alpha.cce/NodeImageID": "cc697ad7-9563-11e8-8ea7-0255ac106311"
    },
    "login": {
      "sshKey": "KeyPair-001"
    },
    "nodes": [ {
      "uid": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
    }, {
      "uid": "yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy"
    } ]
  }
}
```

Example Responses

Status code: 200

The job for migrating a node from a specified cluster to another cluster is successfully delivered.

```
{
  "spec" : {
    "os" : "EulerOS 2.5",
    "login" : {
      "sshKey" : "KeyPair-001"
    },
    "runtime" : {
      "name" : "docker"
    },
    "nodes" : [ {
      "uid" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
    }, {
      "uid" : "yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy"
    } ],
    "serverConfig" : {
      "rootVolume" : {
        "cmkID" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
      }
    }
  },
  "status" : {
    "jobID" : "2ec9b78d-9368-46f3-8f29-d1a95622a568"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Migrate a node running EulerOS 2.5 to another cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class MigrateNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
```

```
ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
MigrateNodeRequest request = new MigrateNodeRequest();
request.withClusterId("{cluster_id}");
request.withTargetClusterId("{target_cluster_id}");
MigrateNodesTask body = new MigrateNodesTask();
List<NodeItem> listSpecNodes = new ArrayList<>();
listSpecNodes.add(
    new NodeItem()
        .withUid("xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx")
);
listSpecNodes.add(
    new NodeItem()
        .withUid("yyyyyyyy-yyyy-yyyy-yyyyyyyyyyyy")
);
Login loginSpec = new Login();
loginSpec.withSshKey("KeyPair-001");
MigrateNodesSpec specbody = new MigrateNodesSpec();
specbody.withOs("EulerOS 2.5")
    .withLogin(loginSpec)
    .withNodes(listSpecNodes);
body.withSpec(specbody);
request.withBody(body);
try {
    MigrateNodeResponse response = client.migrateNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- Migrate a node running EulerOS 2.5 and created using a specified image ID to another cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class MigrateNodeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
    }
}
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();

MigrateNodeRequest request = new MigrateNodeRequest();
request.withClusterId("{cluster_id}");
request.withTargetClusterId("{target_cluster_id}");
MigrateNodesTask body = new MigrateNodesTask();
List<NodeItem> listSpecNodes = new ArrayList<>();
listSpecNodes.add(
    new NodeItem()
        .withUid("xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx")
);
listSpecNodes.add(
    new NodeItem()
        .withUid("yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy")
);
Login loginSpec = new Login();
loginSpec.withSshKey("KeyPair-001");
MigrateNodeExtendParam extendParamSpec = new MigrateNodeExtendParam();
extendParamSpec.withAlphaCceNodeImageID("cc697ad7-9563-11e8-8ea7-0255ac106311");
MigrateNodesSpec specbody = new MigrateNodesSpec();
specbody.withOs("EulerOS 2.5")
    .withExtendParam(extendParamSpec)
    .withLogin(loginSpec)
    .withNodes(listSpecNodes);
body.withSpec(specbody);
request.withBody(body);
try {
    MigrateNodeResponse response = client.migrateNode(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrMsg());
}
}
```

Python

- Migrate a node running EulerOS 2.5 to another cluster.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
```



```
# The AK and SK used for authentication are hard-coded, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
# In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = MigrateNodeRequest()
    request.cluster_id = "{cluster_id}"
    request.target_cluster_id = "{target_cluster_id}"
    listNodesSpec = [
        NodeItem(
            uid="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx"
        ),
        NodeItem(
            uid="yyyyyyyy-yyyy-yyyy-yyy-yyy-yyyyyyyyyyyyy"
        )
    ]
    loginSpec = Login(
        ssh_key="KeyPair-001"
    )
    specbody = MigrateNodesSpec(
        os="EulerOS 2.5",
        login=loginSpec,
        nodes=listNodesSpec
    )
    request.body = MigrateNodesTask(
        spec=specbody
    )
    response = client.migrate_node(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Migrate a node running EulerOS 2.5 and created using a specified image ID to another cluster.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
```

```
credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = MigrateNodeRequest()
    request.cluster_id = "{cluster_id}"
    request.target_cluster_id = "{target_cluster_id}"
    listNodesSpec = [
        NodeItem(
            uid="xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx"
        ),
        NodeItem(
            uid="yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyyy"
        )
    ]
    loginSpec = Login(
        ssh_key="KeyPair-001"
    )
    extendParamSpec = MigrateNodeExtendParam(
        alpha_cce_node_image_id="cc697ad7-9563-11e8-8ea7-0255ac106311"
    )
    specbody = MigrateNodesSpec(
        os="EulerOS 2.5",
        extend_param=extendParamSpec,
        login=loginSpec,
        nodes=listNodesSpec
    )
    request.body = MigrateNodesTask(
        spec=specbody
    )
    response = client.migrate_node(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- Migrate a node running EulerOS 2.5 to another cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
```

```
WithProjectId(projectId).
Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.MigrateNodeRequest{}
request.ClusterId = "{cluster_id}"
request.TargetClusterId = "{target_cluster_id}"
var listNodesSpec = []model.NodeItem{
    {
        Uid: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    },
    {
        Uid: "yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy",
    },
}
sshKeyLogin := "KeyPair-001"
loginSpec := &model.Login{
    SshKey: &sshKeyLogin,
}
specbody := &model.MigrateNodesSpec{
    Os: "EulerOS 2.5",
    Login: loginSpec,
    Nodes: listNodesSpec,
}
request.Body = &model.MigrateNodesTask{
    Spec: specbody,
}
response, err := client.MigrateNode(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Migrate a node running EulerOS 2.5 and created using a specified image ID to another cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("your-region")).
            WithCredential(auth).
            Build())

    request := &model.MigrateNodeRequest{}
    request.ClusterId = "{cluster_id}"
    request.TargetClusterId = "{target_cluster_id}"
    var listNodesSpec = []model.NodeItem{
        {
            Uid: "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
        },
        {
            Uid: "yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy",
        },
    }
    sshKeyLogin := "KeyPair-001"
    loginSpec := &model.Login{
        SshKey: &sshKeyLogin,
    }
    specbody := &model.MigrateNodesSpec{
        Os: "EulerOS 2.5",
        Login: loginSpec,
        Nodes: listNodesSpec,
    }
    request.Body = &model.MigrateNodesTask{
        Spec: specbody,
    }
    response, err := client.MigrateNode(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
client := cce.NewCceClient(  
    cce.CceClientBuilder().  
        WithRegion(region.ValueOf("<YOUR REGION>")).  
        WithCredential(auth).  
        Build())  
  
request := &model.MigrateNodeRequest{  
    request.ClusterId = "{cluster_id}"  
    request.TargetClusterId = "{target_cluster_id}"  
    var listNodesSpec = []model.NodeItem{  
        {  
            Uid: "xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",  
        },  
        {  
            Uid: "yyyyyyyy-yyy-yyy-yyy-yyyyyyyyyyyy",  
        },  
    }  
    sshKeyLogin:= "KeyPair-001"  
    loginSpec := &model.Login{  
        SshKey: &sshKeyLogin,  
    }  
    alphaCceNodeImageIDExtendParam:= "cc697ad7-9563-11e8-8ea7-0255ac106311"  
    extendParamSpec := &model.MigrateNodeExtendParam{  
        AlphaCceNodeImageID: &alphaCceNodeImageIDExtendParam,  
    }  
    specbody := &model.MigrateNodesSpec{  
        Os: "EulerOS 2.5",  
        ExtendParam: extendParamSpec,  
        Login: loginSpec,  
        Nodes: listNodesSpec,  
    }  
    request.Body = &model.MigrateNodesTask{  
        Spec: specbody,  
    }  
    response, err := client.MigrateNode(request)  
    if err == nil {  
        fmt.Printf("%v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for migrating a node from a specified cluster to another cluster is successfully delivered.

Error Codes

See [Error Codes](#).

4.3.15 Migrating a Node to a Custom Node Pool

Function

This API is used to migrate a node to a custom node pool. Only nodes in the default node pool can be migrated. Throughout the migration process, the node will remain unaffected and will not require resetting or restarting. Additionally, the original password will be preserved.

NOTE

The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** specifies the resource path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/nodes/migrate

Table 4-563 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-564 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-565 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Definition API version Constraints The value is fixed and cannot be changed. Range N/A Default Value v3
kind	Yes	String	Definition API type Constraints The value is fixed and cannot be changed. Range N/A Default Value List
nodeList	Yes	Array of MigrateNodesToNodePools objects	Definition List of nodes to be migrated. A maximum of 50 nodes can be migrated at a time. Constraints N/A

Table 4-566 MigrateNodesToNodePool

Parameter	Mandatory	Type	Description
nodeID	Yes	String	Definition Node ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Response Parameters

Status code: 200

The node has been migrated to the custom node pool.

None

Example Requests

Migrate a node from the default node pool to a custom node pool.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepool/{nodepool_id}/nodes/migrate
{
  "kind": "List",
  "apiVersion": "v3",
  "nodeList": [ {
    "nodeID": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx1"
  }, {
    "nodeID": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx2"
  } ]
}
```

Example Responses

None

Status Codes

Status Code	Description
200	The node has been migrated to the custom node pool.

Error Codes

See [Error Codes](#).

4.4 Node Pool Management

4.4.1 Creating a Node Pool

Function

This API is used to create a node pool in a specified cluster. This API can be called only when the cluster is in the available, scaling-out, or scaling-in state.

When creating a node pool in a CCE Turbo cluster of v1.21 or earlier, you can bind security groups to the node pool. A maximum of five security groups can be bound to a node pool.

After the security groups of a node pool are updated, the update takes effect only for newly created pods. You are advised to evict the original pods on the node.

NOTE

If there is no cluster, [create one](#).

The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** specifies the resource path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools

Table 4-567 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-568 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-569 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: NodePool
apiVersion	Yes	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
metadata	Yes	NodePoolMetadata object	Details: Metadata of a node pool Constraints: None

Parameter	Mandatory	Type	Description
spec	Yes	CreateNodePoolRequestSpec object	Details: Node pool specifications Constraints: This field is optional for updating a node pool.

Table 4-570 NodePoolMetadata

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Node pool name Constraints: You cannot create node pools named DefaultPool . Options: Enter 1 to 50 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	No	String	Details: UID of a node pool Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
annotations	No	Map<String,String>	Details: Annotations of a node pool in key-value pairs Constraints: This parameter is used only for query and cannot be input through a request. Entered data of this parameter is invalid.
updateTimestamp	No	String	Details: Time when a node pool was updated Constraints: The value is automatically recorded when a node pool is updated and cannot be specified. Options: N/A Default value: N/A
creationTimestamp	No	String	Details: Time when a node pool was created Constraints: The value is automatically recorded when a node pool is created and cannot be specified. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
resourceVersion	No	Integer	Definition Timestamp of the last update time of a node pool Constraints The value is automatically recorded when a node pool is created and cannot be specified. Range N/A Default Value N/A

Table 4-571 CreateNodePoolRequestSpec

Parameter	Mandatory	Type	Description
type	No	String	Details: Node pool type Constraints: None Options: • vm: ECS • ElasticBMS: C6 general computing-plus BMS. An example flavor is c6.22xlarge.2.physical. • pm: BMS Default value: vm
nodeTemplate	Yes	CreateNodePoolRequestSpecNodeSpec object	Details: Detailed parameters of a node pool template Constraints: None

Parameter	Mandatory	Type	Description
initialNodeCount	No	Integer	Details: The initial number of nodes in a node pool. The obtained value is the desired number of nodes in the node pool. Constraints: None Options: Values that are greater than or equal to 0 and less than or equal to the cluster scale Default value: 0
autoscaling	No	NodePoolNodeAutoscaling object	Details: Auto scaling parameters Constraints: None
nodeManagement	No	NodeManagement object	Details: Node management configuration Constraints: None
podSecurityGroups	No	Array of SecurityID objects	Details: Security group configuration Constraints: None
extensionScaleGroups	No	Array of ExtensionScaleGroup objects	Details: Configuration list of the extended scaling groups in a node pool. For details, see the definition of ExtensionScaleGroup . Constraints: None

Parameter	Mandatory	Type	Description
customSecurityGroups	No	Array of strings	Details: Custom security group configuration for a node pool. New nodes added to a node pool can be associated with a specified security group. • Specifying no security group ID will associate the new nodes with the default node security group. • Specifying a valid security group ID will associate new nodes with that security group. • When specifying a security group, do not modify the rules of the port on which CCE running depends. For details, see How Can I Configure a Security Group Rule in a Cluster? Constraints: None
taintPolicyOnExistingNodes	No	String	Definition: Taint synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The taints of existing nodes in a node pool will not be updated synchronously. • refresh: The taints of existing nodes in a node pool will be updated synchronously. Default value: refresh

Parameter	Mandatory	Type	Description
labelPolicyOnExistingNodes	No	String	Definition: Label synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The labels of existing nodes in a node pool will not be updated synchronously. • refresh: The labels of existing nodes in a node pool will be updated synchronously. Default value: refresh
userTagsPolicyOnExistingNodes	No	String	Definition: Resource tag synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The resource tags of existing nodes in a node pool will not be updated synchronously. • refresh: The resource tags of existing nodes in a node pool will be updated synchronously. Default value: ignore

Table 4-572 CreateNodePoolRequestSpecNodeSpec

Parameter	Mandatory	Type	Description
flavor	Yes	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A
az	Yes	String	Details: AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to a node created using APIs. For details about AZs supported by CCE, see Regions and Endpoints . Constraints: When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random . Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
os	No	String	Details: OS type of a node. For details about the supported OSs, see Node OSs. Constraints: • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node pool creation, the node pool will use private images. In this case, this parameter is optional. • If the alpha.cce/NodeImageID parameter in extendParam is not specified during node pool creation, this parameter is mandatory. • If the disk initialization parameter storage is used and the name field of virtualSpaces in StorageGroups is set to share during node pool creation, this parameter is mandatory. Options: N/A Default value: N/A
login	No	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A

Parameter	Mandatory	Type	Description
rootVolume	Yes	Volume object	Details: Disk information of a node Constraints: None
dataVolumes	No	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume . Constraints: • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left blank only when a system disk is selected for the system components.

Parameter	Mandatory	Type	Description
storage	No	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
nodeNicSpec	No	NodeNicSpec object	Details: ENI of a node Constraints: None

Parameter	Mandatory	Type	Description
publicIP	No	NodeEIPSpec object	Definition EIP parameters of a node Constraints N/A Range N/A Default Value N/A
count	No	Integer	NOTE The initialNodeCount parameter is used to initialize the number of nodes in a node pool during the node pool creation.
billingMode	No	Integer	Details: Billing mode of a node Constraints: None Options: • 0: pay-per-use • 1: yearly/monthly • 2: (deprecated) yearly/monthly billing with auto payment enabled Default value: N/A

Parameter	Mandatory	Type	Description
taints	No	Array of Taint objects	Details: Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints: A maximum of 20 taints can be configured.

Parameter	Mandatory	Type	Description
k8sTags	No	Map<String,String>	Details: Defined in key-value pairs. • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints: A maximum of 20 key-value pairs are allowed.

Parameter	Mandatory	Type	Description
ecsGroupId	No	String	Details: ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints: This configuration does not take effect when a node pool is created. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Options: N/A Default value: N/A
dedicatedHostId	No	String	Details: ID of the DeH to which nodes will be scheduled Constraints: This parameter is not supported when you add a node during node pool creation.
userTags	No	Array of UserTag objects	Details: Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints: The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.

Parameter	Mandatory	Type	Description
runtime	No	Runtime object	Details: Container runtime • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Mandatory	Type	Description
initializedConditions	No	Array of strings	Details: Custom initialization flag, which is left blank by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When creating a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their status is True, the node initialization is complete and the initialization taint is removed. 4. initializedConditions allows you to configure a timeout interval during

Parameter	Mandatory	Type	Description
			node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	No	NodeExtendParam object	Details: Extended parameters for creating a node Constraints: None
hostnameConfig	No	HostnameConfig object	Details: Kubernetes node name configuration parameter Constraints: The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Mandatory	Type	Description
serverEnterpriseProjectID	No	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
partition	No	String	Details: Zone that a node belongs to. It can be the central cloud or an edge site. Constraints: This field can be specified only for CCE Turbo clusters that support distributed clouds. Options: • center: central cloud • AZ ID of the edge site Default value: N/A

Parameter	Mandatory	Type	Description
nodeNameTemplate	No	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product . Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. Options: N/A Default value: N/A

Table 4-573 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-574 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-575 Volume

Parameter	Mandatory	Type	Description
size	Yes	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A
volumetype	Yes	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance . Default value: N/A

Parameter	Mandatory	Type	Description
extendParam	No	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks . Constraints: None
cluster_id	No	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID , that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool . Constraints: None Options: N/A Default value: N/A
cluster_type	No	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss . Default value: N/A

Parameter	Mandatory	Type	Description
hw:passthrough	No	Boolean	Details: • Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. • If the node pool type is ElasticBMS, this field must be set to true. • If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	No	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-576 VolumeMetadata

Parameter	Mandatory	Type	Description
<code>__system__encrypted</code>	No	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	No	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-577 Storage

Parameter	Mandatory	Type	Description
<code>storageSelectors</code>	Yes	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Mandatory	Type	Description
storageGroups	Yes	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-578 StorageSelectors

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
storageType	Yes	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	No	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-579 matchLabels

Parameter	Mandatory	Type	Description
size	No	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	No	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	No	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
metadataCmkid	No	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	No	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-580 StorageGroups

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: • If cceManaged is set to true, the name field must be set to vgpaas. • If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. • If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	No	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
selectorNames	Yes	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.
virtualSpaces	Yes	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-581 VirtualSpace

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	Yes	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Mandatory	Type	Description
lvmConfig	No	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	No	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-582 LVMConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
path	No	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-583 RuntimeConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-584 NodeNicSpec

Parameter	Mandatory	Type	Description
primaryNic	No	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	No	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-585 NicSpec

Parameter	Mandatory	Type	Description
subnetId	No	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	No	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Mandatory	Type	Description
ipBlock	No	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	No	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-586 NodeEIPSpec

Parameter	Mandatory	Type	Description
iptype	Yes	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	No	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-587 NodeBandwidth

Parameter	Mandatory	Type	Description
chargemode	No	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Mandatory	Type	Description
size	No	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	No	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-588 Taint

Parameter	Mandatory	Type	Description
key	Yes	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
value	No	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	Yes	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-589 UserTag

Parameter	Mandatory	Type	Description
key	No	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A
value	No	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, Default value: N/A

Table 4-590 Runtime

Parameter	Mandatory	Type	Description
name	No	String	Details: Container runtime Constraints: None Options: N/A Default value: - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-591 NodeExtendParam

Parameter	Mandatory	Type	Description
ecs:performance-type	No	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
orderId	No	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	No	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
maxPods	No	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node . Constraints: None Options: 16 to 256 Default value: N/A
periodType	No	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Mandatory	Type	Description
periodNum	No	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	No	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Mandatory	Type	Description
isAutoPay	No	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Mandatory	Type	Description
DockerLVMConfigOverride	No	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride":{"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Mandatory	Type	Description
<code>dockerBaseSize</code>	No	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Mandatory	Type	Description
containerBaseSize	No	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single

Parameter	Mandatory	Type	Description
			container is not limited by default.
publicKey	No	String	Definition Public key of a node, which is used for SSH key login Constraints • This parameter is invalid if you use a key pair to log in to the node. • This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	No	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "*Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
alpha.cce/postInstall	No	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "*Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	No	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	No	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
marketType	No	String	Details: When creating a spot ECS, this parameter should be set to spot . Constraints: This parameter is valid only when billingMode is set to 0 . Options: N/A Default value: N/A
spotPrice	No	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
agency_name	No	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	No	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	No	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
init-node-password	No	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-592 HostnameConfig

Parameter	Mandatory	Type	Description
type	Yes	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-593 nodeNameTemplate

Parameter	Mandatory	Type	Description
nodeNamePrefix	No	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	No	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Table 4-594 NodePoolNodeAutoscaling

Parameter	Mandatory	Type	Description
enable	No	Boolean	Details: Whether to enable auto scaling Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
minNodeCount	No	Integer	Details: Minimum number of nodes that can be scaled in if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	No	Integer	Details: Maximum number of nodes that can be added if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to the value of minNodeCount and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0

Parameter	Mandatory	Type	Description
scaleDownCooldownTime	No	Integer	Details: How long nodes will be kept, in minutes. During this period, nodes added will not be deleted. Constraints: None Options: 0 to 2147483646 Default value: 0
priority	No	Integer	Details: Weight of a node pool. A node pool with a higher weight has a higher priority during scale-out. Constraints: None Options: N/A Default value: 0

Parameter	Mandatory	Type	Description
scaleDownUnneededTime	No	Integer	Details: How long a node should be unneeded before it is eligible for scale in. This parameter specifies the duration during which a node is in the unneeded state before the scale-in operation can be performed. CCE Cluster Autoscaler scales in the node only when the node is not needed for a specified period. This prevents frequent scale-in triggered by temporary resource fluctuations and improves system stability. If this parameter is not specified, the CCE Cluster Autoscaler add-on uses the default time threshold. Constraints: None Options: 0 to 2147483647 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Parameter	Mandatory	Type	Description
scaleDownUtilizationThreshold	No	Float	Details: Scale-in utilization threshold of a node, which is defined by the total CPU or memory resources used by all running pods on the node divided by the total available resources of the node. If the calculated value falls below the specified threshold, the node becomes eligible for scale-in. For example, setting this parameter to 0.3 means that if the resource utilization on a node drops below 30%, the node will be scaled in. If this parameter is not specified, the CCE Cluster Autoscaler add-on will use the default utilization threshold. Constraints: None Options: 0 to 1 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Table 4-595 NodeManagement

Parameter	Mandatory	Type	Description
serverGroupReference	No	String	Details: A cloud server group ID. If this parameter is specified, the cloud server group will be associated with the target node pool, and all nodes in the node pool will be created in this cloud server group. After a cloud server group is associated with a node pool, the number of nodes in the node pool cannot exceed the number of cloud servers that can be added to the cloud server group, or the node pool cannot be scaled out. NOTE - After a cloud server group is associated with a node pool, the cloud servers added to the cloud server group will be distributed in strict accordance with the affinity policy and the maximum number of nodes in the node pool will be limited. The anti-affinity capabilities are available during cloud server creation on ECS. If your goal is to enhance service reliability by distributing cloud servers across different hosts and you do not want the number of nodes to be restricted by cloud server groups, you are advised not to associate cloud server groups with any node pools. - A cloud server group can be disassociated from the node pool. After the change, the existing nodes will remain in the original cloud server group, and no new nodes will be added to the cloud server group. If the node pool is empty, you can associate a new cloud server group with the node pool or change the associated cloud server group. Constraints: If a cloud server group is specified, the number of nodes

Parameter	Mandatory	Type	Description
			in the node pool cannot exceed the group quota. Options: • If this parameter is not specified or left blank, the cloud server group will be disassociated from the node pool. • A cloud server group ID: The cloud server group associated with the node pool will be changed to a specified one. Default value: N/A

Table 4-596 SecurityID

Parameter	Mandatory	Type	Description
id	No	String	Definition ID of the container security group on a node Constraints N/A Range N/A Default Value N/A

Table 4-597 ExtensionScaleGroup

Parameter	Mandatory	Type	Description
metadata	No	ExtensionScaleGroupMetadata object	Details: Basic information about an extended scaling group Constraints: None

Parameter	Mandatory	Type	Description
spec	No	ExtensionScaleGroupSpec object	Details: Configurations of an extended scaling group, which carry different configurations from those of the default scaling group Constraints: None

Table 4-598 ExtensionScaleGroupMetadata

Parameter	Mandatory	Type	Description
uid	No	String	Details: UUID of an extended scaling group, which is automatically generated by the system Constraints: None Options: N/A Default value: N/A
name	No	String	Details: Name of an extended scaling group Constraints: The value cannot be default . Options: Enter 1 to 56 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only digits, lowercase letters, and hyphens (-) are allowed. Default value: N/A

Table 4-599 ExtensionScaleGroupSpec

Parameter	Mandatory	Type	Description
flavor	No	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A
az	No	String	Details: Node AZ Constraints: None Options: N/A Default value: If this parameter is not specified or left blank, the default scaling group configurations take effect.
capacityReservationSpecification	No	CapacityReservationSpecification object	Details: Capacity reservation configurations of an extended scaling group Constraints: None
autoscaling	No	ScaleGroupAutoscaling object	Details: Auto scaling configurations of an extended scaling group Constraints: None

Table 4-600 CapacityReservationSpecification

Parameter	Mandatory	Type	Description
id	No	String	Details: Private pool ID Constraints: This value can be ignored when preference is set to none . Options: N/A Default value: N/A
preference	No	String	Details: Private pool capacity Constraints: If the value is none , the capacity reservation is not specified. If the value is targeted , the capacity reservation is specified. In this case, the ID cannot be left blank. Options: N/A Default value: none

Table 4-601 ScaleGroupAutoscaling

Parameter	Mandatory	Type	Description
enable	No	Boolean	Details: Whether to enable auto scaling for a scaling group Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
extensionPriority	No	Integer	Details: Scaling group priority. A larger value indicates a higher priority. Constraints: None Options: N/A Default value: 0
minNodeCount	No	Integer	Details: Minimum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0

Parameter	Mandatory	Type	Description
maxNodeCount	No	Integer	Details: Maximum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to the value of minNodeCount , less than or equal to the maximum number of nodes allowed by the cluster specifications, and less than or equal to the maximum number of nodes in the node pool Default value: 0

Response Parameters

Status code: 201

Table 4-602 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: NodePool

Parameter	Type	Description
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
metadata	NodePoolMetadata object	Details: Metadata of a node pool Constraints: None
spec	NodePoolSpec object	Details: Node pool specifications Constraints: This field is optional for updating a node pool.
status	CreateNodePoolsStatus object	Details: Node pool status Constraints: None

Table 4-603 NodePoolMetadata

Parameter	Type	Description
name	String	Details: Node pool name Constraints: You cannot create node pools named DefaultPool . Options: Enter 1 to 50 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A

Parameter	Type	Description
uid	String	Details: UID of a node pool Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A
annotations	Map<String,String>	Details: Annotations of a node pool in key-value pairs Constraints: This parameter is used only for query and cannot be input through a request. Entered data of this parameter is invalid.
updateTimestamp	String	Details: Time when a node pool was updated Constraints: The value is automatically recorded when a node pool is updated and cannot be specified. Options: N/A Default value: N/A
creationTimestamp	String	Details: Time when a node pool was created Constraints: The value is automatically recorded when a node pool is created and cannot be specified. Options: N/A Default value: N/A

Parameter	Type	Description
resourceVersion	Integer	Definition Timestamp of the last update time of a node pool Constraints The value is automatically recorded when a node pool is created and cannot be specified. Range N/A Default Value N/A

Table 4-604 NodePoolSpec

Parameter	Type	Description
type	String	Details: Node pool type Constraints: None Options: • vm: ECS • ElasticBMS: C6 general computing-plus BMS. An example flavor is c6.22xlarge.2.physical. • pm: BMS Default value: vm
nodeTemplate	NodeTemplate object	Details: Detailed parameters of a node pool template Constraints: None

Parameter	Type	Description
initialNodeCount	Integer	Details: The initial number of nodes in a node pool. The obtained value is the desired number of nodes in the node pool. Constraints: None Options: Values that are greater than or equal to 0 and less than or equal to the cluster scale Default value: 0
autoscaling	NodePoolNodeAutoscaling object	Details: Auto scaling parameters Constraints: None
nodeManagement	NodeManagement object	Details: Node management configuration Constraints: None
podSecurityGroups	Array of SecurityID objects	Details: Security group configuration Constraints: None
extensionScaleGroups	Array of ExtensionScaleGroup objects	Details: Configuration list of the extended scaling groups in a node pool. For details, see the definition of ExtensionScaleGroup . Constraints: None

Parameter	Type	Description
customSecurityGroups	Array of strings	Details: Custom security group configuration for a node pool. New nodes added to a node pool can be bound to a specified security group. • Specifying no security group ID will add the new nodes to the default node security group. • Specifying a valid security group ID will put new nodes in that security group. • When specifying a security group, do not modify the port rules on which CCE depends. For details, see How Can I Configure a Security Group Rule in a Cluster? Constraints: None
taintPolicyOnExistingNodes	String	Definition: Taint synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The taints of existing nodes in a node pool will not be updated synchronously. • refresh: The taints of existing nodes in a node pool will be updated synchronously. Default value: refresh

Parameter	Type	Description
labelPolicyOnExistingNodes	String	Definition: Label synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The labels of existing nodes in a node pool will not be updated synchronously. • refresh: The labels of existing nodes in a node pool will be updated synchronously. Default value: refresh
userTagsPolicyOnExistingNodes	String	Definition: Resource tag synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The resource tags of existing nodes in a node pool will not be updated synchronously. • refresh: The resource tags of existing nodes in a node pool will be updated synchronously. Default value: ignore

Table 4-605 NodeTemplate

Parameter	Type	Description
flavor	String	Definition Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints N/A Range N/A Default Value N/A
az	String	Definition AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to the nodes created using APIs. For details about AZs supported by CCE, see Regions and Endpoints . Constraints When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random . Range N/A Default Value N/A

Parameter	Type	Description
os	String	Definition OS type of a node. For details about the supported OSs, see Node OSs . Constraints • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Range N/A Default Value N/A
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Definition Disk information of a node Constraints N/A

Parameter	Type	Description
dataVolumes	Array of Volume objects	Definition Data disk parameters of a node. For DeC nodes, the parameter definitions are the same as those in rootVolume. Constraints • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left empty only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodeEIPSpec object	Definition EIP parameters of a node Constraints N/A Range N/A Default Value N/A
nodeNicSpec	NodeNicSpec object	Definition Network interfaces of a node Constraints N/A

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Definition Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints A maximum of 20 key-value pairs is allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Definition Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Definition Container runtime. • In clusters earlier than v1.25, the default value is docker. • In clusters of v1.25 or later, the default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints N/A

Parameter	Type	Description
initializedConditions	Array of strings	Definition Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When creating a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. Create a node and add "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions and add two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint will be removed. 4. initializedConditions allows you to configure a timeout interval during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions:

Parameter	Type	Description
		- type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	NodeExtendParam object	Definition Extended parameters for creating a node Constraints N/A
hostnameConfig	HostnameConfig object	Definition Kubernetes node name configuration parameter Constraints The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Definition Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints To create a node or node pool, you can specify an existing enterprise project for it. If this field is left empty, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Range N/A Default Value If no value is specified during the update, this field will not be updated. If this parameter is left empty, the cluster enterprise project is returned.
partition	String	Definition Zone that a node belongs to. It can be the central cloud or an edge site. Constraints This field can be specified only for CCE Turbo clusters that support distributed clouds. Range • center: central cloud • AZ ID of the edge site Default Value N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Definition Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(five-random-digits)-product . Constraints This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Range N/A Default Value N/A

Table 4-606 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-607 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-608 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-609 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-610 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-611 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-612 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-613 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-614 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-615 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-616 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-617 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-618 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE - Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. - Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: - If this field is not specified, the billing is based on bandwidth. - If the field value is empty, the billing is based on bandwidth. - If the field value is traffic, the billing is based on traffic. - If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-619 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-620 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-621 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-622 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-623 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-624 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <code>echo -n "**Content to be encoded*" base64</code> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/ NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-625 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-626 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A

Table 4-627 NodePoolNodeAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
minNodeCount	Integer	Details: Minimum number of nodes that can be scaled in if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be added if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to the value of minNodeCount and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0

Parameter	Type	Description
scaleDownCoolDownTime	Integer	Details: How long nodes will be kept, in minutes. During this period, nodes added will not be deleted. Constraints: None Options: 0 to 2147483646 Default value: 0
priority	Integer	Details: Weight of a node pool. A node pool with a higher weight has a higher priority during scale-out. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
scaleDownUnneededTime	Integer	Details: How long a node should be unneeded before it is eligible for scale in. This parameter specifies the duration during which a node is in the unneeded state before the scale-in operation can be performed. CCE Cluster Autoscaler scales in the node only when the node is not needed for a specified period. This prevents frequent scale-in triggered by temporary resource fluctuations and improves system stability. If this parameter is not specified, the CCE Cluster Autoscaler add-on uses the default time threshold. Constraints: None Options: 0 to 2147483647 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Parameter	Type	Description
scaleDownUtilizationThreshold	Float	Details: Scale-in utilization threshold of a node, which is defined by the total CPU or memory resources used by all running pods on the node divided by the total available resources of the node. If the calculated value falls below the specified threshold, the node becomes eligible for scale-in. For example, setting this parameter to 0.3 means that if the resource utilization on a node drops below 30%, the node will be scaled in. If this parameter is not specified, the CCE Cluster Autoscaler add-on will use the default utilization threshold. Constraints: None Options: 0 to 1 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Table 4-628 NodeManagement

Parameter	Type	Description
serverGroupReference	String	Details: A cloud server group ID. If this parameter is specified, the cloud server group will be associated with the target node pool, and all nodes in the node pool will be created in this cloud server group. After a cloud server group is associated with a node pool, the number of nodes in the node pool cannot exceed the number of cloud servers that can be added to the cloud server group, or the node pool cannot be scaled out. NOTE • After a cloud server group is associated with a node pool, the cloud servers added to the cloud server group will be distributed in strict accordance with the affinity policy and the maximum number of nodes in the node pool will be limited. The anti-affinity capabilities are available during cloud server creation on ECS. If your goal is to enhance service reliability by distributing cloud servers across different hosts and you do not want the number of nodes to be restricted by cloud server groups, you are advised not to associate cloud server groups with any node pools. • A cloud server group can be disassociated from the node pool. After the change, the existing nodes will remain in the original cloud server group, and no new nodes will be added to the cloud server group. If the node pool is empty, you can associate a new cloud server group with the node pool or change the associated cloud server group. Constraints: If a cloud server group is specified, the number of nodes in the node pool cannot exceed the group quota. Options: • If this parameter is not specified or left blank, the cloud server group will be disassociated from the node pool.

Parameter	Type	Description
		A cloud server group ID: The cloud server group associated with the node pool will be changed to a specified one. Default value: N/A

Table 4-629 SecurityID

Parameter	Type	Description
id	String	Definition ID of the container security group on a node Constraints N/A Range N/A Default Value N/A

Table 4-630 ExtensionScaleGroup

Parameter	Type	Description
metadata	ExtensionScaleGroupMetadata object	Details: Basic information about an extended scaling group Constraints: None
spec	ExtensionScaleGroupSpec object	Details: Configurations of an extended scaling group, which carry different configurations from those of the default scaling group Constraints: None

Table 4-631 ExtensionScaleGroupMetadata

Parameter	Type	Description
uid	String	Details: UUID of an extended scaling group, which is automatically generated by the system Constraints: None Options: N/A Default value: N/A
name	String	Details: Name of an extended scaling group Constraints: The value cannot be default . Options: Enter 1 to 56 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only digits, lowercase letters, and hyphens (-) are allowed. Default value: N/A

Table 4-632 ExtensionScaleGroupSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
az	String	Details: Node AZ Constraints: None Options: N/A Default value: If this parameter is not specified or left blank, the default scaling group configurations take effect.
capacityReservationSpecification	CapacityReservationSpecification object	Details: Capacity reservation configurations of an extended scaling group Constraints: None
autoscaling	ScaleGroupAutoscaling object	Details: Auto scaling configurations of an extended scaling group Constraints: None

Table 4-633 CapacityReservationSpecification

Parameter	Type	Description
id	String	Details: Private pool ID Constraints: This value can be ignored when preference is set to none . Options: N/A Default value: N/A

Parameter	Type	Description
preference	String	Details: Private pool capacity Constraints: If the value is none , the capacity reservation is not specified. If the value is targeted , the capacity reservation is specified. In this case, the ID cannot be left blank. Options: N/A Default value: none

Table 4-634 ScaleGroupAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling for a scaling group Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
extensionPriority	Integer	Details: Scaling group priority. A larger value indicates a higher priority. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
minNodeCount	Integer	Details: Minimum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to the value of minNodeCount , less than or equal to the maximum number of nodes allowed by the cluster specifications, and less than or equal to the maximum number of nodes in the node pool Default value: 0

Table 4-635 CreateNodePoolStatus

Parameter	Type	Description
currentNode	Integer	Details: Total number of nodes in the current node pool (excluding the nodes that are being deleted) Constraints: None Options: N/A Default value: N/A
creatingNode	Integer	Details: Number of nodes that are being created in the current node pool Constraints: None Options: N/A Default value: N/A
deletingNode	Integer	Details: Number of nodes that are being deleted in the current node pool Constraints: None Options: N/A Default value: N/A
configurationSyncedNodeCount	Integer	Details: The number of nodes specified in the node pool configuration has been synchronized in the current node pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Node pool status Constraints: None Options: • Null: The node pool is available. (The number of nodes in the current node pool has reached the expected value, and no node scaling is being performed.) • Synchronizing: The node pool is being scaled. (The number of nodes in the current node pool has not reached the expected value, and no node scaling is being performed.) • Synchronized: The node pool scaling is pending. (The number of nodes in the current node pool has not reached the expected value, or node scaling is being performed.) • SoldOut: The node pool cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The node pool is being deleted. • Error: An error occurred. NOTE The preceding node pool status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Node pool scaling status: You can obtain the status of the current node pool using parameters such as currentNode, creatingNode, or deletingNode. • Node pool scale-out: You can use conditions to obtain the detailed status of a node pool. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
conditions	Array of NodePoolCondition objects	Details: Node pool status details. For details, see the definition of Condition . Constraints: None
scaleGroupStatuses	Array of ScaleGroupStatus objects	Details: Detailed status of a scaling group. For details, see the definition of ScaleGroupStatus . Constraints: None

Table 4-636 ScaleGroupStatus

Parameter	Type	Description
name	String	Details: Scaling group name Constraints: None Options: N/A Default value: N/A
uid	String	Details: Scaling group UUID Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
creationTimestamp	String	Details: The time when a scaling group was created Constraints: None Options: N/A Default value: N/A
updateTimestamp	String	Details: The time when a scaling group was updated Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Scaling group status Constraints: None Options: • Null: The scaling group is available. (The number of nodes in the current scaling group has reached the expected value, and no node scaling is being performed.) • Synchronizing: The scaling group is being scaled. (The number of nodes in the current scaling group has not reached the expected value, and no node scaling is being performed.) • Synchronized: The scaling is pending. (The number of nodes in the current scaling group has not reached the expected value, or node scaling is being performed.) • SoldOut: The scaling group cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The scaling group is being deleted. • Error: An error occurred. NOTE The preceding scaling group status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Scaling status of a scaling group: You can obtain the status of the current scaling group using parameters such as desiredNodeCount, existingNodeCount, or upcomingNodeCount. • Scaling group scale-out: You can use conditions to obtain the detailed status of a scaling group. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
desiredNodeCount	Integer	Details: Expected number of nodes in a scaling group Constraints: None Options: N/A Default value: N/A
unpaidScaleNodeCount	Integer	Details: Number of nodes in an unpaid order Constraints: None Options: N/A Default value: N/A
existingNodeCount	existingNodeCount object	Details: Statistics on existing nodes in a scaling group Constraints: None
upcomingNodeCount	upcomingNodeCount object	Details: Statistics on the nodes to be added in a scaling group Constraints: None
scaleDownDisabledNodeCount	Integer	Details: Number of nodes that cannot be deleted in a scaling group Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
conditions	Array of NodePoolCondition objects	Details: Scaling group status details. For details, see the definition of Condition . Constraints: None

Table 4-637 existingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-638 upcomingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-639 NodePoolCondition

Parameter	Type	Description
type	String	Definition Status type Constraints N/A Range • TaintSynchronizing: The node pool is synchronizing Kubernetes taints for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • LabelSynchronizing: The node pool is synchronizing Kubernetes labels for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • UserTagsSynchronizing: The node pool is synchronizing Kubernetes resource tags for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • ConfigurationSynchronizing: The node pool is synchronizing settings for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • Scalable: specifies whether a node pool/scaling group can be scaled out. If the status is False, node pool scale-out will not be triggered again. • QuotaInsufficient: Quotas on which node pool/scaling group scale-out depends are insufficient, affecting the node pool scale-out status. • ResourceInsufficient: Resources on which node pool/scaling group

Parameter	Type	Description
		scale-out depends are insufficient, affecting the node pool scale-out status. • UnexpectedError: The node pool/scaling group fails to be scaled out due to unexpected reasons, affecting the node pool scale-out status. • LockedByOrder: The node pool/scaling group is locked due to a pending order. The value of Reason is the ID of the order. • Error: There is an error in the node pool/scaling group. The error is typically triggered by a deletion failure. Default Value N/A
status	String	Details: Status of the condition Constraints: None Options: • True • False Default value: N/A
lastProbeTime	String	Details: Time when the status was last checked Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
lastTransitTime	String	Details: Time when the status was last changed Constraints: None Options: N/A Default value: N/A
reason	String	Details: Reason why the status was last changed Constraints: None Options: N/A Default value: N/A
message	String	Details: Detailed description about the condition Constraints: None Options: N/A Default value: N/A

Example Requests

Create a node pool billed on a pay-per-use basis with the following configurations: The number of nodes is 0. The node pool flavor is 2 vCPUs and 4 GiB of memory. The node OS is EulerOS 2.5. Docker containers are deployed. The sizes of the system disk and data disk of the node are 40 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
{
  "kind": "NodePool",
  "apiVersion": "v3",
  "metadata": {
    "name": "lc-it-nodepool-79796"
  },
  "spec": {
    "initialNodeCount": 0,
```

```
"type" : "vm",
"autoscaling" : {
  "enable" : false,
  "minNodeCount" : 0,
  "maxNodeCount" : 1,
  "scaleDownCooldownTime" : 0,
  "priority" : 0,
  "scaleDownUnneededTime" : 5,
  "scaleDownUtilizationThreshold" : 0.6
},
"nodeManagement" : {
  "serverGroupReference" : ""
},
"nodeTemplate" : {
  "flavor" : "s6.large.2",
  "az" : "*****",
  "os" : "EulerOS 2.5",
  "login" : {
    "sshKey" : "KeyPair-001"
  },
  "rootVolume" : {
    "volumetype" : "SAS",
    "size" : 40
  },
  "dataVolumes" : [ {
    "volumetype" : "SAS",
    "size" : 100,
    "extendParam" : {
      "useType" : "docker"
    }
  } ],
  "billingMode" : 0,
  "extendParam" : {
    "alpha.cce/preInstall" : "",
    "alpha.cce/postInstall" : "",
    "alpha.cce/NodeImageID" : "",
    "maxPods" : 110
  },
  "nodeNicSpec" : {
    "primaryNic" : {
      "subnetId" : "7e767d10-7548-4df5-ad72-aeac1d08bd8a"
    }
  },
  "podSecurityGroups" : [ {
    "id" : ""
  } ]
}
```

Example Responses

Status code: 201

The job for creating a node pool in a specified cluster is successfully delivered.

```
{
  "kind" : "NodePool",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "lc-it-nodepool-79796",
    "uid" : "99addaa2-69eb-11ea-a592-0255ac1001bb"
  },
  "spec" : {
    "type" : "vm",
    "nodeTemplate" : {
      "flavor" : "s6.large.2",
      "az" : "*****",
      "os" : "EulerOS 2.5",
```

```
"login" : {
  "sshKey" : "KeyPair-001"
},
"rootVolume" : {
  "volumetype" : "SAS",
  "size" : 40
},
"dataVolumes" : [ {
  "volumetype" : "SAS",
  "size" : 100,
  "extendParam" : {
    "useType" : "docker"
  }
} ],
"publicIP" : {
  "eip" : {
    "bandwidth" : { }
  }
},
"nodeNicSpec" : {
  "primaryNic" : {
    "subnetId" : "7e767d10-7548-4df5-ad72-aeac1d08bd8a"
  }
},
"billingMode" : 0,
"extendParam" : {
  "alpha.cce/NodeImageID" : "",
  "alpha.cce/postInstall" : "",
  "alpha.cce/preInstall" : "",
  "maxPods" : 110
},
"k8sTags" : {
  "cce.cloud.com/cce-nodepool" : "lc-it-nodepool-79796"
},
"autoscaling" : {
  "maxNodeCount" : 1
},
"nodeManagement" : { }
},
"status" : {
  "phase" : ""
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Create a node pool billed on a pay-per-use basis with the following configurations: The number of nodes is 0. The node pool flavor is 2 vCPUs and 4 GiB of memory. The node OS is EulerOS 2.5. Docker containers are deployed. The sizes of the system disk and data disk of the node are 40 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
```

```
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class CreateNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateNodePoolRequest request = new CreateNodePoolRequest();
        request.withClusterId("{cluster_id}");
        NodePool body = new NodePool();
        List<SecurityID> listSpecPodSecurityGroups = new ArrayList<>();
        listSpecPodSecurityGroups.add(
            new SecurityID()
                .withId("")
        );
        NodeManagement nodeManagementSpec = new NodeManagement();
        nodeManagementSpec.withServerGroupReference("");
        NodePoolNodeAutoscaling autoscalingSpec = new NodePoolNodeAutoscaling();
        autoscalingSpec.withEnable(false)
            .withMinNodeCount(0)
            .withMaxNodeCount(1)
            .withScaleDownCooldownTime(0)
            .withPriority(0)
            .withScaleDownUnneededTime(5)
            .withScaleDownUtilizationThreshold(0.6f);
        NodeExtendParam extendParamNodeTemplate = new NodeExtendParam();
        extendParamNodeTemplate.withMaxPods(110)
            .withAlphaCcePreInstall("")
            .withAlphaCcePostInstall("")
            .withAlphaCceNodeImageID("");
        NicSpec primaryNicNodeNicSpec = new NicSpec();
        primaryNicNodeNicSpec.withSubnetId("7e767d10-7548-4df5-ad72-aeac1d08bd8a");
        NodeNicSpec nodeNicSpecNodeTemplate = new NodeNicSpec();
        nodeNicSpecNodeTemplate.withPrimaryNic(primaryNicNodeNicSpec);
        Map<String, Object> listDataVolumesExtendParam = new HashMap<>();
        listDataVolumesExtendParam.put("useType", "docker");
        List<Volume> listNodeTemplateDataVolumes = new ArrayList<>();
        listNodeTemplateDataVolumes.add(
            new Volume()
                .withSize(100)
                .withVolumetype("SAS")
                .withExtendParam(listDataVolumesExtendParam)
        );
        Volume rootVolumeNodeTemplate = new Volume();
        rootVolumeNodeTemplate.withSize(40)
            .withVolumetype("SAS");
        Login loginNodeTemplate = new Login();
        loginNodeTemplate.withSshKey("KeyPair-001");
```

```
NodeSpec nodeTemplateSpec = new NodeSpec();
nodeTemplateSpec.withFlavor("s6.large.2")
    .withAz("*****")
    .withOs("EulerOS 2.5")
    .withLogin(loginNodeTemplate)
    .withRootVolume(rootVolumeNodeTemplate)
    .withDataVolumes(listNodeTemplateDataVolumes)
    .withNodeNicSpec(nodeNicSpecNodeTemplate)
    .withBillingMode(0)
    .withExtendParam(extendParamNodeTemplate);
NodePoolSpec specbody = new NodePoolSpec();
specbody.withType(NodePoolSpec.TypeEnum.fromValue("vm"))
    .withNodeTemplate(nodeTemplateSpec)
    .withInitialNodeCount(0)
    .withAutoscaling(autoscalingSpec)
    .withNodeManagement(nodeManagementSpec)
    .withPodSecurityGroups(listSpecPodSecurityGroups);
NodePoolMetadata metadatabody = new NodePoolMetadata();
metadatabody.withName("lc-it-nodepool-79796");
body.withSpec(specbody);
body.withMetadata(metadatabody);
body.withApiVersion("v3");
body.withKind("NodePool");
request.withBody(body);
try {
    CreateNodePoolResponse response = client.createNodePool(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Create a node pool billed on a pay-per-use basis with the following configurations: The number of nodes is 0. The node pool flavor is 2 vCPUs and 4 GiB of memory. The node OS is EulerOS 2.5. Docker containers are deployed. The sizes of the system disk and data disk of the node are 40 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)
```

```
client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateNodePoolRequest()
    request.cluster_id = "{cluster_id}"
    listPodSecurityGroupsSpec = [
        SecurityID(
            id=""
        )
    ]
    nodeManagementSpec = NodeManagement(
        server_group_reference=""
    )
    autoscalingSpec = NodePoolNodeAutoscaling(
        enable=False,
        min_node_count=0,
        max_node_count=1,
        scale_down_cooldown_time=0,
        priority=0,
        scale_down_unneeded_time=5,
        scale_down_utilization_threshold=0.6
    )
    extendParamNodeTemplate = NodeExtendParam(
        max_pods=110,
        alpha_cce_pre_install="",
        alpha_cce_post_install="",
        alpha_cce_node_image_id=""
    )
    primaryNicNodeNicSpec = NicSpec(
        subnet_id="7e767d10-7548-4df5-ad72-aeac1d08bd8a"
    )
    nodeNicSpecNodeTemplate = NodeNicSpec(
        primary_nic=primaryNicNodeNicSpec
    )
    listExtendParamDataVolumes = {
        "useType": "docker"
    }
    listDataVolumesNodeTemplate = [
        Volume(
            size=100,
            volumetype="SAS",
            extend_param=listExtendParamDataVolumes
        )
    ]
    rootVolumeNodeTemplate = Volume(
        size=40,
        volumetype="SAS"
    )
    loginNodeTemplate = Login(
        ssh_key="KeyPair-001"
    )
    nodeTemplateSpec = NodeSpec(
        flavor="s6.large.2",
        az="*****",
        os="EulerOS 2.5",
        login=loginNodeTemplate,
        root_volume=rootVolumeNodeTemplate,
        data_volumes=listDataVolumesNodeTemplate,
        node_nic_spec=nodeNicSpecNodeTemplate,
        billing_mode=0,
        extend_param=extendParamNodeTemplate
    )
    specbody = NodePoolSpec(
        type="vm",
        node_template=nodeTemplateSpec,
```

```
        initial_node_count=0,
        autoscaling=autoscalingSpec,
        node_management=nodeManagementSpec,
        pod_security_groups=listPodSecurityGroupsSpec
    )
    metadatabody = NodePoolMetadata(
        name="lc-it-nodepool-79796"
    )
    request.body = NodePool(
        spec=specbody,
        metadata=metadatabody,
        api_version="v3",
        kind="NodePool"
    )
    response = client.create_node_pool(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Create a node pool billed on a pay-per-use basis with the following configurations: The number of nodes is 0. The node pool flavor is 2 vCPUs and 4 GiB of memory. The node OS is EulerOS 2.5. Docker containers are deployed. The sizes of the system disk and data disk of the node are 40 GB and 100 GB, respectively. Both the disks are of the high I/O type.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateNodePoolRequest{}
    request.ClusterId = "{cluster_id}"
    idPodSecurityGroups := ""
    var listPodSecurityGroupsSpec = []model.SecurityId{
        {
            Id: &idPodSecurityGroups,
```



```
    },
  }
  serverGroupReferenceNodeManagement:= ""
  nodeManagementSpec := &model.NodeManagement{
    ServerGroupReference: &serverGroupReferenceNodeManagement,
  }
  enableAutoscaling:= false
  minNodeCountAutoscaling:= int32(0)
  maxNodeCountAutoscaling:= int32(1)
  scaleDownCooldownTimeAutoscaling:= int32(0)
  priorityAutoscaling:= int32(0)
  scaleDownUnneededTimeAutoscaling:= int32(5)
  scaleDownUtilizationThresholdAutoscaling:= float32(0.6)
  autoscalingSpec := &model.NodePoolNodeAutoscaling{
    Enable: &enableAutoscaling,
    MinNodeCount: &minNodeCountAutoscaling,
    MaxNodeCount: &maxNodeCountAutoscaling,
    ScaleDownCooldownTime: &scaleDownCooldownTimeAutoscaling,
    Priority: &priorityAutoscaling,
    ScaleDownUnneededTime: &scaleDownUnneededTimeAutoscaling,
    ScaleDownUtilizationThreshold: &scaleDownUtilizationThresholdAutoscaling,
  }
  maxPodsExtendParam:= int32(110)
  alphaCcePreInstallExtendParam:= ""
  alphaCcePostInstallExtendParam:= ""
  alphaCceNodeImageIDExtendParam:= ""
  extendParamNodeTemplate := &model.NodeExtendParam{
    MaxPods: &maxPodsExtendParam,
    AlphaCcePreInstall: &alphaCcePreInstallExtendParam,
    AlphaCcePostInstall: &alphaCcePostInstallExtendParam,
    AlphaCceNodeImageID: &alphaCceNodeImageIDExtendParam,
  }
  subnetIdPrimaryNic:= "7e767d10-7548-4df5-ad72-aeac1d08bd8a"
  primaryNicNodeNicSpec := &model.NicSpec{
    SubnetId: &subnetIdPrimaryNic,
  }
  nodeNicSpecNodeTemplate := &model.NodeNicSpec{
    PrimaryNic: primaryNicNodeNicSpec,
  }
  var listExtendParamDataVolumes = map[string]interface{}{
    "useType": "docker",
  }
  var listDataVolumesNodeTemplate = []model.Volume{
    {
      Size: int32(100),
      Volumetype: "SAS",
      ExtendParam: listExtendParamDataVolumes,
    },
  },
  }
  rootVolumeNodeTemplate := &model.Volume{
    Size: int32(40),
    Volumetype: "SAS",
  }
  sshKeyLogin:= "KeyPair-001"
  loginNodeTemplate := &model.Login{
    SshKey: &sshKeyLogin,
  }
  osNodeTemplate:= "EulerOS 2.5"
  billingModeNodeTemplate:= int32(0)
  nodeTemplateSpec := &model.NodeSpec{
    Flavor: "s6.large.2",
    Az: "*****",
    Os: &osNodeTemplate,
    Login: loginNodeTemplate,
    RootVolume: rootVolumeNodeTemplate,
    DataVolumes: listDataVolumesNodeTemplate,
    NodeNicSpec: nodeNicSpecNodeTemplate,
    BillingMode: &billingModeNodeTemplate,
    ExtendParam: extendParamNodeTemplate,
```

```
}
typeSpec:= model.GetNodePoolSpecTypeEnum().VM
initialNodeCountSpec:= int32(0)
specbody := &model.NodePoolSpec{
    Type: &typeSpec,
    NodeTemplate: nodeTemplateSpec,
    InitialNodeCount: &initialNodeCountSpec,
    Autoscaling: autoscalingSpec,
    NodeManagement: nodeManagementSpec,
    PodSecurityGroups: &listPodSecurityGroupsSpec,
}
metadatabody := &model.NodePoolMetadata{
    Name: "lc-it-nodepool-79796",
}
request.Body = &model.NodePool{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "NodePool",
}
response, err := client.CreateNodePool(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	The job for creating a node pool in a specified cluster is successfully delivered.

Error Codes

See [Error Codes](#).

4.4.2 Reading a Specified Node Pool

Function

This API is used to obtain details about a specified node pool.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}

Table 4-640 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-641 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-642 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: NodePool
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
metadata	NodePoolMetadata object	Details: Metadata of a node pool Constraints: None Options: N/A Default value: N/A
spec	NodePoolSpec object	Definition Node pool specifications
status	NodePoolStatus object	Details: Node pool status Constraints: None Options: N/A Default value: N/A

Table 4-643 NodePoolMetadata

Parameter	Type	Description
name	String	Details: Node pool name Constraints: You cannot create node pools named DefaultPool . Options: Enter 1 to 50 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: UID of a node pool Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A
annotations	Map<String,String>	Details: Annotations of a node pool in key-value pairs Constraints: This parameter is used only for query and cannot be input through a request. Entered data of this parameter is invalid.

Parameter	Type	Description
updateTimestamp	String	Details: Time when a node pool was updated Constraints: The value is automatically recorded when a node pool is updated and cannot be specified. Options: N/A Default value: N/A
creationTimestamp	String	Details: Time when a node pool was created Constraints: The value is automatically recorded when a node pool is created and cannot be specified. Options: N/A Default value: N/A
resourceVersion	Integer	Definition Timestamp of the last update time of a node pool Constraints The value is automatically recorded when a node pool is created and cannot be specified. Range N/A Default Value N/A

Table 4-644 NodePoolSpec

Parameter	Type	Description
type	String	Details: Node pool type Constraints: None Options: • vm: ECS • ElasticBMS: C6 general computing-plus BMS. An example flavor is c6.22xlarge.2.physical. • pm: BMS Default value: vm
nodeTemplate	NodeTemplate object	Details: Detailed parameters of a node pool template Constraints: None
initialNodeCount	Integer	Details: The initial number of nodes in a node pool. The obtained value is the desired number of nodes in the node pool. Constraints: None Options: Values that are greater than or equal to 0 and less than or equal to the cluster scale Default value: 0
autoscaling	NodePoolNodeAutoscaling object	Details: Auto scaling parameters Constraints: None
nodeManagement	NodeManagement object	Details: Node management configuration Constraints: None

Parameter	Type	Description
podSecurityGroups	Array of SecurityID objects	Details: Security group configuration Constraints: None
extensionScaleGroups	Array of ExtensionScaleGroup objects	Details: Configuration list of the extended scaling groups in a node pool. For details, see the definition of ExtensionScaleGroup . Constraints: None
customSecurityGroups	Array of strings	Details: Custom security group configuration for a node pool. New nodes added to a node pool can be bound to a specified security group. • Specifying no security group ID will add the new nodes to the default node security group. • Specifying a valid security group ID will put new nodes in that security group. • When specifying a security group, do not modify the port rules on which CCE depends. For details, see How Can I Configure a Security Group Rule in a Cluster? Constraints: None

Parameter	Type	Description
taintPolicyOnExistingNodes	String	Definition: Taint synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The taints of existing nodes in a node pool will not be updated synchronously. • refresh: The taints of existing nodes in a node pool will be updated synchronously. Default value: refresh
labelPolicyOnExistingNodes	String	Definition: Label synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The labels of existing nodes in a node pool will not be updated synchronously. • refresh: The labels of existing nodes in a node pool will be updated synchronously. Default value: refresh

Parameter	Type	Description
userTagsPolicyOnExistingNodes	String	Definition: Resource tag synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The resource tags of existing nodes in a node pool will not be updated synchronously. • refresh: The resource tags of existing nodes in a node pool will be updated synchronously. Default value: ignore

Table 4-645 NodeTemplate

Parameter	Type	Description
flavor	String	Definition Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
az	String	Definition AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to the nodes created using APIs. For details about AZs supported by CCE, see Regions and Endpoints. Constraints When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random. Range N/A Default Value N/A
os	String	Definition OS type of a node. For details about the supported OSs, see Node OSs. Constraints • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Range N/A Default Value N/A

Parameter	Type	Description
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Definition Disk information of a node Constraints N/A
dataVolumes	Array of Volume objects	Definition Data disk parameters of a node. For DeC nodes, the parameter definitions are the same as those in rootVolume . Constraints • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left empty only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodeEIPSpec object	Definition EIP parameters of a node Constraints N/A Range N/A Default Value N/A
nodeNicSpec	NodeNicSpec object	Definition Network interfaces of a node Constraints N/A

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Definition Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints A maximum of 20 key-value pairs is allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Definition Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Definition Container runtime. • In clusters earlier than v1.25, the default value is docker. • In clusters of v1.25 or later, the default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints N/A

Parameter	Type	Description
initializedConditions	Array of strings	Definition Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When creating a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. Create a node and add "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions and add two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint will be removed. 4. initializedConditions allows you to configure a timeout interval during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions:

Parameter	Type	Description
		- type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	NodeExtendParam object	Definition Extended parameters for creating a node Constraints N/A
hostnameConfig	HostnameConfig object	Definition Kubernetes node name configuration parameter Constraints The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Definition Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints To create a node or node pool, you can specify an existing enterprise project for it. If this field is left empty, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Range N/A Default Value If no value is specified during the update, this field will not be updated. If this parameter is left empty, the cluster enterprise project is returned.
partition	String	Definition Zone that a node belongs to. It can be the central cloud or an edge site. Constraints This field can be specified only for CCE Turbo clusters that support distributed clouds. Range • center: central cloud • AZ ID of the edge site Default Value N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Definition Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(five-random-digits)-product . Constraints This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Range N/A Default Value N/A

Table 4-646 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-647 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-648 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss . Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-649 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-650 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-651 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-652 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-653 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-654 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-655 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-656 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-657 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-658 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-659 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-660 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-661 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-662 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-663 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-664 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-665 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-666 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Table 4-667 NodePoolNodeAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
minNodeCount	Integer	Details: Minimum number of nodes that can be scaled in if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be added if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to the value of minNodeCount and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0

Parameter	Type	Description
scaleDownCoolDownTime	Integer	Details: How long nodes will be kept, in minutes. During this period, nodes added will not be deleted. Constraints: None Options: 0 to 2147483646 Default value: 0
priority	Integer	Details: Weight of a node pool. A node pool with a higher weight has a higher priority during scale-out. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
scaleDownUnneededTime	Integer	Details: How long a node should be unneeded before it is eligible for scale in. This parameter specifies the duration during which a node is in the unneeded state before the scale-in operation can be performed. CCE Cluster Autoscaler scales in the node only when the node is not needed for a specified period. This prevents frequent scale-in triggered by temporary resource fluctuations and improves system stability. If this parameter is not specified, the CCE Cluster Autoscaler add-on uses the default time threshold. Constraints: None Options: 0 to 2147483647 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Parameter	Type	Description
scaleDownUtilizationThreshold	Float	Details: Scale-in utilization threshold of a node, which is defined by the total CPU or memory resources used by all running pods on the node divided by the total available resources of the node. If the calculated value falls below the specified threshold, the node becomes eligible for scale-in. For example, setting this parameter to 0.3 means that if the resource utilization on a node drops below 30%, the node will be scaled in. If this parameter is not specified, the CCE Cluster Autoscaler add-on will use the default utilization threshold. Constraints: None Options: 0 to 1 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Table 4-668 NodeManagement

Parameter	Type	Description
serverGroupReference	String	Details: A cloud server group ID. If this parameter is specified, the cloud server group will be associated with the target node pool, and all nodes in the node pool will be created in this cloud server group. After a cloud server group is associated with a node pool, the number of nodes in the node pool cannot exceed the number of cloud servers that can be added to the cloud server group, or the node pool cannot be scaled out. NOTE • After a cloud server group is associated with a node pool, the cloud servers added to the cloud server group will be distributed in strict accordance with the affinity policy and the maximum number of nodes in the node pool will be limited. The anti-affinity capabilities are available during cloud server creation on ECS. If your goal is to enhance service reliability by distributing cloud servers across different hosts and you do not want the number of nodes to be restricted by cloud server groups, you are advised not to associate cloud server groups with any node pools. • A cloud server group can be disassociated from the node pool. After the change, the existing nodes will remain in the original cloud server group, and no new nodes will be added to the cloud server group. If the node pool is empty, you can associate a new cloud server group with the node pool or change the associated cloud server group. Constraints: If a cloud server group is specified, the number of nodes in the node pool cannot exceed the group quota. Options: • If this parameter is not specified or left blank, the cloud server group will be disassociated from the node pool.

Parameter	Type	Description
		A cloud server group ID: The cloud server group associated with the node pool will be changed to a specified one. Default value: N/A

Table 4-669 SecurityID

Parameter	Type	Description
id	String	Definition ID of the container security group on a node Constraints N/A Range N/A Default Value N/A

Table 4-670 ExtensionScaleGroup

Parameter	Type	Description
metadata	ExtensionScaleGroupMetadata object	Details: Basic information about an extended scaling group Constraints: None
spec	ExtensionScaleGroupSpec object	Details: Configurations of an extended scaling group, which carry different configurations from those of the default scaling group Constraints: None

Table 4-671 ExtensionScaleGroupMetadata

Parameter	Type	Description
uid	String	Details: UUID of an extended scaling group, which is automatically generated by the system Constraints: None Options: N/A Default value: N/A
name	String	Details: Name of an extended scaling group Constraints: The value cannot be default . Options: Enter 1 to 56 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only digits, lowercase letters, and hyphens (-) are allowed. Default value: N/A

Table 4-672 ExtensionScaleGroupSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
az	String	Details: Node AZ Constraints: None Options: N/A Default value: If this parameter is not specified or left blank, the default scaling group configurations take effect.
capacityReservationSpecification	CapacityReservationSpecification object	Details: Capacity reservation configurations of an extended scaling group Constraints: None
autoscaling	ScaleGroupAutoscaling object	Details: Auto scaling configurations of an extended scaling group Constraints: None

Table 4-673 CapacityReservationSpecification

Parameter	Type	Description
id	String	Details: Private pool ID Constraints: This value can be ignored when preference is set to none . Options: N/A Default value: N/A

Parameter	Type	Description
preference	String	Details: Private pool capacity Constraints: If the value is none , the capacity reservation is not specified. If the value is targeted , the capacity reservation is specified. In this case, the ID cannot be left blank. Options: N/A Default value: none

Table 4-674 ScaleGroupAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling for a scaling group Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
extensionPriority	Integer	Details: Scaling group priority. A larger value indicates a higher priority. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
minNodeCount	Integer	Details: Minimum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to the value of minNodeCount , less than or equal to the maximum number of nodes allowed by the cluster specifications, and less than or equal to the maximum number of nodes in the node pool Default value: 0

Table 4-675 NodePoolStatus

Parameter	Type	Description
currentNode	Integer	Details: Total number of nodes in the current node pool (excluding the nodes that are being deleted) Constraints: None Options: N/A Default value: N/A
creatingNode	Integer	Details: Number of nodes that are being created in the current node pool Constraints: None Options: N/A Default value: N/A
deletingNode	Integer	Details: Number of nodes that are being deleted in the current node pool Constraints: None Options: N/A Default value: N/A
configurationSyncedNodeCount	Integer	Details: The number of nodes specified in the node pool configuration has been synchronized in the current node pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Node pool status Constraints: None Options: • Null: The node pool is available. (The number of nodes in the current node pool has reached the expected value, and no node scaling is being performed.) • Synchronizing: The node pool is being scaled. (The number of nodes in the current node pool has not reached the expected value, and no node scaling is being performed.) • Synchronized: The node pool scaling is pending. (The number of nodes in the current node pool has not reached the expected value, or node scaling is being performed.) • SoldOut: The node pool cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The node pool is being deleted. • Error: An error occurred. NOTE The preceding node pool status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Node pool scaling status: You can obtain the status of the current node pool using parameters such as currentNode, creatingNode, or deletingNode. • Node pool scale-out: You can use conditions to obtain the detailed status of a node pool. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
jobId	String	Details: ID of a job executed on the node pool. This field is available only when the node pool is being deleted. Constraints: None Options: N/A Default value: N/A
conditions	Array of NodePoolCondition objects	Details: Node pool status details. For details, see the definition of Condition . Constraints: None
scaleGroupStatuses	Array of ScaleGroupStatus objects	Details: Detailed status of a scaling group. For details, see the definition of ScaleGroupStatus . Constraints: None

Table 4-676 ScaleGroupStatus

Parameter	Type	Description
name	String	Details: Scaling group name Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
uid	String	Details: Scaling group UUID Constraints: None Options: N/A Default value: N/A
creationTimestamp	String	Details: The time when a scaling group was created Constraints: None Options: N/A Default value: N/A
updateTimestamp	String	Details: The time when a scaling group was updated Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Scaling group status Constraints: None Options: • Null: The scaling group is available. (The number of nodes in the current scaling group has reached the expected value, and no node scaling is being performed.) • Synchronizing: The scaling group is being scaled. (The number of nodes in the current scaling group has not reached the expected value, and no node scaling is being performed.) • Synchronized: The scaling is pending. (The number of nodes in the current scaling group has not reached the expected value, or node scaling is being performed.) • SoldOut: The scaling group cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The scaling group is being deleted. • Error: An error occurred. NOTE The preceding scaling group status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Scaling status of a scaling group: You can obtain the status of the current scaling group using parameters such as desiredNodeCount, existingNodeCount, or upcomingNodeCount. • Scaling group scale-out: You can use conditions to obtain the detailed status of a scaling group. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
desiredNodeCount	Integer	Details: Expected number of nodes in a scaling group Constraints: None Options: N/A Default value: N/A
unpaidScaleNodeCount	Integer	Details: Number of nodes in an unpaid order Constraints: None Options: N/A Default value: N/A
existingNodeCount	existingNodeCount object	Details: Statistics on existing nodes in a scaling group Constraints: None
upcomingNodeCount	upcomingNodeCount object	Details: Statistics on the nodes to be added in a scaling group Constraints: None
scaleDownDisabledNodeCount	Integer	Details: Number of nodes that cannot be deleted in a scaling group Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
conditions	Array of NodePoolCondition objects	Details: Scaling group status details. For details, see the definition of Condition . Constraints: None

Table 4-677 existingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-678 upcomingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-679 NodePoolCondition

Parameter	Type	Description
type	String	Definition Status type Constraints N/A Range • TaintSynchronizing: The node pool is synchronizing Kubernetes taints for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • LabelSynchronizing: The node pool is synchronizing Kubernetes labels for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • UserTagsSynchronizing: The node pool is synchronizing Kubernetes resource tags for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • ConfigurationSynchronizing: The node pool is synchronizing settings for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • Scalable: specifies whether a node pool/scaling group can be scaled out. If the status is False, node pool scale-out will not be triggered again. • QuotaInsufficient: Quotas on which node pool/scaling group scale-out depends are insufficient, affecting the node pool scale-out status. • ResourceInsufficient: Resources on which node pool/scaling group

Parameter	Type	Description
		scale-out depends are insufficient, affecting the node pool scale-out status. • UnexpectedError: The node pool/scaling group fails to be scaled out due to unexpected reasons, affecting the node pool scale-out status. • LockedByOrder: The node pool/scaling group is locked due to a pending order. The value of Reason is the ID of the order. • Error: There is an error in the node pool/scaling group. The error is typically triggered by a deletion failure. Default Value N/A
status	String	Details: Status of the condition Constraints: None Options: • True • False Default value: N/A
lastProbeTime	String	Details: Time when the status was last checked Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
lastTransitTime	String	Details: Time when the status was last changed Constraints: None Options: N/A Default value: N/A
reason	String	Details: Reason why the status was last changed Constraints: None Options: N/A Default value: N/A
message	String	Details: Detailed description about the condition Constraints: None Options: N/A Default value: N/A

Example Requests

None

Example Responses

Status code: 200

Details about the specified node pool are obtained successfully.

```
{
  "kind" : "NodePool",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "lc-it-nodepool-79796",
```

```
"uid" : "99addaa2-69eb-11ea-a592-0255ac1001bb"
},
"spec" : {
  "type" : "vm",
  "nodeTemplate" : {
    "flavor" : "s6.large.2",
    "az" : "*****",
    "os" : "EulerOS 2.5",
    "login" : {
      "sshKey" : "KeyPair-001"
    },
    "rootVolume" : {
      "volumetype" : "SAS",
      "size" : 40
    },
    "dataVolumes" : [ {
      "volumetype" : "SAS",
      "size" : 100,
      "extendParam" : {
        "useType" : "docker"
      }
    } ],
    "publicIP" : {
      "iptype" : "5_g-vm",
      "bandwidth" : {
        "Chargemode" : "bandwidth",
        "Size" : 5,
        "Sharetype" : "PER"
      }
    },
    "nodeNicSpec" : {
      "primaryNic" : {
        "subnetId" : "7e767d10-7548-4df5-ad72-aeac1d08bd8a"
      }
    },
    "billingMode" : 0,
    "extendParam" : {
      "maxPods" : 110
    },
    "k8sTags" : {
      "cce.cloud.com/cce-nodepool" : "lc-it-nodepool-79796"
    }
  },
  "autoscaling" : { },
  "nodeManagement" : { }
},
"status" : {
  "phase" : "Deleting",
  "jobId" : "3281fa02-69ee-11ea-a592-0255ac1001bb"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;
```

```
public class ShowNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowNodePoolRequest request = new ShowNodePoolRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        try {
            ShowNodePoolResponse response = client.showNodePool(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()
```



```
try:
    request = ShowNodePoolRequest()
    request.cluster_id = "{cluster_id}"
    request.nodepool_id = "{nodepool_id}"
    response = client.show_node_pool(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowNodePoolRequest{
        request.ClusterId = "{cluster_id}"
        request.NodepoolId = "{nodepool_id}"
    }
    response, err := client.ShowNodePool(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Details about the specified node pool are obtained successfully.

Error Codes

See [Error Codes](#).

4.4.3 Listing All Node Pools in a Specified Cluster

Function

This API is used to obtain information about all node pools in a specified cluster.

NOTE

- The URL for cluster management is in the format of **https://Endpoint/uri**. In the URL, **uri** indicates the resource path, that is, the path for API access.
- A node pool is a group of nodes within a cluster that all have the same configuration.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools

Table 4-680 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-681 Query Parameters

Parameter	Mandatory	Type	Description
showDefaultNodePool	No	String	Details: Whether to display the default node pool Constraints: N/A Options: If this parameter is set to true , the default node pool will be displayed. If this parameter is not specified, the default node pool will not be displayed. Default value: None

Request Parameters

Table 4-682 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-683 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: List
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
items	Array of NodePoolResp objects	Definition Node pool information

Table 4-684 NodePoolResp

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: NodePool

Parameter	Type	Description
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
metadata	NodePoolMetadata object	Details: Metadata of a node pool Constraints: None Options: N/A Default value: N/A
spec	NodePoolSpec object	Definition Node pool specifications
status	NodePoolStatus object	Details: Node pool status Constraints: None Options: N/A Default value: N/A

Table 4-685 NodePoolMetadata

Parameter	Type	Description
name	String	Details: Node pool name Constraints: You cannot create node pools named DefaultPool . Options: Enter 1 to 50 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: UID of a node pool Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A
annotations	Map<String,String>	Details: Annotations of a node pool in key-value pairs Constraints: This parameter is used only for query and cannot be input through a request. Entered data of this parameter is invalid.

Parameter	Type	Description
updateTimestamp	String	Details: Time when a node pool was updated Constraints: The value is automatically recorded when a node pool is updated and cannot be specified. Options: N/A Default value: N/A
creationTimestamp	String	Details: Time when a node pool was created Constraints: The value is automatically recorded when a node pool is created and cannot be specified. Options: N/A Default value: N/A
resourceVersion	Integer	Definition Timestamp of the last update time of a node pool Constraints The value is automatically recorded when a node pool is created and cannot be specified. Range N/A Default Value N/A

Table 4-686 NodePoolSpec

Parameter	Type	Description
type	String	Details: Node pool type Constraints: None Options: • vm: ECS • ElasticBMS: C6 general computing-plus BMS. An example flavor is c6.22xlarge.2.physical. • pm: BMS Default value: vm
nodeTemplate	NodeTemplate object	Details: Detailed parameters of a node pool template Constraints: None
initialNodeCount	Integer	Details: The initial number of nodes in a node pool. The obtained value is the desired number of nodes in the node pool. Constraints: None Options: Values that are greater than or equal to 0 and less than or equal to the cluster scale Default value: 0
autoscaling	NodePoolNodeAutoscaling object	Details: Auto scaling parameters Constraints: None
nodeManagement	NodeManagement object	Details: Node management configuration Constraints: None

Parameter	Type	Description
podSecurityGroups	Array of SecurityID objects	Details: Security group configuration Constraints: None
extensionScaleGroups	Array of ExtensionScaleGroup objects	Details: Configuration list of the extended scaling groups in a node pool. For details, see the definition of ExtensionScaleGroup . Constraints: None
customSecurityGroups	Array of strings	Details: Custom security group configuration for a node pool. New nodes added to a node pool can be bound to a specified security group. • Specifying no security group ID will add the new nodes to the default node security group. • Specifying a valid security group ID will put new nodes in that security group. • When specifying a security group, do not modify the port rules on which CCE depends. For details, see How Can I Configure a Security Group Rule in a Cluster? Constraints: None

Parameter	Type	Description
taintPolicyOnExistingNodes	String	Definition: Taint synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The taints of existing nodes in a node pool will not be updated synchronously. • refresh: The taints of existing nodes in a node pool will be updated synchronously. Default value: refresh
labelPolicyOnExistingNodes	String	Definition: Label synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The labels of existing nodes in a node pool will not be updated synchronously. • refresh: The labels of existing nodes in a node pool will be updated synchronously. Default value: refresh

Parameter	Type	Description
userTagsPolicyOnExistingNodes	String	Definition: Resource tag synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The resource tags of existing nodes in a node pool will not be updated synchronously. • refresh: The resource tags of existing nodes in a node pool will be updated synchronously. Default value: ignore

Table 4-687 NodeTemplate

Parameter	Type	Description
flavor	String	Definition Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
az	String	Definition AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to the nodes created using APIs. For details about AZs supported by CCE, see Regions and Endpoints. Constraints When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random. Range N/A Default Value N/A
os	String	Definition OS type of a node. For details about the supported OSs, see Node OSs. Constraints • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Range N/A Default Value N/A

Parameter	Type	Description
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Definition Disk information of a node Constraints N/A
dataVolumes	Array of Volume objects	Definition Data disk parameters of a node. For DeC nodes, the parameter definitions are the same as those in rootVolume . Constraints • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left empty only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodeEIPSpec object	Definition EIP parameters of a node Constraints N/A Range N/A Default Value N/A
nodeNicSpec	NodeNicSpec object	Definition Network interfaces of a node Constraints N/A

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Definition Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints A maximum of 20 key-value pairs is allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Definition Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Definition Container runtime. • In clusters earlier than v1.25, the default value is docker. • In clusters of v1.25 or later, the default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints N/A

Parameter	Type	Description
initializedConditions	Array of strings	Definition Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When creating a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. Create a node and add "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions and add two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint will be removed. 4. initializedConditions allows you to configure a timeout interval during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions:

Parameter	Type	Description
		- type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	NodeExtendParam object	Definition Extended parameters for creating a node Constraints N/A
hostnameConfig	HostnameConfig object	Definition Kubernetes node name configuration parameter Constraints The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Definition Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints To create a node or node pool, you can specify an existing enterprise project for it. If this field is left empty, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Range N/A Default Value If no value is specified during the update, this field will not be updated. If this parameter is left empty, the cluster enterprise project is returned.
partition	String	Definition Zone that a node belongs to. It can be the central cloud or an edge site. Constraints This field can be specified only for CCE Turbo clusters that support distributed clouds. Range • center: central cloud • AZ ID of the edge site Default Value N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Definition Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(five-random-digits)-product . Constraints This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Range N/A Default Value N/A

Table 4-688 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-689 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-690 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-691 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with __system__encrypted . Constraints: None Options: N/A Default value: N/A

Table 4-692 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-693 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-694 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-695 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: • If cceManaged is set to true, the name field must be set to vgpaas. • If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. • If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-696 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-697 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-698 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-699 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-700 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-701 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-702 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-703 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-704 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-705 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-706 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/ postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <code>echo -n "**Content to be encoded*" base64</code> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/ NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-707 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-708 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Table 4-709 NodePoolNodeAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
minNodeCount	Integer	Details: Minimum number of nodes that can be scaled in if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be added if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to the value of minNodeCount and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0

Parameter	Type	Description
scaleDownCoolDownTime	Integer	Details: How long nodes will be kept, in minutes. During this period, nodes added will not be deleted. Constraints: None Options: 0 to 2147483646 Default value: 0
priority	Integer	Details: Weight of a node pool. A node pool with a higher weight has a higher priority during scale-out. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
scaleDownUnneededTime	Integer	Details: How long a node should be unneeded before it is eligible for scale in. This parameter specifies the duration during which a node is in the unneeded state before the scale-in operation can be performed. CCE Cluster Autoscaler scales in the node only when the node is not needed for a specified period. This prevents frequent scale-in triggered by temporary resource fluctuations and improves system stability. If this parameter is not specified, the CCE Cluster Autoscaler add-on uses the default time threshold. Constraints: None Options: 0 to 2147483647 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Parameter	Type	Description
scaleDownUtilizationThreshold	Float	Details: Scale-in utilization threshold of a node, which is defined by the total CPU or memory resources used by all running pods on the node divided by the total available resources of the node. If the calculated value falls below the specified threshold, the node becomes eligible for scale-in. For example, setting this parameter to 0.3 means that if the resource utilization on a node drops below 30%, the node will be scaled in. If this parameter is not specified, the CCE Cluster Autoscaler add-on will use the default utilization threshold. Constraints: None Options: 0 to 1 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Table 4-710 NodeManagement

Parameter	Type	Description
serverGroupReference	String	Details: A cloud server group ID. If this parameter is specified, the cloud server group will be associated with the target node pool, and all nodes in the node pool will be created in this cloud server group. After a cloud server group is associated with a node pool, the number of nodes in the node pool cannot exceed the number of cloud servers that can be added to the cloud server group, or the node pool cannot be scaled out. NOTE • After a cloud server group is associated with a node pool, the cloud servers added to the cloud server group will be distributed in strict accordance with the affinity policy and the maximum number of nodes in the node pool will be limited. The anti-affinity capabilities are available during cloud server creation on ECS. If your goal is to enhance service reliability by distributing cloud servers across different hosts and you do not want the number of nodes to be restricted by cloud server groups, you are advised not to associate cloud server groups with any node pools. • A cloud server group can be disassociated from the node pool. After the change, the existing nodes will remain in the original cloud server group, and no new nodes will be added to the cloud server group. If the node pool is empty, you can associate a new cloud server group with the node pool or change the associated cloud server group. Constraints: If a cloud server group is specified, the number of nodes in the node pool cannot exceed the group quota. Options: • If this parameter is not specified or left blank, the cloud server group will be disassociated from the node pool.

Parameter	Type	Description
		A cloud server group ID: The cloud server group associated with the node pool will be changed to a specified one. Default value: N/A

Table 4-711 SecurityID

Parameter	Type	Description
id	String	Definition ID of the container security group on a node Constraints N/A Range N/A Default Value N/A

Table 4-712 ExtensionScaleGroup

Parameter	Type	Description
metadata	ExtensionScaleGroupMetadata object	Details: Basic information about an extended scaling group Constraints: None
spec	ExtensionScaleGroupSpec object	Details: Configurations of an extended scaling group, which carry different configurations from those of the default scaling group Constraints: None

Table 4-713 ExtensionScaleGroupMetadata

Parameter	Type	Description
uid	String	Details: UUID of an extended scaling group, which is automatically generated by the system Constraints: None Options: N/A Default value: N/A
name	String	Details: Name of an extended scaling group Constraints: The value cannot be default . Options: Enter 1 to 56 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only digits, lowercase letters, and hyphens (-) are allowed. Default value: N/A

Table 4-714 ExtensionScaleGroupSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
az	String	Details: Node AZ Constraints: None Options: N/A Default value: If this parameter is not specified or left blank, the default scaling group configurations take effect.
capacityReservationSpecification	CapacityReservationSpecification object	Details: Capacity reservation configurations of an extended scaling group Constraints: None
autoscaling	ScaleGroupAutoscaling object	Details: Auto scaling configurations of an extended scaling group Constraints: None

Table 4-715 CapacityReservationSpecification

Parameter	Type	Description
id	String	Details: Private pool ID Constraints: This value can be ignored when preference is set to none . Options: N/A Default value: N/A

Parameter	Type	Description
preference	String	Details: Private pool capacity Constraints: If the value is none , the capacity reservation is not specified. If the value is targeted , the capacity reservation is specified. In this case, the ID cannot be left blank. Options: N/A Default value: none

Table 4-716 ScaleGroupAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling for a scaling group Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
extensionPriority	Integer	Details: Scaling group priority. A larger value indicates a higher priority. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
minNodeCount	Integer	Details: Minimum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to the value of minNodeCount , less than or equal to the maximum number of nodes allowed by the cluster specifications, and less than or equal to the maximum number of nodes in the node pool Default value: 0

Table 4-717 NodePoolStatus

Parameter	Type	Description
currentNode	Integer	Details: Total number of nodes in the current node pool (excluding the nodes that are being deleted) Constraints: None Options: N/A Default value: N/A
creatingNode	Integer	Details: Number of nodes that are being created in the current node pool Constraints: None Options: N/A Default value: N/A
deletingNode	Integer	Details: Number of nodes that are being deleted in the current node pool Constraints: None Options: N/A Default value: N/A
configurationSyncedNodeCount	Integer	Details: The number of nodes specified in the node pool configuration has been synchronized in the current node pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Node pool status Constraints: None Options: • Null: The node pool is available. (The number of nodes in the current node pool has reached the expected value, and no node scaling is being performed.) • Synchronizing: The node pool is being scaled. (The number of nodes in the current node pool has not reached the expected value, and no node scaling is being performed.) • Synchronized: The node pool scaling is pending. (The number of nodes in the current node pool has not reached the expected value, or node scaling is being performed.) • SoldOut: The node pool cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The node pool is being deleted. • Error: An error occurred. NOTE The preceding node pool status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Node pool scaling status: You can obtain the status of the current node pool using parameters such as currentNode, creatingNode, or deletingNode. • Node pool scale-out: You can use conditions to obtain the detailed status of a node pool. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
jobId	String	Details: ID of a job executed on the node pool. This field is available only when the node pool is being deleted. Constraints: None Options: N/A Default value: N/A
conditions	Array of NodePoolCondition objects	Details: Node pool status details. For details, see the definition of Condition . Constraints: None
scaleGroupStatuses	Array of ScaleGroupStatus objects	Details: Detailed status of a scaling group. For details, see the definition of ScaleGroupStatus . Constraints: None

Table 4-718 ScaleGroupStatus

Parameter	Type	Description
name	String	Details: Scaling group name Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
uid	String	Details: Scaling group UUID Constraints: None Options: N/A Default value: N/A
creationTimestamp	String	Details: The time when a scaling group was created Constraints: None Options: N/A Default value: N/A
updateTimestamp	String	Details: The time when a scaling group was updated Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Scaling group status Constraints: None Options: • Null: The scaling group is available. (The number of nodes in the current scaling group has reached the expected value, and no node scaling is being performed.) • Synchronizing: The scaling group is being scaled. (The number of nodes in the current scaling group has not reached the expected value, and no node scaling is being performed.) • Synchronized: The scaling is pending. (The number of nodes in the current scaling group has not reached the expected value, or node scaling is being performed.) • SoldOut: The scaling group cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The scaling group is being deleted. • Error: An error occurred. NOTE The preceding scaling group status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Scaling status of a scaling group: You can obtain the status of the current scaling group using parameters such as desiredNodeCount, existingNodeCount, or upcomingNodeCount. • Scaling group scale-out: You can use conditions to obtain the detailed status of a scaling group. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
desiredNodeCount	Integer	Details: Expected number of nodes in a scaling group Constraints: None Options: N/A Default value: N/A
unpaidScaleNodeCount	Integer	Details: Number of nodes in an unpaid order Constraints: None Options: N/A Default value: N/A
existingNodeCount	existingNodeCount object	Details: Statistics on existing nodes in a scaling group Constraints: None
upcomingNodeCount	upcomingNodeCount object	Details: Statistics on the nodes to be added in a scaling group Constraints: None
scaleDownDisabledNodeCount	Integer	Details: Number of nodes that cannot be deleted in a scaling group Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
conditions	Array of NodePoolCondition objects	Details: Scaling group status details. For details, see the definition of Condition . Constraints: None

Table 4-719 existingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-720 upcomingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-721 NodePoolCondition

Parameter	Type	Description
type	String	Definition Status type Constraints N/A Range • TaintSynchronizing: The node pool is synchronizing Kubernetes taints for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • LabelSynchronizing: The node pool is synchronizing Kubernetes labels for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • UserTagsSynchronizing: The node pool is synchronizing Kubernetes resource tags for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • ConfigurationSynchronizing: The node pool is synchronizing settings for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • Scalable: specifies whether a node pool/scaling group can be scaled out. If the status is False, node pool scale-out will not be triggered again. • QuotaInsufficient: Quotas on which node pool/scaling group scale-out depends are insufficient, affecting the node pool scale-out status. • ResourceInsufficient: Resources on which node pool/scaling group

Parameter	Type	Description
		scale-out depends are insufficient, affecting the node pool scale-out status. • UnexpectedError: The node pool/scaling group fails to be scaled out due to unexpected reasons, affecting the node pool scale-out status. • LockedByOrder: The node pool/scaling group is locked due to a pending order. The value of Reason is the ID of the order. • Error: There is an error in the node pool/scaling group. The error is typically triggered by a deletion failure. Default Value N/A
status	String	Details: Status of the condition Constraints: None Options: • True • False Default value: N/A
lastProbeTime	String	Details: Time when the status was last checked Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
lastTransitTime	String	Details: Time when the status was last changed Constraints: None Options: N/A Default value: N/A
reason	String	Details: Reason why the status was last changed Constraints: None Options: N/A Default value: N/A
message	String	Details: Detailed description about the condition Constraints: None Options: N/A Default value: N/A

Example Requests

None

Example Responses

Status code: 200

Information about all node pools in the cluster is successfully obtained.

```
{
  "kind" : "List",
  "apiVersion" : "v3",
  "items" : [ {
    "kind" : "NodePool",
```

```
"apiVersion" : "v3",
"metadata" : {
  "name" : "az1.dc1#s1.large#EulerOS 2.2",
  "uid" : "az1.dc1#s1.large#EulerOS 2.2"
},
"spec" : {
  "nodeTemplate" : {
    "flavor" : "s1.large",
    "az" : "az1.dc1",
    "os" : "EulerOS 2.2",
    "login" : {
      "sshKey" : "KeyPair-001"
    },
    "rootVolume" : { },
    "publicIP" : {
      "iptype" : "5_g-vm",
      "bandwidth" : {
        "Chargemode" : "bandwidth",
        "Size" : 5,
        "Sharetype" : "PER"
      }
    },
    "billingMode" : 0
  },
  "autoscaling" : {
    "enable" : true,
    "maxNodeCount" : 50
  },
  "status" : {
    "currentNode" : 1
  }
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListNodePoolsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
```



```
.withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ListNodePoolsRequest request = new ListNodePoolsRequest();
request.withClusterId("{cluster_id}");
try {
    ListNodePoolsResponse response = client.listNodePools(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListNodePoolsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_node_pools(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListNodePoolsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ListNodePools(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Information about all node pools in the cluster is successfully obtained.

Error Codes

See [Error Codes](#).

4.4.4 Updating a Specified Node Pool

Function

This API is used to update information about a specified node pool. This API can be called only when the cluster is in the Available, Scale Out, or Scale In state.

NOTE

- The URL for cluster management is in the following format: `https://Endpoint/uri`. `uri` indicates the resource path, that is, the API access path.
- Currently, only the node pool name can be updated. `initialNodeCount`, `k8sTags`, `taints`, `login` and `userTags` under `spec` are related to the scaling configuration of the node pool. If no value is set for this update, the initial value is used by default.

Calling Method

For details, see [Calling APIs](#).

URI

PUT `/api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}`

Table 4-722 Path Parameters

Parameter	Mandatory	Type	Description
<code>project_id</code>	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-723 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-724 Request body parameters

Parameter	Mandatory	Type	Description
metadata	No	NodePoolMetadataUpdate object	Details: Metadata of a node pool Constraints: None Options: N/A Default value: N/A
spec	No	NodePoolSpecUpdate object	Details: Node pool specifications Constraints: None Options: N/A Default value: N/A

Table 4-725 NodePoolMetadataUpdate

Parameter	Mandatory	Type	Description
name	No	String	Details: Node pool name Constraints: You are not allowed to create node pools named DefaultPool . Options: Enter 1 to 50 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A

Table 4-726 NodePoolSpecUpdate

Parameter	Mandatory	Type	Description
nodeTemplate	No	NodeSpecUpdate object	Details: Detailed parameters of a node pool template Constraints: None Options: N/A Default value: N/A
initialNodeCount	Yes	Integer	Details: Desired number of nodes in a node pool Constraints: This field is mandatory for updating a node pool. NOTE Note that if this field is not specified when updating a node pool, the expected number of nodes in the node pool will default to 0. If there are more nodes in the node pool than this number, the excess nodes will be scaled in. Options: Greater than 0 and less than the number of nodes that can be managed by the cluster Default value: 0

Parameter	Mandatory	Type	Description
ignoreInitialNodeCount	No	Boolean	Details: This parameter controls the default settings of spec.initialNodeCount (the expected number of nodes in a node pool) when a node pool is updated. If this parameter is not specified or set to false and spec.initialNodeCount is not specified in the request body, spec.initialNodeCount is automatically initialized to 0 during the updates. If this parameter is set to true, the spec.initialNodeCount parameter will be ignored. NOTE If there is no need to update spec.initialNodeCount of a node pool, this parameter should be set to true, and spec.initialNodeCount should not be specified in the body of the node pool to be updated. If the value of spec.initialNodeCount of a node pool is not 0, nodes in the node pool will be scaled in. Constraints: None Options: • false: If the spec.initialNodeCount parameter is not specified when a node pool is updated, the value of spec.initialNodeCount is initialized to 0. • true: When updating a node pool, the spec.initialNodeCount parameter is ignored, and the spec.initialNodeCount parameter settings of the node pool remain unchanged. Default value: false

Parameter	Mandatory	Type	Description
autoscaling	No	NodePoolNodeAutoscaling object	Details: Auto scaling, which is supported only by pay-per-use node pools Constraints: None Options: N/A Default value: N/A
nodeManagementUpdate	No	NodeManagement object	Details: Node management configuration Constraints: None Options: N/A Default value: N/A
taintPolicyOnExistingNodes	No	String	Details: Whether to synchronize Kubernetes taints Constraints: None Options: • refresh: Kubernetes taint modifications will be automatically synchronized to existing nodes. • ignore: Kubernetes taint modifications will not be automatically synchronized to existing nodes. Default value: None

Parameter	Mandatory	Type	Description
labelPolicyOnExistingNodes	No	String	Details: Whether to synchronize Kubernetes labels Constraints: None Options: • refresh: Kubernetes label modifications will be automatically synchronized to existing nodes. • ignore: Kubernetes label modifications will not be automatically synchronized to existing nodes. Default value: None
userTagsPolicyOnExistingNodes	No	String	Details: Whether to synchronize node pool tags Constraints: None Options: • refresh: Node pool tag modifications will be automatically synchronized to existing nodes. • ignore: Node pool tag modifications will not be automatically synchronized to existing nodes. Default value: None

Parameter	Mandatory	Type	Description
extensionScaleGroups	No	Array of ExtensionScaleGroup objects	Details: Configuration list of the extended scaling groups in a node pool. For details, see the definition of ExtensionScaleGroup . Constraints: None Options: N/A Default value: N/A

Table 4-727 NodeSpecUpdate

Parameter	Mandatory	Type	Description
flavor	No	String	Definition Node flavor Constraints The node flavor cannot be changed. NOTE This parameter can be left empty only during the deletion of the default scaling group of a node pool. The default scaling group can be deleted only when the az field is also left empty. If no other scaling group is added to the node pool, the default scaling group cannot be deleted. Range For details about the node flavors supported by CCE clusters, see Node Specifications . Default Value N/A

Parameter	Mandatory	Type	Description
az	No	String	Definition AZ where a node is located. The AZ name must be specified. Constraints The AZ of the node cannot be changed. NOTE This parameter can be left empty only during the deletion of the default scaling group of a node pool. The default scaling group can be deleted only when the flavor field is also left empty. If no other scaling group is added to the node pool, the default scaling group cannot be deleted. Range For details about AZs supported by CCE, see Regions and Endpoints. Default Value N/A

Parameter	Mandatory	Type	Description
os	No	String	Details: Node OS. For details, see Node OSs . Constraints: • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If alpha.cce/NodeImageID in extendParam is specified during node creation, you do not need to configure this parameter. • If this parameter is not specified, CCE will automatically select a supported OS version based on the cluster version. • This parameter is mandatory when creating a node pool. • This parameter is mandatory if the disk initialization parameter storage is used and the name field of virtualSpaces in StorageGroups is set to share during node creation. Options: N/A Default value: N/A
login	No	Login object	Definition How to log in to a node Constraints N/A
rootVolumeUpdate	No	Volume object	Details: Disk information of a node Constraints: None

Parameter	Mandatory	Type	Description
dataVolumes Update	No	Array of Volume objects	Details: Data disk parameters of a node. For DeC nodes, the parameter details are the same as those in rootVolume . Constraints: A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable.

Parameter	Mandatory	Type	Description
storage	No	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node. Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.

Parameter	Mandatory	Type	Description
runtime	No	Runtime object	Details: Container runtime - Clusters earlier than v1.25: The default value is docker. - Clusters v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints: None

Parameter	Mandatory	Type	Description
taints	No	Array of Taint objects	Definition Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints • A maximum of 20 taints can be configured. • If this parameter is not specified, the custom taints

Parameter	Mandatory	Type	Description
			of the node pool will not be updated. • If the parameter is left empty, the custom taints of the node pool will be deleted.
waitPostInstallFinish	No	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node pool updates. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Mandatory	Type	Description
k8sTags	No	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints • A maximum of 20 key-value pairs is allowed. • If this parameter is not specified, the custom Kubernetes labels of the node pool will not be updated. • If the parameter is left empty, the custom Kubernetes labels of the node pool will be deleted.

Parameter	Mandatory	Type	Description
userTags	No	Array of UserTag objects	Definition Cloud server resource tags Constraints • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8. • If this parameter is not specified, the custom cloud server resource tags of the node pool will not be updated. • If the parameter is left empty, the custom cloud server resource tags of the node pool will be deleted.

Parameter	Mandatory	Type	Description
nodeNameTemplate	No	nodeNameTemplate object	Details: Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(Five random digits)-product. Constraints: This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
initializedConditions	No	Array of strings	Details: Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When creating a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. Create a node and add parameter "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. Update a node and add parameter "initializedConditions": ["NodeInitial"]. By default, when a new node in the node pool is added to the cluster, the node does not accept pods from other nodes. 3. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions by adding two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 4. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or

Parameter	Mandatory	Type	Description
			CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint is removed. 5. initializedConditions allows you to configure a timeout interval during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions: - type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints: • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).

Parameter	Mandatory	Type	Description
serverEnterpriseProjectID	No	String	Details: Enterprise project ID of a server. CCE does not support features provided by EPS. This field is only used for synchronizing enterprise project IDs of servers. Constraints: To create a node or node pool, you can specify an existing enterprise project for it. If this field is left blank, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Options: N/A Default value: If no value is specified during the update, this field will not be updated. If this parameter is left blank, the cluster enterprise project is returned.
nodeNicSpecUpdate	No	nodeNicSpecUpdate object	Details: Update the ENI on a node. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
extendParam	No	NodePoolUpdateExtendParam object	Details: Extended parameters supported when a node pool is updated Constraints: None Options: N/A Default value: N/A
publicIP	No	NodeEIPSpec object	Definition EIP parameters of a node Constraints To delete this parameter, transfer an empty object. Range N/A Default Value N/A

Table 4-728 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Mandatory	Type	Description
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-729 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-730 Volume

Parameter	Mandatory	Type	Description
size	Yes	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A
volumetype	Yes	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance . Default value: N/A

Parameter	Mandatory	Type	Description
extendParam	No	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks . Constraints: None
cluster_id	No	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID , that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool . Constraints: None Options: N/A Default value: N/A
cluster_type	No	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss . Default value: N/A

Parameter	Mandatory	Type	Description
hw:passthrough	No	Boolean	Details: • Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. • If the node pool type is ElasticBMS, this field must be set to true. • If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	No	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-731 VolumeMetadata

Parameter	Mandatory	Type	Description
<code>__system__encrypted</code>	No	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	No	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-732 Storage

Parameter	Mandatory	Type	Description
<code>storageSelectors</code>	Yes	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Mandatory	Type	Description
storageGroups	Yes	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-733 StorageSelectors

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
storageType	Yes	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	No	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-734 matchLabels

Parameter	Mandatory	Type	Description
size	No	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	No	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	No	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
metadataCmkid	No	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	No	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-735 StorageGroups

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: • If cceManaged is set to true, the name field must be set to vgpaas. • If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. • If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	No	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
selectorNames	Yes	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.
virtualSpaces	Yes	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-736 VirtualSpace

Parameter	Mandatory	Type	Description
name	Yes	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	Yes	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Mandatory	Type	Description
lvmConfig	No	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	No	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-737 LVMConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
path	No	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-738 RuntimeConfig

Parameter	Mandatory	Type	Description
lvType	Yes	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-739 Runtime

Parameter	Mandatory	Type	Description
name	No	String	Details: Container runtime Constraints: None Options: N/A Default value: - Clusters earlier than v1.25: The default value is docker. - Clusters of v1.25 or later: The default value varies with the OSs. - For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-740 Taint

Parameter	Mandatory	Type	Description
key	Yes	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
value	No	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	Yes	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-741 UserTag

Parameter	Mandatory	Type	Description
key	No	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, / Default value: N/A
value	No	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0–31) or the following special characters: =*<>, Default value: N/A

Table 4-742 nodeNameTemplate

Parameter	Mandatory	Type	Description
nodeNamePrefix	No	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	No	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Table 4-743 nodeNicSpecUpdate

Parameter	Mandatory	Type	Description
primaryNic	No	primaryNic object	Details: Description of the primary ENI Constraints: None Options: N/A Default value: N/A

Table 4-744 primaryNic

Parameter	Mandatory	Type	Description
subnetId	No	String	Details: Network ID of the subnet to which an ENI belongs Constraints: If subnetId is not specified when a primary ENI is created, the cluster subnet will be used. If subnetList is specified for a node pool, the subnetList field is used for adding subnets to the node pool. When creating an extended ENI, you must specify subnetId . Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
subnetList	No	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured. Options: N/A Default value: N/A

Table 4-745 NodePoolUpdateExtendParam

Parameter	Mandatory	Type	Description
agency_name	No	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. Constraints: This parameter is returned only when it is transferred during node creation. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
alpha.cce/ preInstall	No	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/ postInstall	No	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
spotPrice	No	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and be greater than or equal to the ECS market price. Options: N/A Default value: N/A
alpha.cce/ NodeImageID	No	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A

Table 4-746 NodeEIPSpec

Parameter	Mandatory	Type	Description
iptype	Yes	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	No	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-747 NodeBandwidth

Parameter	Mandatory	Type	Description
chargemode	No	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A

Parameter	Mandatory	Type	Description
size	No	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
sharetype	No	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-748 NodePoolNodeAutoscaling

Parameter	Mandatory	Type	Description
enable	No	Boolean	Details: Whether to enable auto scaling Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false

Parameter	Mandatory	Type	Description
minNodeCount	No	Integer	Details: Minimum number of nodes that can be scaled in if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	No	Integer	Details: Maximum number of nodes that can be added if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to the value of minNodeCount and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
scaleDownCooldownTime	No	Integer	Details: How long nodes will be kept, in minutes. During this period, nodes added will not be deleted. Constraints: None Options: 0 to 2147483646 Default value: 0

Parameter	Mandatory	Type	Description
priority	No	Integer	Details: Weight of a node pool. A node pool with a higher weight has a higher priority during scale-out. Constraints: None Options: N/A Default value: 0
scaleDownUnneededTime	No	Integer	Details: How long a node should be unneeded before it is eligible for scale in. This parameter specifies the duration during which a node is in the unneeded state before the scale-in operation can be performed. CCE Cluster Autoscaler scales in the node only when the node is not needed for a specified period. This prevents frequent scale-in triggered by temporary resource fluctuations and improves system stability. If this parameter is not specified, the CCE Cluster Autoscaler add-on uses the default time threshold. Constraints: None Options: 0 to 2147483647 NOTE Note: If the parameter is set to -1 , it indicates that the value is empty. Default value: This parameter is left empty by default.

Parameter	Mandatory	Type	Description
scaleDownUtilizationThreshold	No	Float	Details: Scale-in utilization threshold of a node, which is defined by the total CPU or memory resources used by all running pods on the node divided by the total available resources of the node. If the calculated value falls below the specified threshold, the node becomes eligible for scale-in. For example, setting this parameter to 0.3 means that if the resource utilization on a node drops below 30%, the node will be scaled in. If this parameter is not specified, the CCE Cluster Autoscaler add-on will use the default utilization threshold. Constraints: None Options: 0 to 1 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Table 4-749 NodeManagement

Parameter	Mandatory	Type	Description
serverGroupReference	No	String	Details: A cloud server group ID. If this parameter is specified, the cloud server group will be associated with the target node pool, and all nodes in the node pool will be created in this cloud server group. After a cloud server group is associated with a node pool, the number of nodes in the node pool cannot exceed the number of cloud servers that can be added to the cloud server group, or the node pool cannot be scaled out. NOTE - After a cloud server group is associated with a node pool, the cloud servers added to the cloud server group will be distributed in strict accordance with the affinity policy and the maximum number of nodes in the node pool will be limited. The anti-affinity capabilities are available during cloud server creation on ECS. If your goal is to enhance service reliability by distributing cloud servers across different hosts and you do not want the number of nodes to be restricted by cloud server groups, you are advised not to associate cloud server groups with any node pools. - A cloud server group can be disassociated from the node pool. After the change, the existing nodes will remain in the original cloud server group, and no new nodes will be added to the cloud server group. If the node pool is empty, you can associate a new cloud server group with the node pool or change the associated cloud server group. Constraints: If a cloud server group is specified, the number of nodes

Parameter	Mandatory	Type	Description
			in the node pool cannot exceed the group quota. Options: • If this parameter is not specified or left blank, the cloud server group will be disassociated from the node pool. • A cloud server group ID: The cloud server group associated with the node pool will be changed to a specified one. Default value: N/A

Table 4-750 ExtensionScaleGroup

Parameter	Mandatory	Type	Description
metadata	No	ExtensionScaleGroupMetadata object	Details: Basic information about an extended scaling group Constraints: None
spec	No	ExtensionScaleGroupSpec object	Details: Configurations of an extended scaling group, which carry different configurations from those of the default scaling group Constraints: None

Table 4-751 ExtensionScaleGroupMetadata

Parameter	Mandatory	Type	Description
uid	No	String	Details: UUID of an extended scaling group, which is automatically generated by the system Constraints: None Options: N/A Default value: N/A
name	No	String	Details: Name of an extended scaling group Constraints: The value cannot be default . Options: Enter 1 to 56 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only digits, lowercase letters, and hyphens (-) are allowed. Default value: N/A

Table 4-752 ExtensionScaleGroupSpec

Parameter	Mandatory	Type	Description
flavor	No	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
az	No	String	Details: Node AZ Constraints: None Options: N/A Default value: If this parameter is not specified or left blank, the default scaling group configurations take effect.
capacityReservationSpecification	No	CapacityReservationSpecification object	Details: Capacity reservation configurations of an extended scaling group Constraints: None
autoscaling	No	ScaleGroupAutoscaling object	Details: Auto scaling configurations of an extended scaling group Constraints: None

Table 4-753 CapacityReservationSpecification

Parameter	Mandatory	Type	Description
id	No	String	Details: Private pool ID Constraints: This value can be ignored when preference is set to none . Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
preference	No	String	Details: Private pool capacity Constraints: If the value is none , the capacity reservation is not specified. If the value is targeted , the capacity reservation is specified. In this case, the ID cannot be left blank. Options: N/A Default value: none

Table 4-754 ScaleGroupAutoscaling

Parameter	Mandatory	Type	Description
enable	No	Boolean	Details: Whether to enable auto scaling for a scaling group Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
extensionPriority	No	Integer	Details: Scaling group priority. A larger value indicates a higher priority. Constraints: None Options: N/A Default value: 0

Parameter	Mandatory	Type	Description
minNodeCount	No	Integer	Details: Minimum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	No	Integer	Details: Maximum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to the value of minNodeCount , less than or equal to the maximum number of nodes allowed by the cluster specifications, and less than or equal to the maximum number of nodes in the node pool Default value: 0

Response Parameters

Status code: 200

Table 4-755 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: NodePool
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
metadata	NodePoolMetadata object	Details: Metadata of a node pool Constraints: None Options: N/A Default value: N/A
spec	NodePoolSpec object	Details: Node pool specifications Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
status	UpdateNodePoolStatus object	Details: Node pool status Constraints: None Options: N/A Default value: N/A

Table 4-756 NodePoolMetadata

Parameter	Type	Description
name	String	Details: Node pool name Constraints: You cannot create node pools named DefaultPool . Options: Enter 1 to 50 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: UID of a node pool Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A

Parameter	Type	Description
annotations	Map<String,String>	Details: Annotations of a node pool in key-value pairs Constraints: This parameter is used only for query and cannot be input through a request. Entered data of this parameter is invalid.
updateTimestamp	String	Details: Time when a node pool was updated Constraints: The value is automatically recorded when a node pool is updated and cannot be specified. Options: N/A Default value: N/A
creationTimestamp	String	Details: Time when a node pool was created Constraints: The value is automatically recorded when a node pool is created and cannot be specified. Options: N/A Default value: N/A
resourceVersion	Integer	Definition Timestamp of the last update time of a node pool Constraints The value is automatically recorded when a node pool is created and cannot be specified. Range N/A Default Value N/A

Table 4-757 NodePoolSpec

Parameter	Type	Description
type	String	Details: Node pool type Constraints: None Options: • vm: ECS • ElasticBMS: C6 general computing-plus BMS. An example flavor is c6.22xlarge.2.physical. • pm: BMS Default value: vm
nodeTemplate	NodeTemplate object	Details: Detailed parameters of a node pool template Constraints: None
initialNodeCount	Integer	Details: The initial number of nodes in a node pool. The obtained value is the desired number of nodes in the node pool. Constraints: None Options: Values that are greater than or equal to 0 and less than or equal to the cluster scale Default value: 0
autoscaling	NodePoolNodeAutoscaling object	Details: Auto scaling parameters Constraints: None
nodeManagement	NodeManagement object	Details: Node management configuration Constraints: None

Parameter	Type	Description
podSecurityGroups	Array of SecurityID objects	Details: Security group configuration Constraints: None
extensionScaleGroups	Array of ExtensionScaleGroup objects	Details: Configuration list of the extended scaling groups in a node pool. For details, see the definition of ExtensionScaleGroup . Constraints: None
customSecurityGroups	Array of strings	Details: Custom security group configuration for a node pool. New nodes added to a node pool can be bound to a specified security group. • Specifying no security group ID will add the new nodes to the default node security group. • Specifying a valid security group ID will put new nodes in that security group. • When specifying a security group, do not modify the port rules on which CCE depends. For details, see How Can I Configure a Security Group Rule in a Cluster? Constraints: None

Parameter	Type	Description
taintPolicyOnExistingNodes	String	Definition: Taint synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The taints of existing nodes in a node pool will not be updated synchronously. • refresh: The taints of existing nodes in a node pool will be updated synchronously. Default value: refresh
labelPolicyOnExistingNodes	String	Definition: Label synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The labels of existing nodes in a node pool will not be updated synchronously. • refresh: The labels of existing nodes in a node pool will be updated synchronously. Default value: refresh

Parameter	Type	Description
userTagsPolicyOnExistingNodes	String	Definition: Resource tag synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The resource tags of existing nodes in a node pool will not be updated synchronously. • refresh: The resource tags of existing nodes in a node pool will be updated synchronously. Default value: ignore

Table 4-758 NodeTemplate

Parameter	Type	Description
flavor	String	Definition Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
az	String	Definition AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to the nodes created using APIs. For details about AZs supported by CCE, see Regions and Endpoints. Constraints When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random. Range N/A Default Value N/A
os	String	Definition OS type of a node. For details about the supported OSs, see Node OSs. Constraints • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Range N/A Default Value N/A

Parameter	Type	Description
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Definition Disk information of a node Constraints N/A
dataVolumes	Array of Volume objects	Definition Data disk parameters of a node. For DeC nodes, the parameter definitions are the same as those in rootVolume . Constraints • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left empty only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodeEIPSpec object	Definition EIP parameters of a node Constraints N/A Range N/A Default Value N/A
nodeNicSpec	NodeNicSpec object	Definition Network interfaces of a node Constraints N/A

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Definition Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints A maximum of 20 key-value pairs is allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Definition Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Definition Container runtime. • In clusters earlier than v1.25, the default value is docker. • In clusters of v1.25 or later, the default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints N/A

Parameter	Type	Description
initializedConditions	Array of strings	Definition Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When creating a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. Create a node and add "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions and add two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint will be removed. 4. initializedConditions allows you to configure a timeout interval during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions:

Parameter	Type	Description
		- type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	NodeExtendParam object	Definition Extended parameters for creating a node Constraints N/A
hostnameConfig	HostnameConfig object	Definition Kubernetes node name configuration parameter Constraints The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Definition Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints To create a node or node pool, you can specify an existing enterprise project for it. If this field is left empty, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Range N/A Default Value If no value is specified during the update, this field will not be updated. If this parameter is left empty, the cluster enterprise project is returned.
partition	String	Definition Zone that a node belongs to. It can be the central cloud or an edge site. Constraints This field can be specified only for CCE Turbo clusters that support distributed clouds. Range • center: central cloud • AZ ID of the edge site Default Value N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Definition Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(five-random-digits)-product . Constraints This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Range N/A Default Value N/A

Table 4-759 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-760 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-761 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss. Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-762 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-763 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-764 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-765 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-766 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-767 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-768 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-769 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-770 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-771 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-772 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-773 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-774 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-775 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-776 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-777 NodeExtendParam

Parameter	Type	Description
ecs:performancety pe	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <code>echo -n "**Content to be encoded*" base64</code> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-778 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-779 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123. Range N/A Default Value N/A

Table 4-780 NodePoolNodeAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
minNodeCount	Integer	Details: Minimum number of nodes that can be scaled in if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be added if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to the value of minNodeCount and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0

Parameter	Type	Description
scaleDownCoolDownTime	Integer	Details: How long nodes will be kept, in minutes. During this period, nodes added will not be deleted. Constraints: None Options: 0 to 2147483646 Default value: 0
priority	Integer	Details: Weight of a node pool. A node pool with a higher weight has a higher priority during scale-out. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
scaleDownUnneededTime	Integer	Details: How long a node should be unneeded before it is eligible for scale in. This parameter specifies the duration during which a node is in the unneeded state before the scale-in operation can be performed. CCE Cluster Autoscaler scales in the node only when the node is not needed for a specified period. This prevents frequent scale-in triggered by temporary resource fluctuations and improves system stability. If this parameter is not specified, the CCE Cluster Autoscaler add-on uses the default time threshold. Constraints: None Options: 0 to 2147483647 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Parameter	Type	Description
scaleDownUtilizationThreshold	Float	Details: Scale-in utilization threshold of a node, which is defined by the total CPU or memory resources used by all running pods on the node divided by the total available resources of the node. If the calculated value falls below the specified threshold, the node becomes eligible for scale-in. For example, setting this parameter to 0.3 means that if the resource utilization on a node drops below 30%, the node will be scaled in. If this parameter is not specified, the CCE Cluster Autoscaler add-on will use the default utilization threshold. Constraints: None Options: 0 to 1 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Table 4-781 NodeManagement

Parameter	Type	Description
serverGroupReference	String	Details: A cloud server group ID. If this parameter is specified, the cloud server group will be associated with the target node pool, and all nodes in the node pool will be created in this cloud server group. After a cloud server group is associated with a node pool, the number of nodes in the node pool cannot exceed the number of cloud servers that can be added to the cloud server group, or the node pool cannot be scaled out. NOTE - After a cloud server group is associated with a node pool, the cloud servers added to the cloud server group will be distributed in strict accordance with the affinity policy and the maximum number of nodes in the node pool will be limited. The anti-affinity capabilities are available during cloud server creation on ECS. If your goal is to enhance service reliability by distributing cloud servers across different hosts and you do not want the number of nodes to be restricted by cloud server groups, you are advised not to associate cloud server groups with any node pools. - A cloud server group can be disassociated from the node pool. After the change, the existing nodes will remain in the original cloud server group, and no new nodes will be added to the cloud server group. If the node pool is empty, you can associate a new cloud server group with the node pool or change the associated cloud server group. Constraints: If a cloud server group is specified, the number of nodes in the node pool cannot exceed the group quota. Options: - If this parameter is not specified or left blank, the cloud server group will be disassociated from the node pool.

Parameter	Type	Description
		A cloud server group ID: The cloud server group associated with the node pool will be changed to a specified one. Default value: N/A

Table 4-782 SecurityID

Parameter	Type	Description
id	String	Definition ID of the container security group on a node Constraints N/A Range N/A Default Value N/A

Table 4-783 ExtensionScaleGroup

Parameter	Type	Description
metadata	ExtensionScaleGroupMetadata object	Details: Basic information about an extended scaling group Constraints: None
spec	ExtensionScaleGroupSpec object	Details: Configurations of an extended scaling group, which carry different configurations from those of the default scaling group Constraints: None

Table 4-784 ExtensionScaleGroupMetadata

Parameter	Type	Description
uid	String	Details: UUID of an extended scaling group, which is automatically generated by the system Constraints: None Options: N/A Default value: N/A
name	String	Details: Name of an extended scaling group Constraints: The value cannot be default . Options: Enter 1 to 56 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only digits, lowercase letters, and hyphens (-) are allowed. Default value: N/A

Table 4-785 ExtensionScaleGroupSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
az	String	Details: Node AZ Constraints: None Options: N/A Default value: If this parameter is not specified or left blank, the default scaling group configurations take effect.
capacityReservationSpecification	CapacityReservationSpecification object	Details: Capacity reservation configurations of an extended scaling group Constraints: None
autoscaling	ScaleGroupAutoscaling object	Details: Auto scaling configurations of an extended scaling group Constraints: None

Table 4-786 CapacityReservationSpecification

Parameter	Type	Description
id	String	Details: Private pool ID Constraints: This value can be ignored when preference is set to none . Options: N/A Default value: N/A

Parameter	Type	Description
preference	String	Details: Private pool capacity Constraints: If the value is none , the capacity reservation is not specified. If the value is targeted , the capacity reservation is specified. In this case, the ID cannot be left blank. Options: N/A Default value: none

Table 4-787 ScaleGroupAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling for a scaling group Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
extensionPriority	Integer	Details: Scaling group priority. A larger value indicates a higher priority. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
minNodeCount	Integer	Details: Minimum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to the value of minNodeCount , less than or equal to the maximum number of nodes allowed by the cluster specifications, and less than or equal to the maximum number of nodes in the node pool Default value: 0

Table 4-788 UpdateNodePoolStatus

Parameter	Type	Description
currentNode	Integer	Details: Total number of nodes in the current node pool (excluding the nodes that are being deleted) Constraints: None Options: N/A Default value: N/A
creatingNode	Integer	Details: Number of nodes that are being created in the current node pool Constraints: None Options: N/A Default value: N/A
deletingNode	Integer	Details: Number of nodes that are being deleted in the current node pool Constraints: None Options: N/A Default value: N/A
configurationSyncedNodeCount	Integer	Details: The number of nodes specified in the node pool configuration has been synchronized in the current node pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Node pool status Constraints: None Options: • Null: The node pool is available. (The number of nodes in the current node pool has reached the expected value, and no node scaling is being performed.) • Synchronizing: The node pool is being scaled. (The number of nodes in the current node pool has not reached the expected value, and no node scaling is being performed.) • Synchronized: The node pool scaling is pending. (The number of nodes in the current node pool has not reached the expected value, and node scaling is being performed.) • SoldOut: The node pool cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The node pool is being deleted. • Error: An error occurred. NOTE The preceding node pool statuses have been deprecated. They are reserved only for compatibility. Do not use them anymore. Use the following instead: • Node pool scaling status: You can obtain the status of the current node pool using parameters such as currentNode, creatingNode, or deletingNode. • Node pool scale-out: You can use conditions to obtain the detailed status of a node pool. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
conditions	Array of NodePoolCondition objects	Details: Details of node pool statuses. For details, see the definition of Condition . Constraints: None
scaleGroupStatuses	Array of ScaleGroupStatus objects	Details: Detailed status of a scaling group. For details, see the definition of ScaleGroupStatus . Constraints: None

Table 4-789 ScaleGroupStatus

Parameter	Type	Description
name	String	Details: Scaling group name Constraints: None Options: N/A Default value: N/A
uid	String	Details: Scaling group UUID Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
creationTimestamp	String	Details: The time when a scaling group was created Constraints: None Options: N/A Default value: N/A
updateTimestamp	String	Details: The time when a scaling group was updated Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Scaling group status Constraints: None Options: • Null: The scaling group is available. (The number of nodes in the current scaling group has reached the expected value, and no node scaling is being performed.) • Synchronizing: The scaling group is being scaled. (The number of nodes in the current scaling group has not reached the expected value, and no node scaling is being performed.) • Synchronized: The scaling is pending. (The number of nodes in the current scaling group has not reached the expected value, or node scaling is being performed.) • SoldOut: The scaling group cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The scaling group is being deleted. • Error: An error occurred. NOTE The preceding scaling group status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Scaling status of a scaling group: You can obtain the status of the current scaling group using parameters such as desiredNodeCount, existingNodeCount, or upcomingNodeCount. • Scaling group scale-out: You can use conditions to obtain the detailed status of a scaling group. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
desiredNodeCount	Integer	Details: Expected number of nodes in a scaling group Constraints: None Options: N/A Default value: N/A
unpaidScaleNodeCount	Integer	Details: Number of nodes in an unpaid order Constraints: None Options: N/A Default value: N/A
existingNodeCount	existingNodeCount object	Details: Statistics on existing nodes in a scaling group Constraints: None
upcomingNodeCount	upcomingNodeCount object	Details: Statistics on the nodes to be added in a scaling group Constraints: None
scaleDownDisabledNodeCount	Integer	Details: Number of nodes that cannot be deleted in a scaling group Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
conditions	Array of NodePoolCondition objects	Details: Scaling group status details. For details, see the definition of Condition . Constraints: None

Table 4-790 existingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-791 upcomingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-792 NodePoolCondition

Parameter	Type	Description
type	String	Definition Status type Constraints N/A Range • TaintSynchronizing: The node pool is synchronizing Kubernetes taints for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • LabelSynchronizing: The node pool is synchronizing Kubernetes labels for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • UserTagsSynchronizing: The node pool is synchronizing Kubernetes resource tags for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • ConfigurationSynchronizing: The node pool is synchronizing settings for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • Scalable: specifies whether a node pool/scaling group can be scaled out. If the status is False, node pool scale-out will not be triggered again. • QuotaInsufficient: Quotas on which node pool/scaling group scale-out depends are insufficient, affecting the node pool scale-out status. • ResourceInsufficient: Resources on which node pool/scaling group

Parameter	Type	Description
		scale-out depends are insufficient, affecting the node pool scale-out status. • UnexpectedError: The node pool/scaling group fails to be scaled out due to unexpected reasons, affecting the node pool scale-out status. • LockedByOrder: The node pool/scaling group is locked due to a pending order. The value of Reason is the ID of the order. • Error: There is an error in the node pool/scaling group. The error is typically triggered by a deletion failure. Default Value N/A
status	String	Details: Status of the condition Constraints: None Options: • True • False Default value: N/A
lastProbeTime	String	Details: Time when the status was last checked Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
lastTransitTime	String	Details: Time when the status was last changed Constraints: None Options: N/A Default value: N/A
reason	String	Details: Reason why the status was last changed Constraints: None Options: N/A Default value: N/A
message	String	Details: Detailed description about the condition Constraints: None Options: N/A Default value: N/A

Example Requests

- Change the number of nodes in the node pool to 1.

```
PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}
```

```
{
  "metadata" : {
    "name" : "lc-it-nodepool-3"
  },
  "spec" : {
    "nodeTemplate" : {
      "k8sTags" : { },
      "taints" : [ {
        "key" : "status",
        "value" : "unavailable",
        "effect" : "NoSchedule"
      } ],
    }
  }
}
```

```
    "userTags" : [ ]
  },
  "autoscaling" : {
    "enable" : false,
    "minNodeCount" : 0,
    "maxNodeCount" : 0,
    "scaleDownCooldownTime" : 0,
    "priority" : 0,
    "scaleDownUnneededTime" : 5,
    "scaleDownUtilizationThreshold" : 0.6
  },
  "initialNodeCount" : 1
}
}
```

- **Modify the key pair.**

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}

```
{
  "metadata" : {
    "name" : "lc-it-nodepool-2"
  },
  "spec" : {
    "nodeTemplate" : {
      "k8sTags" : { },
      "taints" : [ {
        "key" : "status",
        "value" : "unavailable",
        "effect" : "NoSchedule"
      } ],
      "userTags" : [ ],
      "login" : {
        "sshKey" : "KeyPair-IES"
      }
    },
    "autoscaling" : {
      "enable" : false,
      "minNodeCount" : 0,
      "maxNodeCount" : 0,
      "scaleDownCooldownTime" : 0,
      "priority" : 0,
      "scaleDownUnneededTime" : 5,
      "scaleDownUtilizationThreshold" : 0.6
    },
    "initialNodeCount" : 1
  }
}
```

- **Modify the cloud server group of a node pool.**

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}

```
{
  "metadata" : {
    "name" : "lc-it-nodepool-3"
  },
  "spec" : {
    "nodeTemplate" : {
      "k8sTags" : { },
      "taints" : [ {
        "key" : "status",
        "value" : "unavailable",
        "effect" : "NoSchedule"
      } ],
      "userTags" : [ ]
    },
    "autoscaling" : { },
    "initialNodeCount" : 1,
    "nodeManagementUpdate" : {
      "serverGroupReference" : "8a611bcf-xxxx-xxxx-xxxx-be4ac7bc9075"
    }
  }
}
```

```
}  
}
```

Example Responses

Status code: 200

Information about the specified node pool is successfully updated.

```
{  
  "kind" : "NodePool",  
  "apiVersion" : "v3",  
  "metadata" : {  
    "name" : "lc-it-nodepool-3",  
    "uid" : "1deef848-690d-11ea-a11b-0255ac1001b7"  
  },  
  "spec" : {  
    "initialNodeCount" : 1,  
    "type" : "vm",  
    "nodeTemplate" : {  
      "flavor" : "Sit3.xlarge.2",  
      "az" : "*****",  
      "os" : "EulerOS 2.5",  
      "login" : {  
        "sshKey" : "KeyPair-001"  
      },  
      "rootVolume" : {  
        "volumetype" : "SAS",  
        "size" : 40  
      },  
      "dataVolumes" : [ {  
        "volumetype" : "SAS",  
        "size" : 100,  
        "extendParam" : {  
          "useType" : "docker"  
        }  
      } ],  
      "publicIP" : {  
        "iptype" : "5_g-vm",  
        "bandwidth" : {  
          "Chargemode" : "bandwidth",  
          "Size" : 5,  
          "Sharetype" : "PER"  
        }  
      },  
      "nodeNicSpec" : {  
        "primaryNic" : {  
          "subnetId" : "7e767d10-7548-4df5-ad72-aeac1d08bd8a"  
        }  
      },  
      "billingMode" : 0,  
      "extendParam" : {  
        "maxPods" : 110  
      },  
      "k8sTags" : {  
        "cce.cloud.com/cce-nodepool" : "lc-it-nodepool-3"  
      }  
    },  
    "autoscaling" : { },  
    "nodeManagement" : {  
      "serverGroupReference" : "8a611bcf-xxxx-xxxx-xxxx-be4ac7bc9075"  
    }  
  },  
  "status" : {  
    "phase" : ""  
  }  
}
```


SDK Sample Code

The SDK sample code is as follows.

Java

- Change the number of nodes in the node pool to 1.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class UpdateNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateNodePoolRequest request = new UpdateNodePoolRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        NodePoolUpdate body = new NodePoolUpdate();
        NodePoolNodeAutoscaling autoscalingSpec = new NodePoolNodeAutoscaling();
        autoscalingSpec.withEnable(false)
            .withMinNodeCount(0)
            .withMaxNodeCount(0)
            .withScaleDownCooldownTime(0)
            .withPriority(0)
            .withScaleDownUnneededTime(5)
            .withScaleDownUtilizationThreshold(0.6f);
        List<Taint> listNodeTemplateTaints = new ArrayList<>();
        listNodeTemplateTaints.add(
            new Taint()
                .withKey("status")
                .withValue("unavailable")
                .withEffect(Taint.EffectEnum.fromValue("NoSchedule"))
        );
        NodeSpecUpdate nodeTemplateSpec = new NodeSpecUpdate();
        nodeTemplateSpec.withTaints(listNodeTemplateTaints);
        NodePoolSpecUpdate specbody = new NodePoolSpecUpdate();
        specbody.withNodeTemplate(nodeTemplateSpec)
```

```
        .withInitialNodeCount(1)
        .withAutoscaling(autoscalingSpec);
NodePoolMetadataUpdate metadatabody = new NodePoolMetadataUpdate();
metadatabody.setName("lc-it-nodepool-3");
body.withSpec(specbody);
body.withMetadata(metadatabody);
request.withBody(body);
try {
    UpdateNodePoolResponse response = client.updateNodePool(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- **Modify the key pair.**

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class UpdateNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateNodePoolRequest request = new UpdateNodePoolRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        NodePoolUpdate body = new NodePoolUpdate();
        NodePoolNodeAutoscaling autoscalingSpec = new NodePoolNodeAutoscaling();
        autoscalingSpec.withEnable(false)
            .withMinNodeCount(0)
```

```
.withMaxNodeCount(0)
.withScaleDownCooldownTime(0)
.withPriority(0)
.withScaleDownUnneededTime(5)
.withScaleDownUtilizationThreshold(0.6f);
List<Taint> listNodeTemplateTaints = new ArrayList<>();
listNodeTemplateTaints.add(
    new Taint()
        .withKey("status")
        .withValue("unavailable")
        .withEffect(Taint.EffectEnum.fromValue("NoSchedule"))
);
Login loginNodeTemplate = new Login();
loginNodeTemplate.withSshKey("KeyPair-IES");
NodeSpecUpdate nodeTemplateSpec = new NodeSpecUpdate();
nodeTemplateSpec.withLogin(loginNodeTemplate)
    .withTaints(listNodeTemplateTaints);
NodePoolSpecUpdate specbody = new NodePoolSpecUpdate();
specbody.withNodeTemplate(nodeTemplateSpec)
    .withInitialNodeCount(1)
    .withAutoscaling(autoscalingSpec);
NodePoolMetadataUpdate metadatabody = new NodePoolMetadataUpdate();
metadatabody.withName("lc-it-nodepool-2");
body.withSpec(specbody);
body.withMetadata(metadatabody);
request.withBody(body);
try {
    UpdateNodePoolResponse response = client.updateNodePool(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

- Modify the cloud server group of a node pool.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;
import java.util.Map;
import java.util.HashMap;

public class UpdateNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateNodePoolRequest request = new UpdateNodePoolRequest();
request.withClusterId("{cluster_id}");
request.withNodepoolId("{nodepool_id}");
NodePoolUpdate body = new NodePoolUpdate();
NodeManagement nodeManagementUpdateSpec = new NodeManagement();
nodeManagementUpdateSpec.withServerGroupReference("8a611bcf-xxxx-xxxx-xxxx-
be4ac7bc9075");
List<Taint> listNodeTemplateTaints = new ArrayList<>();
listNodeTemplateTaints.add(
    new Taint()
        .withKey("status")
        .withValue("unavailable")
        .withEffect(Taint.EffectEnum.fromValue("NoSchedule"))
);
NodeSpecUpdate nodeTemplateSpec = new NodeSpecUpdate();
nodeTemplateSpec.withTaints(listNodeTemplateTaints);
NodePoolSpecUpdate specbody = new NodePoolSpecUpdate();
specbody.withNodeTemplate(nodeTemplateSpec)
    .withInitialNodeCount(1)
    .withNodeManagementUpdate(nodeManagementUpdateSpec);
NodePoolMetadataUpdate metadatabody = new NodePoolMetadataUpdate();
metadatabody.withName("lc-it-nodepool-3");
body.withSpec(specbody);
body.withMetadata(metadatabody);
request.withBody(body);
try {
    UpdateNodePoolResponse response = client.updateNodePool(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

- Change the number of nodes in the node pool to 1.

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```

```
if __name__ == "__main__":
```

```
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
```

```
# In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = UpdateNodePoolRequest()
    request.cluster_id = "{cluster_id}"
    request.nodepool_id = "{nodepool_id}"
    autoscalingSpec = NodePoolNodeAutoscaling(
        enable=False,
        min_node_count=0,
        max_node_count=0,
        scale_down_cooldown_time=0,
        priority=0,
        scale_down_unneeded_time=5,
        scale_down_utilization_threshold=0.6
    )
    listTaintsNodeTemplate = [
        Taint(
            key="status",
            value="unavailable",
            effect="NoSchedule"
        )
    ]
    nodeTemplateSpec = NodeSpecUpdate(
        taints=listTaintsNodeTemplate
    )
    specbody = NodePoolSpecUpdate(
        node_template=nodeTemplateSpec,
        initial_node_count=1,
        autoscaling=autoscalingSpec
    )
    metadatabody = NodePoolMetadataUpdate(
        name="lc-it-nodepool-3"
    )
    request.body = NodePoolUpdate(
        spec=specbody,
        metadata=metadatabody
    )
    response = client.update_node_pool(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- **Modify the key pair.**

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
```

```
# In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = UpdateNodePoolRequest()
    request.cluster_id = "{cluster_id}"
    request.nodepool_id = "{nodepool_id}"
    autoscalingSpec = NodePoolNodeAutoscaling(
        enable=False,
        min_node_count=0,
        max_node_count=0,
        scale_down_cooldown_time=0,
        priority=0,
        scale_down_unneeded_time=5,
        scale_down_utilization_threshold=0.6
    )
    listTaintsNodeTemplate = [
        Taint(
            key="status",
            value="unavailable",
            effect="NoSchedule"
        )
    ]
    loginNodeTemplate = Login(
        ssh_key="KeyPair-IES"
    )
    nodeTemplateSpec = NodeSpecUpdate(
        login=loginNodeTemplate,
        taints=listTaintsNodeTemplate
    )
    specbody = NodePoolSpecUpdate(
        node_template=nodeTemplateSpec,
        initial_node_count=1,
        autoscaling=autoscalingSpec
    )
    metadatabody = NodePoolMetadataUpdate(
        name="lc-it-nodepool-2"
    )
    request.body = NodePoolUpdate(
        spec=specbody,
        metadata=metadatabody
    )
    response = client.update_node_pool(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Modify the cloud server group of a node pool.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateNodePoolRequest()
        request.cluster_id = "{cluster_id}"
        request.nodepool_id = "{nodepool_id}"
        nodeManagementUpdateSpec = NodeManagement(
            server_group_reference="8a611bcf-xxxx-xxxx-xxxx-be4ac7bc9075"
        )
        listTaintsNodeTemplate = [
            Taint(
                key="status",
                value="unavailable",
                effect="NoSchedule"
            )
        ]
        nodeTemplateSpec = NodeSpecUpdate(
            taints=listTaintsNodeTemplate
        )
        specbody = NodePoolSpecUpdate(
            node_template=nodeTemplateSpec,
            initial_node_count=1,
            node_management_update=nodeManagementUpdateSpec
        )
        metadatabody = NodePoolMetadataUpdate(
            name="lc-it-nodepool-3"
        )
        request.body = NodePoolUpdate(
            spec=specbody,
            metadata=metadatabody
        )
        response = client.update_node_pool(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- Change the number of nodes in the node pool to 1.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before
running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateNodePoolRequest{
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    enableAutoscaling:= false
    minNodeCountAutoscaling:= int32(0)
    maxNodeCountAutoscaling:= int32(0)
    scaleDownCooldownTimeAutoscaling:= int32(0)
    priorityAutoscaling:= int32(0)
    scaleDownUnneededTimeAutoscaling:= int32(5)
    scaleDownUtilizationThresholdAutoscaling:= float32(0.6)
    autoscalingSpec := &model.NodePoolNodeAutoscaling{
        Enable: &enableAutoscaling,
        MinNodeCount: &minNodeCountAutoscaling,
        MaxNodeCount: &maxNodeCountAutoscaling,
        ScaleDownCooldownTime: &scaleDownCooldownTimeAutoscaling,
        Priority: &priorityAutoscaling,
        ScaleDownUnneededTime: &scaleDownUnneededTimeAutoscaling,
        ScaleDownUtilizationThreshold: &scaleDownUtilizationThresholdAutoscaling,
    }
    valueTaints:= "unavailable"
    var listTaintsNodeTemplate = []model.Taint{
        {
            Key: "status",
            Value: &valueTaints,
            Effect: model.GetTaintEffectEnum().NO_SCHEDULE,
        },
    }
    nodeTemplateSpec := &model.NodeSpecUpdate{
        Taints: listTaintsNodeTemplate,
    }
    specbody := &model.NodePoolSpecUpdate{
        NodeTemplate: nodeTemplateSpec,
        InitialNodeCount: int32(1),
        Autoscaling: autoscalingSpec,
    }
    metadatabody := &model.NodePoolMetadataUpdate{
        Name: "lc-it-nodepool-3",
    }
    request.Body = &model.NodePoolUpdate{
        Spec: specbody,
        Metadata: metadatabody,
    }
    response, err := client.UpdateNodePool(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```


- Modify the key pair.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateNodePoolRequest{
        request.ClusterId = "{cluster_id}"
        request.NodepoolId = "{nodepool_id}"
        enableAutoscaling:= false
        minNodeCountAutoscaling:= int32(0)
        maxNodeCountAutoscaling:= int32(0)
        scaleDownCooldownTimeAutoscaling:= int32(0)
        priorityAutoscaling:= int32(0)
        scaleDownUnneededTimeAutoscaling:= int32(5)
        scaleDownUtilizationThresholdAutoscaling:= float32(0.6)
        autoscalingSpec := &model.NodePoolNodeAutoscaling{
            Enable: &enableAutoscaling,
            MinNodeCount: &minNodeCountAutoscaling,
            MaxNodeCount: &maxNodeCountAutoscaling,
            ScaleDownCooldownTime: &scaleDownCooldownTimeAutoscaling,
            Priority: &priorityAutoscaling,
            ScaleDownUnneededTime: &scaleDownUnneededTimeAutoscaling,
            ScaleDownUtilizationThreshold: &scaleDownUtilizationThresholdAutoscaling,
        }
        valueTaints:= "unavailable"
        var listTaintsNodeTemplate = []model.Taint{
            {
                Key: "status",
                Value: &valueTaints,
                Effect: model.GetTaintEffectEnum().NO_SCHEDULE,
            },
        }
        sshKeyLogin:= "KeyPair-IES"
        loginNodeTemplate := &model.Login{
            SshKey: &sshKeyLogin,
        }
        nodeTemplateSpec := &model.NodeSpecUpdate{
            Login: loginNodeTemplate,
```

```
Taints: listTaintsNodeTemplate,
}
specbody := &model.NodePoolSpecUpdate{
    NodeTemplate: nodeTemplateSpec,
    InitialNodeCount: int32(1),
    Autoscaling: autoscalingSpec,
}
metadatabody := &model.NodePoolMetadataUpdate{
    Name: "lc-it-nodepool-2",
}
request.Body = &model.NodePoolUpdate{
    Spec: specbody,
    Metadata: metadatabody,
}
response, err := client.UpdateNodePool(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

- Modify the cloud server group of a node pool.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateNodePoolRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    serverGroupReferenceNodeManagementUpdate := "8a611bcf-xxxx-xxxx-xxxx-be4ac7bc9075"
    nodeManagementUpdateSpec := &model.NodeManagement{
        ServerGroupReference: &serverGroupReferenceNodeManagementUpdate,
    }
    valueTaints := "unavailable"
    var listTaintsNodeTemplate = []model.Taint{
        {
            Key: "status",
            Value: &valueTaints,
            Effect: model.GetTaintEffectEnum().NO_SCHEDULE,
        },
    },
}
```

```
}
nodeTemplateSpec := &model.NodeSpecUpdate{
    Taints: listTaintsNodeTemplate,
}
specbody := &model.NodePoolSpecUpdate{
    NodeTemplate: nodeTemplateSpec,
    InitialNodeCount: int32(1),
    NodeManagementUpdate: nodeManagementUpdateSpec,
}
metadatabody := &model.NodePoolMetadataUpdate{
    Name: "lc-it-nodepool-3",
}
request.Body = &model.NodePoolUpdate{
    Spec: specbody,
    Metadata: metadatabody,
}
response, err := client.UpdateNodePool(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Information about the specified node pool is successfully updated.

Error Codes

See [Error Codes](#).

4.4.5 Deleting a Node Pool

Function

This API is used to delete a specified node pool.

NOTE

The URL for cluster management is in the format of https://Endpoint/uri. In the URL, **uri** indicates the resource path, that is, the path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}

Table 4-793 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-794 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-795 Response body parameters

Parameter	Type	Description
kind	String	Details: API type Constraints: The value cannot be changed. Options: N/A Default value: NodePool
apiVersion	String	Details: API version Constraints: The value cannot be changed. Options: N/A Default value: v3
metadata	NodePoolMetadata object	Details: Metadata of a node pool Constraints: None Options: N/A Default value: N/A
spec	NodePoolSpec object	Definition Node pool specifications
status	DeleteNodePoolStatus object	Details: Node pool status Constraints: None Options: N/A Default value: N/A

Table 4-796 NodePoolMetadata

Parameter	Type	Description
name	String	Details: Node pool name Constraints: You cannot create node pools named DefaultPool . Options: Enter 1 to 50 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, and hyphens (-) are allowed. Default value: N/A
uid	String	Details: UID of a node pool Constraints: The value is automatically generated after the object is created. A user-defined value will not take effect. Options: N/A Default value: N/A
annotations	Map<String,String>	Details: Annotations of a node pool in key-value pairs Constraints: This parameter is used only for query and cannot be input through a request. Entered data of this parameter is invalid.

Parameter	Type	Description
updateTimestamp	String	Details: Time when a node pool was updated Constraints: The value is automatically recorded when a node pool is updated and cannot be specified. Options: N/A Default value: N/A
creationTimestamp	String	Details: Time when a node pool was created Constraints: The value is automatically recorded when a node pool is created and cannot be specified. Options: N/A Default value: N/A
resourceVersion	Integer	Definition Timestamp of the last update time of a node pool Constraints The value is automatically recorded when a node pool is created and cannot be specified. Range N/A Default Value N/A

Table 4-797 NodePoolSpec

Parameter	Type	Description
type	String	Details: Node pool type Constraints: None Options: • vm: ECS • ElasticBMS: C6 general computing-plus BMS. An example flavor is c6.22xlarge.2.physical. • pm: BMS Default value: vm
nodeTemplate	NodeTemplate object	Details: Detailed parameters of a node pool template Constraints: None
initialNodeCount	Integer	Details: The initial number of nodes in a node pool. The obtained value is the desired number of nodes in the node pool. Constraints: None Options: Values that are greater than or equal to 0 and less than or equal to the cluster scale Default value: 0
autoscaling	NodePoolNodeAutoscaling object	Details: Auto scaling parameters Constraints: None
nodeManagement	NodeManagement object	Details: Node management configuration Constraints: None

Parameter	Type	Description
podSecurityGroups	Array of SecurityID objects	Details: Security group configuration Constraints: None
extensionScaleGroups	Array of ExtensionScaleGroup objects	Details: Configuration list of the extended scaling groups in a node pool. For details, see the definition of ExtensionScaleGroup . Constraints: None
customSecurityGroups	Array of strings	Details: Custom security group configuration for a node pool. New nodes added to a node pool can be bound to a specified security group. • Specifying no security group ID will add the new nodes to the default node security group. • Specifying a valid security group ID will put new nodes in that security group. • When specifying a security group, do not modify the port rules on which CCE depends. For details, see How Can I Configure a Security Group Rule in a Cluster? Constraints: None

Parameter	Type	Description
taintPolicyOnExistingNodes	String	Definition: Taint synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The taints of existing nodes in a node pool will not be updated synchronously. • refresh: The taints of existing nodes in a node pool will be updated synchronously. Default value: refresh
labelPolicyOnExistingNodes	String	Definition: Label synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The labels of existing nodes in a node pool will not be updated synchronously. • refresh: The labels of existing nodes in a node pool will be updated synchronously. Default value: refresh

Parameter	Type	Description
userTagsPolicyOnExistingNodes	String	Definition: Resource tag synchronization policy of existing nodes. Constraints: N/A Options: • ignore: The resource tags of existing nodes in a node pool will not be updated synchronously. • refresh: The resource tags of existing nodes in a node pool will be updated synchronously. Default value: ignore

Table 4-798 NodeTemplate

Parameter	Type	Description
flavor	String	Definition Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
az	String	Definition AZ where the node to be created is located. You need to specify the AZ name. Random AZs will not be allocated to the nodes created using APIs. For details about AZs supported by CCE, see Regions and Endpoints. Constraints When you are creating a node pool and configuring a scaling group, this parameter cannot be set to random. Range N/A Default Value N/A
os	String	Definition OS type of a node. For details about the supported OSs, see Node OSs. Constraints • If this parameter is not specified when you create a node, CCE automatically selects an OS version supported by the cluster version. • If the selected OS is not supported by the current cluster version, it will be automatically replaced with a supported OS of the same series. • If the alpha.cce/NodeImageID parameter in extendParam is specified during node creation, the node will use private images. In this case, this parameter is optional. Range N/A Default Value N/A

Parameter	Type	Description
login	Login object	Definition How to log in to a node. If this parameter is not specified, the password of the selected image will be kept. Ensure that you have set a password for the image. Constraints N/A
rootVolume	Volume object	Definition Disk information of a node Constraints N/A
dataVolumes	Array of Volume objects	Definition Data disk parameters of a node. For DeC nodes, the parameter definitions are the same as those in rootVolume . Constraints • A VM can have up to 16 disks attached, while a BMS can only have up to 10 disks attached. The maximum number of disks that can be attached to either a VM or BMS is limited by their respective maximum disk limits. (You can add more data disks to each of your nodes through the CCE console or APIs.) • The data disk used by the container runtime and kubelet cannot be detached, or the node will become unavailable. • This parameter can be left empty only when a system disk is selected for the system components.

Parameter	Type	Description
storage	Storage object	Details: Disk initialization management parameter. This parameter is complex to configure. For details, see Attaching Disks to a Node . Constraints: • This field is supported by clusters v1.15.11-r0 and later versions. If the cluster version is not supported, this field will be ignored. • If a node has both local and EVS disks attached, make sure to not keep the default value for this parameter, or it may result in unexpected disk partitions. • If you want to change the value range of a data disk to 20 to 32768, do not use the default value. • If you want to use the shared disk space (with the runtime and Kubernetes partitions cancelled), do not retain the default value of this parameter. For details about the shared disk space, see Space Allocation of a Data Disk. • If you want to store system components in the system disk, do not use the default value.
publicIP	NodeEIPSpec object	Definition EIP parameters of a node Constraints N/A Range N/A Default Value N/A
nodeNicSpec	NodeNicSpec object	Definition Network interfaces of a node Constraints N/A

Parameter	Type	Description
count	Integer	Definition Number of nodes that can be created at a time Constraints • This parameter can be left empty when it is used for a node pool. • This parameter is not returned during node pool creation or updates. • This parameter is mandatory during node creation. Range Positive integers that are greater than or equal to 1 and less than or equal to the maximum limit Default Value N/A
billingMode	Integer	Definition Billing mode of a node Constraints N/A Range • 0: pay-per-use • 1: yearly/monthly • 2: (Deprecated) yearly/monthly billing with auto payment enabled Default Value 0

Parameter	Type	Description
taints	Array of Taint objects	Definition Taints, which can be added to nodes for anti-affinity configuration. Each taint contains the following parameters: • Key: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key. • Value: Enter 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. • Effect: Available options are NoSchedule, PreferNoSchedule, and NoExecute. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"taints": [{ "key": "status", "value": "unavailable", "effect": "NoSchedule" }, { "key": "looks", "value": "bad", "effect": "NoSchedule" }]</pre> Constraints A maximum of 20 taints can be configured.

Parameter	Type	Description
waitPostInstallFinish	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during node creation. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Parameter	Type	Description
k8sTags	Map<String,String>	Definition Defined in key-value pairs. • Key: The key must contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key, for example, example.com/my-key, and contain a maximum of 253 characters. • Value: The value can be left empty or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Example: <pre>"k8sTags": { "key": "value" }</pre> Constraints A maximum of 20 key-value pairs is allowed.
ecsGroupId	String	Definition ECS group ID. If this parameter is specified, nodes will be created in the specific ECS group. Constraints This configuration does not take effect during node pool creation or updates. To ensure that all nodes in a node pool are in the same ECS group, configure nodeManagement in the node pool. Range N/A Default Value N/A

Parameter	Type	Description
dedicatedHostId	String	Definition ID of the DeH to which nodes will be scheduled Constraints This parameter is not supported when you add a node during node pool creation. Range N/A Default Value N/A
userTags	Array of UserTag objects	Definition Cloud server resource tags. When creating a node, you can specify an initial value for this field and this field is not returned during query. In node pools, the initial value can be specified in the node template and this field is returned during query. In other scenarios, this field is not returned during query. Constraints • The key of a tag must be unique. The maximum number of custom tags supported by CCE varies depending on regions and cannot exceed 8.
runtime	Runtime object	Definition Container runtime. • In clusters earlier than v1.25, the default value is docker. • In clusters of v1.25 or later, the default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default container runtime is docker. For nodes running other OSs, the default container runtime is containerd. Constraints N/A

Parameter	Type	Description
initializedConditions	Array of strings	Definition Custom initialization flag, which is left empty by default. Before CCE nodes are initialized, node.cloudprovider.kubernetes.io/uninitialized is added to the nodes to prevent pods from being scheduled to these nodes. When creating a node, you can configure the initializedConditions parameter to control the taint removal time. By default, the timeout period is not configured. Example: 1. Create a node and add "initializedConditions": ["CCEInitial", "CustomedInitial"]. 2. After custom initialization is complete, CCE will call a Kubernetes API (for example, PATCH /v1/nodes/{node_ip}/status) to update the node conditions and add two flags whose type values are CCEInitial and CustomedInitial, respectively, and both status values are True. 3. CCE polls status.Conditions of nodes to check whether there are conditions whose type is CCEInitial or CustomedInitial. If such conditions exist and their statuses are True, the node initialization is complete and the initialization taint will be removed. 4. initializedConditions allows you to configure a timeout interval during node creation. For example, "initializedConditions": ["CCEInitial:15m", "CustomedInitial:15m"] indicates that the timeout interval is 15 minutes. After the timeout interval is reached, the initialization conditions will be automatically ignored and the initialization taint will be removed when CCE polls the node. status: conditions:

Parameter	Type	Description
		- type: CCEInitial status: 'True' - type: CustomedInitial status: 'True' Constraints • The timeout interval in initializedConditions ranges from 1 to 99 in the unit of seconds. • Each value must be between 1 and 20 characters long and can only contain letters and digits. • The maximum number of flags cannot exceed 2. • The unit of the timeout interval is minute (m).
extendParam	NodeExtendParam object	Definition Extended parameters for creating a node Constraints N/A
hostnameConfig	HostnameConfig object	Definition Kubernetes node name configuration parameter Constraints The supported cluster version is v1.23.6-r0 to v1.25 or v1.25.2-r0 or later.

Parameter	Type	Description
serverEnterprise-ProjectID	String	Definition Enterprise project ID of a server. CCE does not support features provided by EPS. This field is used only for synchronizing enterprise project IDs of servers. Constraints To create a node or node pool, you can specify an existing enterprise project for it. If this field is left empty, it inherits the enterprise project attributes of the cluster that runs the node or node pool. When updating a node pool, any configuration modifications you made will apply only to new nodes. You need to modify the enterprise projects of existing nodes on the EPS console. Range N/A Default Value If no value is specified during the update, this field will not be updated. If this parameter is left empty, the cluster enterprise project is returned.
partition	String	Definition Zone that a node belongs to. It can be the central cloud or an edge site. Constraints This field can be specified only for CCE Turbo clusters that support distributed clouds. Range • center: central cloud • AZ ID of the edge site Default Value N/A

Parameter	Type	Description
nodeNameTemplate	nodeNameTemplate object	Definition Fixed prefix and suffix of node names. Assume that different node pools in a cluster are used by different departments of a company. You can use the prefix and suffix to distinguish the departments and how they use the nodes. For example, you can set nodeNamePrefix to finance- to indicate the department name, set nodeNameSuffix to -product to indicate the production usage, and set the node pool name to gpu to indicate the service type. The nodes will have names in the format of finance-gpu(five-random-digits)-product . Constraints This function is only supported by clusters of v1.28.1, v1.27.3, v1.25.6, v1.23.11, v1.21.12, or later. This parameter is only valid for node pools. Range N/A Default Value N/A

Table 4-799 Login

Parameter	Type	Description
sshKey	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	UserPassword object	Details: Password used for node login Constraints: None

Parameter	Type	Description
removeUserPassword	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false
removeSSHKey	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-800 UserPassword

Parameter	Type	Description
username	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Type	Description
password	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node. Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^-_=+[{ }];,./? • Not contain the username or the username spelled backwards. Default value: N/A

Table 4-801 Volume

Parameter	Type	Description
size	Integer	Details: Disk size, in GiB Constraints: None Options: • System disks: 40 to 1,024 • The first data disk: 20 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) • Other data disks: 10 to 32,768 (When storage is not specified, this value range becomes 100 to 32,768.) Default value: N/A

Parameter	Type	Description
volumetype	String	Details: Disk type. For details about possible values, see the root_volume parameter in the API used to create an ECS. Constraints: None Options: • SAS: high I/O SAS disks • SSD: ultra-high I/O SSD disks • SATA: common I/O SATA disks. SATA disks have been removed from EVS. You can find them attached only to existing nodes. NOTE For details about disk types, see Disk Types and Performance. Default value: N/A
extendParam	Map<String,Object>	Details: Extended disk parameters, defined in extendparam in the API used to create an ECS. For details, see extendparam Field Description for Creating Disks. Constraints: None
cluster_id	String	Details: ID of the storage pool used by the ECS system disk. This field is used only for DeC clusters and functions as dssPoolID, that is, the ID of the DSS storage pool. To obtain the value, see the ID field in the API used to obtain details of a DSS storage pool. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
cluster_type	String	Details: Storage class of the ECS system disk Constraints: None Options: This field is used only for DeC clusters, and the value is always dss . Default value: N/A
hw:passthrough	Boolean	Details: - Pay attention to this field if your ECS is SDI-compliant. If the value of this field is true, the created disk is of SCSI type. - If the node pool type is ElasticBMS, this field must be set to true. - If the node specification involves both local and EVS disks, configure the disk initialization parameters. For details, see Attaching Disks to a Node. Constraints: None Options: N/A Default value: N/A
metadata	VolumeMetadata object	Details: EVS disk encryption information. This parameter is mandatory only when the system disk or data disk of the node to be created needs to be encrypted. Constraints: None

Table 4-802 VolumeMetadata

Parameter	Type	Description
<code>__system__encrypted</code>	String	Details: Whether an EVS disk is encrypted. The value 0 indicates that the EVS disk is not encrypted, and the value 1 indicates that the EVS disk is encrypted. Constraints: None Options: N/A Default value: If this parameter does not appear, the disk is not encrypted.
<code>__system__cmkid</code>	String	Details: CMK ID, which indicates encryption in metadata . This field is used with <code>__system__encrypted</code> . Constraints: None Options: N/A Default value: N/A

Table 4-803 Storage

Parameter	Type	Description
<code>storageSelectors</code>	Array of StorageSelectors objects	Details: Disk selection. Matched disks are managed based on matchLabels and storageType settings. Disks are matched in sequence. The matching rule in the front is matched first. Constraints: None

Parameter	Type	Description
storageGroups	Array of StorageGroups objects	Details: A storage group consisting of multiple storage devices, which are used to divide storage space. Constraints: None

Table 4-804 StorageSelectors

Parameter	Type	Description
name	String	Details: Name of a selector, which is used as the index of selectorNames in storageGroup . Constraints: The name of each selector must be unique. Options: N/A Default value: N/A

Parameter	Type	Description
storageType	String	Details: Storage type Constraints: - The local storage type does not support disk selection. All local disks form a VG. Therefore, only one local storage selector is allowed. - The system storage type does not support disk selection. It is recommended that you use the system disk to store system components. As a result, only one storage selector of the system type is allowed, and the name must be cceUse. Options: Only evs (EVS disks), local (local disks), and system (system disks) are supported. Default value: N/A
matchLabels	matchLabels object	Details: Matching field of an EVS disk Constraints: If storageType is set to system (system disk), you do not need to configure this parameter. Options: The size, volumeType, metadataEncrypted, metadataCmkid, and count fields in DataVolume are supported. Default value: N/A

Table 4-805 matchLabels

Parameter	Type	Description
size	String	Details: Matched disk size, for example, 100 . If this parameter is left blank, the disk size is not limited. Constraints: None Options: N/A Default value: N/A
volumeType	String	Definition EVS disk type. Supported types include SSD , GPSSD , SAS , ESSD , and SATA . If this parameter is left empty, there is no restriction on the EVS disk type. Constraints N/A Range N/A Default Value N/A
metadataEncrypted	String	Details: Disk encryption identifier. If this parameter is set to 0 , it specifies that the disk is not encrypted. If this parameter is set to 1 , it specifies that the disk is encrypted. If this parameter is left blank, there is no restriction on the disk encryption identifier. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
metadataCmkid	String	Details: CMK ID of an encrypted disk. The value is a string of 36 bytes. If this parameter is left blank, there is no restriction on the disk key ID. Constraints: None Options: N/A Default value: N/A
count	String	Details: Number of disks to be selected. If this parameter is left blank, all disks of this type are selected. Constraints: None Options: N/A Default value: N/A

Table 4-806 StorageGroups

Parameter	Type	Description
name	String	Details: Name of a virtual storage group, which must be unique Constraints: None Options: - If cceManaged is set to true, the name field must be set to vgpaas. - If the data disk is used as a temporary storage volume, the name field must be set to vg-everest-localvolume-ephemeral. - If the data disk is used as a persistent storage volume, the name field must be set to vg-everest-localvolume-persistent. Default value: N/A
cceManaged	Boolean	Details: Storage space to which Kubernetes and runtime belong. Only one group is set to true . If this parameter is left blank, the default value false is used. Constraints: None Options: N/A Default value: N/A
selectorNames	Array of strings	Details: Corresponds to name in storageSelectors . One group can select multiple selectors, but one selector can be selected by only one group. Constraints: System components cannot be separately stored in the system disk and data disks. Therefore, when the selector type is set to system , only one selector can be selected for a group.

Parameter	Type	Description
virtualSpaces	Array of VirtualSpace objects	Details: Detailed management of space configurations in a group Constraints: None

Table 4-807 VirtualSpace

Parameter	Type	Description
name	String	Details: Name of a virtualSpace. Only types of share , kubernetes , runtime , and user are supported. Constraints: None Options: • Kubernetes: Kubernetes space configuration. lvmConfig needs to be configured. • runtime: runtime space configuration. runtimeConfig needs to be configured. • user: user space configuration. lvmConfig needs to be configured. Default value: N/A
size	String	Details: Size of a virtualSpace. The value must be an integer in percentage, for example, 90%. Constraints: None Options: 10 to 90. The sum of percentages of all virtualSpaces in a group cannot exceed 100%. Default value: N/A

Parameter	Type	Description
lvmConfig	LVMConfig object	Details: LVM configuration management, which is applicable to share, Kubernetes, and user space configuration Constraints: One virtualSpace supports only one config.
runtimeConfig	RuntimeConfig object	Details: Runtime configuration management, which is applicable to runtime space configuration Constraints: One virtualSpace supports only one config.

Table 4-808 LVMConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
path	String	Details: Disk mount path. This parameter only takes effect in user configuration. The value can be an absolute path containing digits, uppercase letters, lowercase letters, periods (.), hyphens (-), and underscores (_). Constraints: Do not leave the path empty or set it to a key OS path such as the root directory. Key OS paths that cannot be entered: /, /home, /home/, /bin, /bin/, /lib, /lib/, /root, /root/, /boot, /boot/, /dev, /dev/, /etc, /etc/, /lost+found, /lost+found/, /mnt, /mnt/, /proc, /proc/, /sbin, /sbin/, /srv, /srv/, /tmp, /tmp/, /var, /var/, /media, /media/, /opt, /opt/, /selinux, /selinux/, /sys, /sys/, /usr, /usr/, /opt/cloud/, /mnt/paas/, /home/paas/, /var/paas/, /var/lib/, /var/script/ Options: N/A Default value: N/A

Table 4-809 RuntimeConfig

Parameter	Type	Description
lvType	String	Details: LVM write mode. linear indicates the linear mode. striped indicates the striped mode, in which multiple disks are used to form a strip to improve disk performance. Constraints: None Options: N/A Default value: N/A

Table 4-810 NodeEIPSpec

Parameter	Type	Description
iptype	String	Details: EIP type, specified in publicip.type in the API for assigning an EIP. For details, see Assigning an EIP . Constraints: None Options: N/A Default value: N/A
bandwidth	NodeBandwidth object	Details: Bandwidth parameters of the EIP Constraints: None

Table 4-811 NodeBandwidth

Parameter	Type	Description
chargemode	String	Details: Bandwidth billing mode NOTE • Billed by bandwidth: The billing will be based on the data transmission rate (in Mbps) of public networks. This billing mode is recommended if your bandwidth usage is higher than 10%. • Billed by traffic: The billing is based on the total amount of data (in GB) transmitted over the public network. This mode is available only when you are creating a pay-per-use node. This billing mode is recommended if your bandwidth usage is lower than 10%. Constraints: None Options: • If this field is not specified, the billing is based on bandwidth. • If the field value is empty, the billing is based on bandwidth. • If the field value is traffic, the billing is based on traffic. • If the value is out of the preceding options, the cloud server will fail to be created. Default value: N/A
size	Integer	Details: Bandwidth size, specified in bandwidth.size in the API for assigning an EIP. For details, see Assigning an EIP. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
sharetype	String	Details: Bandwidth sharing type. Dedicated bandwidth is supported only. PER specifies dedicated bandwidth. Constraints: None Options: N/A Default value: N/A

Table 4-812 NodeNicSpec

Parameter	Type	Description
primaryNic	NicSpec object	Details: Description of the primary ENI Constraints: None
extNics	Array of NicSpec objects	Details: Extension ENI Constraints: This parameter is not supported when you add a node during node pool creation.

Table 4-813 NicSpec

Parameter	Type	Description
subnetId	String	Details: Network ID of the subnet to which an ENI belongs. If subnetList is also configured for a node pool, the subnetList field is used for adding subnets to the node pool. Constraints: • If subnetId is not specified when a primary ENI is creating, the cluster subnet will be used. • When creating an extended ENI, you must specify subnetId. Options: N/A Default value: N/A
fixedIps	Array of strings	Definition The IP address of a primary network interface is specified using fixedIps. The number of IP addresses cannot be greater than the number of created nodes. Constraints • Either fixedIps or ipBlock can be specified. • fixedIps cannot be specified for extended network interfaces. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools.

Parameter	Type	Description
ipBlock	String	Definition CIDR format of the IP address range of the primary network interface. The IP address of a created node falls in this range. Constraints • Either fixedIps or ipBlock can be specified. • This parameter cannot be configured during node pool creation. • This parameter is not present in the responses related to node pools. Range N/A Default Value N/A
subnetList	Array of strings	Details: Network ID list of the subnet where the ENI resides. Multiple subnets can be configured for a node pool. Constraints: A maximum of 20 subnets can be configured.

Table 4-814 Taint

Parameter	Type	Description
key	String	Definition Key of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A

Parameter	Type	Description
value	String	Definition Value of a Kubernetes taint Constraints The value must contain 1 to 63 characters starting and ending with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Range N/A Default Value N/A
effect	String	Definition Effect of a Kubernetes taint Constraints N/A Range NoSchedule, PreferNoSchedule, and NoExecute Default Value N/A

Table 4-815 UserTag

Parameter	Type	Description
key	String	Details: Key of the cloud server tag Constraints: The value cannot start with CCE- , __type_baremetal , or sys . Options: The value contains a maximum of 128 characters. The tag key must be unique and cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<>, / Default value: N/A

Parameter	Type	Description
value	String	Details: Value of the cloud server tag Constraints: None Options: The value contains a maximum of 255 characters. The tag value cannot start or end with a space. It cannot contain non-printable ASCII characters (0-31) or the following special characters: =*<> Default value: N/A

Table 4-816 Runtime

Parameter	Type	Description
name	String	Details: Container runtime Constraints: None Options: N/A Default value: • Clusters earlier than v1.25: The default value is docker. • Clusters of v1.25 or later: The default value varies with the OSs. • For nodes running EulerOS 2.5 or EulerOS 2.8, the default value is docker. For nodes running other OSs, the default value is containerd.

Table 4-817 NodeExtendParam

Parameter	Type	Description
ecs:performancetype	String	Details: ECS flavor types. This field is returned in the response. Constraints: None Options: N/A Default value: N/A
orderId	String	Definition Order ID Constraints This parameter is included in the responses only during node creation if the node is billed on a yearly/monthly basis with auto payment enabled. It is not returned in the responses related to node pools. Range N/A Default Value N/A
productId	String	Definition Product ID Constraints This parameter is included in the responses if the node is billed on a yearly/monthly basis with auto payment enabled or the node pool is billed on a yearly/monthly basis. Range N/A Default Value N/A

Parameter	Type	Description
maxPods	Integer	Details: Maximum number of pods that can be created on a node, including the default system pods. This limit prevents the node from being overloaded with pods. The number of pods that can be created on a node is determined by multiple parameters. For details, see Maximum Number of Pods That Can Be Created on a Node. Constraints: None Options: 16 to 256 Default value: N/A
periodType	String	Details: Subscription term unit Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - month - year Default value: N/A

Parameter	Type	Description
periodNum	Integer	Details: Number of subscription periods Constraints: - As a request parameter, it is valid and mandatory only when billingMode is set to 1 (yearly/monthly billing) or 2 (yearly/monthly billing with auto payment enabled) (This option has been deprecated). - As a response parameter, it is returned only when a yearly/monthly node is created. Options: - If periodType is month, the value ranges from 1 to 9. - If periodType is year, the value ranges from 1 to 3. Default value: N/A
isAutoRenew	String	Details: Whether auto-renewal is enabled Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified, auto-renewal is not enabled. Options: true: Auto-renewal is enabled. false: Auto-renewal is not enabled. Default value: N/A

Parameter	Type	Description
isAutoPay	String	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1, auto payment is not enabled. If not specified when billingMode is set to 2 (deprecated), auto payment is enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Parameter	Type	Description
DockerLVMConfigOverride	String	Details: Docker data disk configuration item. (This parameter has been deprecated. Use the storage field instead.) Example of the default configuration: <pre>"DockerLVMConfigOverride": {"dockerThinpool=vgpaas/90%VG;kubernetesLV=vgpaas/10%VG;diskType=evs;lvType=linear"}</pre> By default, if no VD disk is available, an error occurs because the data disk fails to be found. Specify diskType based on the actual drive letter type. Constraints: None Options: The following fields are included: • userLV: (Optional) size of the user space, for example, vgpaas/20%VG • userPath (Optional): mount path of the user space, for example, /home/wqt-test • diskType: disk type. Only evs, hdd, and ssd are supported. • lvType: type of a logic volume. The value can be linear or striped. • dockerThinpool: Docker space size, for example, vgpaas/60%VG • kubernetesLV: kubelet space size, for example, vgpaas/20%VG Default value: N/A

Parameter	Type	Description
dockerBaseSize	Integer	Definition Available disk space of a single container on a node (unit: G). This parameter has been deprecated. Use containerBaseSize. For details about how to allocate the space for the container runtime, see Data Disk Space Allocation. Constraints - When Device Mapper is used, it is recommended that dockerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount external or local storage. - The dockerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0. Range 10 to 500 Default Value If the value is not specified or is set to 0, the default value is used. - The default value is 10 in Device Mapper mode. - In OverlayFS mode, no fixed upper limit is imposed on the available space of a single container.

Parameter	Type	Description
containerBaseSize	Integer	Details: Available disk space of a single container on a node (unit: G). For details about how to allocate the space for the container runtime, see Space Allocation of a Data Disk. Constraints: When Device Mapper is used, it is recommended that containerBaseSize be set to a value less than or equal to 80 GiB. If the value is too large, the container runtime may fail to be started due to long initialization. If there are special requirements for the container disk space, you can mount an external or local storage device. In new versions, the Device Mapper mode is used only by BMSs in the same resource pool and is being deprecated. • The containerBaseSize setting takes effect only on nodes running EulerOS or Huawei Cloud EulerOS 2.0 in clusters of the new version (v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0 or later). • This parameter cannot be updated when a node pool is being updated. Options: 10 to 500 Default value: If the value is not specified or is set to 0, the default value is used. • The default value is 10 in Device Mapper mode. • In OverlayFS mode, the available space of a single container is not limited by default.

Parameter	Type	Description
publicKey	String	Definition Public key of a node, which is used for SSH key login Constraints - This parameter is invalid if you use a key pair to log in to the node. - This parameter cannot be configured during node pool creation and updates. Range N/A Default Value N/A
alpha.cce/ preInstall	String	Details: Pre-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A

Parameter	Type	Description
alpha.cce/postInstall	String	Details: Post-installation script. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "**Content to be encoded*" base64</pre> Constraints: The characters of both the pre-installation and post-installation scripts are centrally calculated, and the total number of characters after transcoding cannot exceed 10,240. Options: N/A Default value: N/A
alpha.cce/NodeImageID	String	Definition Private image ID of a node, which is obtained from the IMS console. This parameter is required when private images are used. Constraints N/A Range N/A Default Value N/A
chargingMode	Integer	Details: Billing mode of a node. This parameter has been deprecated. Use billingMode in NodeSpec instead. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
marketType	String	Details: When creating a spot ECS, this parameter should be set to spot. Constraints: This parameter is valid only when billingMode is set to 0. Options: N/A Default value: N/A
spotPrice	String	Details: The highest price per hour a user accepts for a spot ECS Constraints: • This parameter takes effect only when billingMode is set to 0 and marketType is set to spot. • When billingMode is set to 0 and marketType is set to spot, if spotPrice is not specified, the pay-per-use price is used by default. • The spotPrice value must be less than or equal to the pay-per-use price and greater than or equal to the ECS market price. Options: N/A Default value: N/A

Parameter	Type	Description
agency_name	String	Details: Agency name. An agency is created by a tenant administrator on Identity and Access Management (IAM) to provide temporary credentials for CCE nodes to access cloud servers. This parameter is returned only when it is transferred during node creation. Constraints: None Options: N/A Default value: N/A
kubeReservedMem	Integer	Details: Reserved node memory, which is reserved for Kubernetes components. Constraints: None Options: N/A Default value: N/A
systemReservedMem	Integer	Details: Reserved node memory, which is reserved for system components. Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
init-node-password	String	Definition Node password. If this parameter is used only in a response, asterisks (*) are displayed. It is not returned in the responses related to node pools. Constraints N/A Range N/A Default Value N/A

Table 4-818 HostnameConfig

Parameter	Type	Description
type	String	Details: Configuration type of the Kubernetes node name. The default value is privatelp. Constraints: • The node names, Kubernetes node names, and VM names of the nodes configured with cceNodeName are the same. Node names cannot be changed. Once a VM name is altered on the ECS console, the node name cannot be modified accordingly during ECS synchronization. • To prevent conflicts between Kubernetes node names, a suffix is automatically added to the node name of a node configured with cceNodeName. The suffix consists of a hyphen (-) followed by five random lowercase letters and digits (0 to 9). Options: • privatelp: Use the private node IP address as the Kubernetes node name. • cceNodeName: Use the CCE node name as the Kubernetes node name. Default value: privatelp

Table 4-819 nodeNameTemplate

Parameter	Type	Description
nodeNamePrefix	String	Definition Node name prefix. If this parameter is left empty or set to the default value, no prefix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The value must start with a lowercase letter and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A
nodeNameSuffix	String	Definition Node name suffix. If this parameter is left empty or set to the default value, no suffix will be added to the node name. Constraints Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. The suffix must end with a lowercase letter or digit and comply with the DNS subdomain naming rules defined in RFC 1123 . Range N/A Default Value N/A

Table 4-820 NodePoolNodeAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
minNodeCount	Integer	Details: Minimum number of nodes that can be scaled in if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be added if auto scaling is enabled Constraints: None Options: Values that are greater than or equal to the value of minNodeCount and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0

Parameter	Type	Description
scaleDownCoolDownTime	Integer	Details: How long nodes will be kept, in minutes. During this period, nodes added will not be deleted. Constraints: None Options: 0 to 2147483646 Default value: 0
priority	Integer	Details: Weight of a node pool. A node pool with a higher weight has a higher priority during scale-out. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
scaleDownUnneededTime	Integer	Details: How long a node should be unneeded before it is eligible for scale in. This parameter specifies the duration during which a node is in the unneeded state before the scale-in operation can be performed. CCE Cluster Autoscaler scales in the node only when the node is not needed for a specified period. This prevents frequent scale-in triggered by temporary resource fluctuations and improves system stability. If this parameter is not specified, the CCE Cluster Autoscaler add-on uses the default time threshold. Constraints: None Options: 0 to 2147483647 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Parameter	Type	Description
scaleDownUtilizationThreshold	Float	Details: Scale-in utilization threshold of a node, which is defined by the total CPU or memory resources used by all running pods on the node divided by the total available resources of the node. If the calculated value falls below the specified threshold, the node becomes eligible for scale-in. For example, setting this parameter to 0.3 means that if the resource utilization on a node drops below 30%, the node will be scaled in. If this parameter is not specified, the CCE Cluster Autoscaler add-on will use the default utilization threshold. Constraints: None Options: 0 to 1 NOTE Note: If the parameter is set to -1, it indicates that the value is empty. Default value: This parameter is left empty by default.

Table 4-821 NodeManagement

Parameter	Type	Description
serverGroupReference	String	Details: A cloud server group ID. If this parameter is specified, the cloud server group will be associated with the target node pool, and all nodes in the node pool will be created in this cloud server group. After a cloud server group is associated with a node pool, the number of nodes in the node pool cannot exceed the number of cloud servers that can be added to the cloud server group, or the node pool cannot be scaled out. NOTE • After a cloud server group is associated with a node pool, the cloud servers added to the cloud server group will be distributed in strict accordance with the affinity policy and the maximum number of nodes in the node pool will be limited. The anti-affinity capabilities are available during cloud server creation on ECS. If your goal is to enhance service reliability by distributing cloud servers across different hosts and you do not want the number of nodes to be restricted by cloud server groups, you are advised not to associate cloud server groups with any node pools. • A cloud server group can be disassociated from the node pool. After the change, the existing nodes will remain in the original cloud server group, and no new nodes will be added to the cloud server group. If the node pool is empty, you can associate a new cloud server group with the node pool or change the associated cloud server group. Constraints: If a cloud server group is specified, the number of nodes in the node pool cannot exceed the group quota. Options: • If this parameter is not specified or left blank, the cloud server group will be disassociated from the node pool.

Parameter	Type	Description
		A cloud server group ID: The cloud server group associated with the node pool will be changed to a specified one. Default value: N/A

Table 4-822 SecurityID

Parameter	Type	Description
id	String	Definition ID of the container security group on a node Constraints N/A Range N/A Default Value N/A

Table 4-823 ExtensionScaleGroup

Parameter	Type	Description
metadata	ExtensionScaleGroupMetadata object	Details: Basic information about an extended scaling group Constraints: None
spec	ExtensionScaleGroupSpec object	Details: Configurations of an extended scaling group, which carry different configurations from those of the default scaling group Constraints: None

Table 4-824 ExtensionScaleGroupMetadata

Parameter	Type	Description
uid	String	Details: UUID of an extended scaling group, which is automatically generated by the system Constraints: None Options: N/A Default value: N/A
name	String	Details: Name of an extended scaling group Constraints: The value cannot be default . Options: Enter 1 to 56 characters starting with a lowercase letter and ending with a lowercase letter or digit. Only digits, lowercase letters, and hyphens (-) are allowed. Default value: N/A

Table 4-825 ExtensionScaleGroupSpec

Parameter	Type	Description
flavor	String	Details: Node flavors. For details about the node flavors supported by CCE clusters, see Node Specifications . Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
az	String	Details: Node AZ Constraints: None Options: N/A Default value: If this parameter is not specified or left blank, the default scaling group configurations take effect.
capacityReservationSpecification	CapacityReservationSpecification object	Details: Capacity reservation configurations of an extended scaling group Constraints: None
autoscaling	ScaleGroupAutoscaling object	Details: Auto scaling configurations of an extended scaling group Constraints: None

Table 4-826 CapacityReservationSpecification

Parameter	Type	Description
id	String	Details: Private pool ID Constraints: This value can be ignored when preference is set to none . Options: N/A Default value: N/A

Parameter	Type	Description
preference	String	Details: Private pool capacity Constraints: If the value is none , the capacity reservation is not specified. If the value is targeted , the capacity reservation is specified. In this case, the ID cannot be left blank. Options: N/A Default value: none

Table 4-827 ScaleGroupAutoscaling

Parameter	Type	Description
enable	Boolean	Details: Whether to enable auto scaling for a scaling group Constraints: None Options: • false: Auto scaling is disabled. • true: Auto scaling is enabled. Default value: false
extensionPriority	Integer	Details: Scaling group priority. A larger value indicates a higher priority. Constraints: None Options: N/A Default value: 0

Parameter	Type	Description
minNodeCount	Integer	Details: Minimum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to 0 and cannot be greater than the maximum number of nodes allowed by the cluster specifications Default value: 0
maxNodeCount	Integer	Details: Maximum number of nodes that can be kept in a scaling group during auto scaling Constraints: None Options: Values that are greater than or equal to the value of minNodeCount, less than or equal to the maximum number of nodes allowed by the cluster specifications, and less than or equal to the maximum number of nodes in the node pool Default value: 0

Table 4-828 DeleteNodePoolStatus

Parameter	Type	Description
currentNode	Integer	Details: Total number of nodes in the current node pool (excluding the nodes that are being deleted) Constraints: None Options: N/A Default value: N/A
creatingNode	Integer	Details: Number of nodes that are being created in the current node pool Constraints: None Options: N/A Default value: N/A
deletingNode	Integer	Details: Number of nodes that are being deleted in the current node pool Constraints: None Options: N/A Default value: N/A
configurationSyncedNodeCount	Integer	Details: The number of nodes specified in the node pool configuration has been synchronized in the current node pool. Constraints: None Options: None Default value: N/A

Parameter	Type	Description
phase	String	Details: Node pool status Constraints: None Options: • Null: The node pool is available. (The number of nodes in the current node pool has reached the expected value, and no node scaling is being performed.) • Synchronizing: The node pool is being scaled. (The number of nodes in the current node pool has not reached the expected value, and no node scaling is being performed.) • Synchronized: The node pool scaling is pending. (The number of nodes in the current node pool has not reached the expected value, and node scaling is being performed.) • SoldOut: The node pool cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The node pool is being deleted. • Error: An error occurred. NOTE The preceding node pool statuses have been deprecated. They are reserved only for compatibility. Do not use them anymore. Use the following instead: • Node pool scaling status: You can obtain the status of the current node pool using parameters such as currentNode, creatingNode, or deletingNode. • Node pool scale-out: You can use conditions to obtain the detailed status of a node pool. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
jobId	String	Details: ID of a job executed on the node pool Constraints: None Options: N/A Default value: N/A
conditions	Array of NodePoolCondition objects	Details: Details of node pool statuses. For details, see the definition of Condition . Constraints: None
scaleGroupStatuses	Array of ScaleGroupStatus objects	Details: Detailed status of a scaling group. For details, see the definition of ScaleGroupStatus . Constraints: None

Table 4-829 ScaleGroupStatus

Parameter	Type	Description
name	String	Details: Scaling group name Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
uid	String	Details: Scaling group UUID Constraints: None Options: N/A Default value: N/A
creationTimestamp	String	Details: The time when a scaling group was created Constraints: None Options: N/A Default value: N/A
updateTimestamp	String	Details: The time when a scaling group was updated Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
phase	String	Details: Scaling group status Constraints: None Options: • Null: The scaling group is available. (The number of nodes in the current scaling group has reached the expected value, and no node scaling is being performed.) • Synchronizing: The scaling group is being scaled. (The number of nodes in the current scaling group has not reached the expected value, and no node scaling is being performed.) • Synchronized: The scaling is pending. (The number of nodes in the current scaling group has not reached the expected value, or node scaling is being performed.) • SoldOut: The scaling group cannot be scaled out. (This field is used in multiple scenarios, for example, resources in the node pool have been sold out or the resource quota is insufficient.) • Deleting: The scaling group is being deleted. • Error: An error occurred. NOTE The preceding scaling group status has been deprecated. It is reserved only for compatibility. Do not use it anymore. Use the following instead: • Scaling status of a scaling group: You can obtain the status of the current scaling group using parameters such as desiredNodeCount, existingNodeCount, or upcomingNodeCount. • Scaling group scale-out: You can use conditions to obtain the detailed status of a scaling group. Scalable can replace SoldOut. Default value: N/A

Parameter	Type	Description
desiredNodeCount	Integer	Details: Expected number of nodes in a scaling group Constraints: None Options: N/A Default value: N/A
unpaidScaleNodeCount	Integer	Details: Number of nodes in an unpaid order Constraints: None Options: N/A Default value: N/A
existingNodeCount	existingNodeCount object	Details: Statistics on existing nodes in a scaling group Constraints: None
upcomingNodeCount	upcomingNodeCount object	Details: Statistics on the nodes to be added in a scaling group Constraints: None
scaleDownDisabledNodeCount	Integer	Details: Number of nodes that cannot be deleted in a scaling group Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
conditions	Array of NodePoolCondition objects	Details: Scaling group status details. For details, see the definition of Condition . Constraints: None

Table 4-830 existingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-831 upcomingNodeCount

Parameter	Type	Description
postPaid	Integer	Details: Number of pay-per-use nodes Constraints: None Options: N/A Default value: N/A
prePaid	Integer	Details: Number of yearly/monthly nodes Constraints: None Options: N/A Default value: N/A
total	Integer	Details: Total number of pay-per-use and yearly/monthly nodes Constraints: None Options: N/A Default value: N/A

Table 4-832 NodePoolCondition

Parameter	Type	Description
type	String	Definition Status type Constraints N/A Range • TaintSynchronizing: The node pool is synchronizing Kubernetes taints for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • LabelSynchronizing: The node pool is synchronizing Kubernetes labels for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • UserTagsSynchronizing: The node pool is synchronizing Kubernetes resource tags for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • ConfigurationSynchronizing: The node pool is synchronizing settings for its nodes. This status does not impact the scale-out behavior of the node pool. (It only applies to node pools and is not present in the scaling groups.) • Scalable: specifies whether a node pool/scaling group can be scaled out. If the status is False, node pool scale-out will not be triggered again. • QuotaInsufficient: Quotas on which node pool/scaling group scale-out depends are insufficient, affecting the node pool scale-out status. • ResourceInsufficient: Resources on which node pool/scaling group

Parameter	Type	Description
		scale-out depends are insufficient, affecting the node pool scale-out status. • UnexpectedError: The node pool/scaling group fails to be scaled out due to unexpected reasons, affecting the node pool scale-out status. • LockedByOrder: The node pool/scaling group is locked due to a pending order. The value of Reason is the ID of the order. • Error: There is an error in the node pool/scaling group. The error is typically triggered by a deletion failure. Default Value N/A
status	String	Details: Status of the condition Constraints: None Options: • True • False Default value: N/A
lastProbeTime	String	Details: Time when the status was last checked Constraints: None Options: N/A Default value: N/A

Parameter	Type	Description
lastTransitTime	String	Details: Time when the status was last changed Constraints: None Options: N/A Default value: N/A
reason	String	Details: Reason why the status was last changed Constraints: None Options: N/A Default value: N/A
message	String	Details: Detailed description about the condition Constraints: None Options: N/A Default value: N/A

Example Requests

None

Example Responses

Status code: 200

The job for deleting the node pool is successfully delivered.

```
{
  "kind" : "NodePool",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "lc-it-nodepool-79796",
```

```
"uid" : "99addaa2-69eb-11ea-a592-0255ac1001bb"
},
"spec" : {
  "type" : "vm",
  "nodeTemplate" : {
    "flavor" : "s6.large.2",
    "az" : "*****",
    "os" : "EulerOS 2.5",
    "login" : {
      "sshKey" : "KeyPair-001"
    },
    "rootVolume" : {
      "volumetype" : "SAS",
      "size" : 40
    },
    "dataVolumes" : [ {
      "volumetype" : "SAS",
      "size" : 100,
      "extendParam" : {
        "useType" : "docker"
      }
    } ],
    "publicIP" : {
      "eip" : {
        "bandwidth" : { }
      }
    },
    "nodeNicSpec" : {
      "primaryNic" : {
        "subnetId" : "7e767d10-7548-4df5-ad72-aeac1d08bd8a"
      }
    },
    "billingMode" : 0,
    "extendParam" : {
      "maxPods" : 110
    },
    "k8sTags" : {
      "cce.cloud.com/cce-nodepool" : "lc-it-nodepool-79796"
    }
  },
  "autoscaling" : { },
  "nodeManagement" : { }
},
"status" : {
  "phase" : "Deleting",
  "jobId" : "3281fa02-69ee-11ea-a592-0255ac1001bb"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class DeleteNodePoolSolution {
```



```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");
    String projectId = "{project_id}";

    ICredential auth = new BasicCredentials()
        .withProjectId(projectId)
        .withAk(ak)
        .withSk(sk);

    CceClient client = CceClient.newBuilder()
        .withCredential(auth)
        .withRegion(CceRegion.valueOf("<YOUR REGION>"))
        .build();
    DeleteNodePoolRequest request = new DeleteNodePoolRequest();
    request.withClusterId("{cluster_id}");
    request.withNodepoolId("{nodepool_id}");
    try {
        DeleteNodePoolResponse response = client.deleteNodePool(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteNodePoolRequest()
```

```
request.cluster_id = "{cluster_id}"
request.nodepool_id = "{nodepool_id}"
response = client.delete_node_pool(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteNodePoolRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    response, err := client.DeleteNodePool(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for deleting the node pool is successfully delivered.

Error Codes

See [Error Codes](#).

4.4.6 Scaling a Node Pool

Function

This API is used to scale in or out a node pool.

NOTE

The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** specifies the resource path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/operation/scale

Table 4-833 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-834 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-835 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	Details: API type Constraints: None Options: The value is fixed at NodePool . Default value: N/A
apiVersion	Yes	String	Details: API version Constraints: None Options: The value is fixed at v3 . Default value: N/A
spec	Yes	ScaleNodePoolSpec object	Details: Parameters in the request for scaling a node pool Constraints: None

Table 4-836 ScaleNodePoolSpec

Parameter	Mandatory	Type	Description
desiredNodeCount	Yes	Integer	Details: Desired number of nodes in a node pool Constraints: This parameter is mandatory. If it is omitted, the default value 0 is used. This will result in the deletion of all nodes in the node pool scaling group. Options: 0 or a positive integer Default value: 0
scaleGroups	Yes	Array of strings	Definition: Name of the scaling group in a node pool to be scaled. Constraints: To use the default scaling group, set the parameter to default . Options: N/A Default value: N/A
options	No	ScaleNodePoolOptions object	Details: Configurations of node pool scaling options Constraints: None

Table 4-837 ScaleNodePoolOptions

Parameter	Mandatory	Type	Description
scalableChecking	No	String	Details: Policy for checking the scale-out Constraints: None Options: • instant: When a scale-out request is delivered, the system synchronously checks whether the underlying resources are sold out. • async: When a scale-out request is delivered, the system asynchronously checks whether the underlying resources are sold out. Default value: instant

Parameter	Mandatory	Type	Description
scalePolicy	No	String	Definition: Scale-out policy. This parameter can be left empty and is only valid when multiple scaleGroups values are transferred. Constraints: N/A Options: • AZBalance: The system attempts to balance the number of nodes in each AZ during node pool scale-out. The flavor of each node is randomly specified in the selected scaling group. This policy simplifies configuration and reduces individual faults. It is suitable if there are no special requirements regarding node costs and AZs. Note: If resources in an AZ are insufficient, the nodes that should be added in this AZ will be added in other AZs, which may cause uneven node distribution among AZs. If this issue occurs, you can add nodes in this AZ again when resources in the AZ are sufficient. • Random: The system randomly selects a scaling group from the scaleGroups list for scale-out. Default value: Random

Parameter	Mandatory	Type	Description
billingConfigOverride	No	ScaleUpBillingConfigOverride object	Details: The default billing mode of a node pool will be overwritten when the node pool is scaled out. If the scale-out of yearly/monthly nodes is needed in a new node pool, this parameter is required. In other scenarios, you are advised not to use this parameter. Constraints: Only clusters v1.21.11-r0, v1.23.9-r0, v1.25.4-r0, and later versions support the scale-out of yearly/monthly nodes in pay-per-use node pools.

Table 4-838 ScaleUpBillingConfigOverride

Parameter	Mandatory	Type	Description
billingMode	No	Integer	Details: Billing mode of a node Constraints: This parameter is optional. If it is not specified, the default billing mode of the node pool where the node is in is used. Options: • 0: pay-per-use • 1: yearly/monthly Default value: N/A

Parameter	Mandatory	Type	Description
extendParam	No	ScaleUpExtendParam object	Details: Parameters for configuring the yearly/monthly billing mode Constraints: • This parameter is not considered in the pay-per-use billing mode. • This parameter is mandatory for the yearly/monthly billing mode.

Table 4-839 ScaleUpExtendParam

Parameter	Mandatory	Type	Description
periodNum	Yes	Integer	Details: Yearly/Monthly duration Constraints: None Options: • If periodType is month, the value ranges from 1 to 9. • If periodType is year, the value ranges from 1 to 3. Default value: N/A
periodType	Yes	String	Details: Yearly/Monthly type Constraints: None Options: • year • month Default value: N/A

Parameter	Mandatory	Type	Description
isAutoRenew	No	Boolean	Details: Whether to enable auto renewal Constraints: It is valid when billingMode is set to 1 . If it is left blank, auto renewal is not enabled by default. Options: • true: Auto renewal is enabled. • false: Auto renewal is not enabled. Default value: N/A
isAutoPay	No	Boolean	Details: Whether an order is automatically or manually paid Constraints: This field is valid when billingMode is set to 1 or 2 (deprecated). If not specified when billingMode is set to 1 , auto payment is not enabled. Options: • true: Auto payment is enabled. • false: Auto payment is not enabled. Default value: N/A

Response Parameters

Status code: 202

Table 4-840 Response body parameters

Parameter	Type	Description
orderId	String	Definition Order ID, which is returned only when yearly/monthly nodes are added Range N/A

Example Requests

- Scale out nodes in the default scaling group of a node pool (pay-per-use).

```
{
  "kind": "NodePool",
  "apiVersion": "v3",
  "spec": {
    "desiredNodeCount": 1,
    "scaleGroups": [ "default" ]
  }
}
```

- Scale out nodes in the default scaling group of a node pool (yearly/monthly).

```
{
  "kind": "NodePool",
  "apiVersion": "v3",
  "spec": {
    "desiredNodeCount": 1,
    "scaleGroups": [ "default" ],
    "options": {
      "billingConfigOverride": {
        "billingMode": 1,
        "extendParam": {
          "periodNum": 1,
          "periodType": "month",
          "isAutoRenew": false,
          "isAutoPay": false
        }
      }
    }
  }
}
```

Example Responses

Status code: 202

The node pool scaling is accepted. Nodes in the node pool will be added or deleted based on the expected number of nodes in the node pool after scaling.

```
{
  "orderId": "CS2504231542LJVS6"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

- Scale out nodes in the default scaling group of a node pool (pay-per-use).

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class ScaleNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        ScaleNodePoolRequest request = new ScaleNodePoolRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        ScaleNodePoolRequestBody body = new ScaleNodePoolRequestBody();
        List listSpecScaleGroups = new ArrayList<>();
        listSpecScaleGroups.add("default");
        ScaleNodePoolSpec specbody = new ScaleNodePoolSpec();
        specbody.withDesiredNodeCount(1)
            .withScaleGroups(listSpecScaleGroups);
        body.withSpec(specbody);
        body.withApiVersion("v3");
        body.withKind("NodePool");
        request.withBody(body);
        try {
            ScaleNodePoolResponse response = client.scaleNodePool(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

- Scale out nodes in the default scaling group of a node pool (yearly/monthly).

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class ScaleNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ScaleNodePoolRequest request = new ScaleNodePoolRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        ScaleNodePoolRequestBody body = new ScaleNodePoolRequestBody();
        ScaleUpExtendParam extendParamBillingConfigOverride = new ScaleUpExtendParam();
        extendParamBillingConfigOverride.withPeriodNum(1)
            .withPeriodType("month")
            .withIsAutoRenew(false)
            .withIsAutoPay(false);
        ScaleUpBillingConfigOverride billingConfigOverrideOptions = new
        ScaleUpBillingConfigOverride();
        billingConfigOverrideOptions.withBillingMode(1)
            .withExtendParam(extendParamBillingConfigOverride);
        ScaleNodePoolOptions optionsSpec = new ScaleNodePoolOptions();
        optionsSpec.withBillingConfigOverride(billingConfigOverrideOptions);
        List<String> listSpecScaleGroups = new ArrayList<>();
        listSpecScaleGroups.add("default");
        ScaleNodePoolSpec specbody = new ScaleNodePoolSpec();
        specbody.withDesiredNodeCount(1)
            .withScaleGroups(listSpecScaleGroups)
            .withOptions(optionsSpec);
        body.withSpec(specbody);
        body.withApiVersion("v3");
        body.withKind("NodePool");
        request.withBody(body);
        try {
            ScaleNodePoolResponse response = client.scaleNodePool(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

```
    } catch (ServiceResponseException e) {  
        e.printStackTrace();  
        System.out.println(e.getHttpStatusCode());  
        System.out.println(e.getRequestId());  
        System.out.println(e.getErrorCode());  
        System.out.println(e.getErrorMsg());  
    }  
}  
}
```

Python

- Scale out nodes in the default scaling group of a node pool (pay-per-use).

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
    # environment variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before  
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
    # environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = CceClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(CceRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ScaleNodePoolRequest()  
        request.cluster_id = "{cluster_id}"  
        request.nodepool_id = "{nodepool_id}"  
        listScaleGroupsSpec = [  
            "default"  
        ]  
        specbody = ScaleNodePoolSpec(  
            desired_node_count=1,  
            scale_groups=listScaleGroupsSpec  
        )  
        request.body = ScaleNodePoolRequestBody(  
            spec=specbody,  
            api_version="v3",  
            kind="NodePool"  
        )  
        response = client.scale_node_pool(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

- Scale out nodes in the default scaling group of a node pool (yearly/monthly).

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
```



```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ScaleNodePoolRequest()
        request.cluster_id = "{cluster_id}"
        request.nodepool_id = "{nodepool_id}"
        extendParamBillingConfigOverride = ScaleUpExtendParam(
            period_num=1,
            period_type="month",
            is_auto_renew=False,
            is_auto_pay=False
        )
        billingConfigOverrideOptions = ScaleUpBillingConfigOverride(
            billing_mode=1,
            extend_param=extendParamBillingConfigOverride
        )
        optionsSpec = ScaleNodePoolOptions(
            billing_config_override=billingConfigOverrideOptions
        )
        listScaleGroupsSpec = [
            "default"
        ]
        specbody = ScaleNodePoolSpec(
            desired_node_count=1,
            scale_groups=listScaleGroupsSpec,
            options=optionsSpec
        )
        request.body = ScaleNodePoolRequestBody(
            spec=specbody,
            api_version="v3",
            kind="NodePool"
        )
        response = client.scale_node_pool(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

- Scale out nodes in the default scaling group of a node pool (pay-per-use).

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
```

```
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
  )

  func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
      WithAk(ak).
      WithSk(sk).
      WithProjectId(projectId).
      Build()

    client := cce.NewCceClient(
      cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

    request := &model.ScaleNodePoolRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    var listScaleGroupsSpec = []string{
      "default",
    }
    specbody := &model.ScaleNodePoolSpec{
      DesiredNodeCount: int32(1),
      ScaleGroups: listScaleGroupsSpec,
    }
    request.Body = &model.ScaleNodePoolRequestBody{
      Spec: specbody,
      ApiVersion: "v3",
      Kind: "NodePool",
    }
    response, err := client.ScaleNodePool(request)
    if err == nil {
      fmt.Printf("%+v\n", response)
    } else {
      fmt.Println(err)
    }
  }
}
```

- Scale out nodes in the default scaling group of a node pool (yearly/monthly).

```
package main

import (
  "fmt"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
  cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
  "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
  region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
  // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
  security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
  environment variables and decrypted during use to ensure security.
  // In this example, AK and SK are stored in environment variables for authentication. Before
  running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
  environment
  ak := os.Getenv("CLOUD_SDK_AK")
  sk := os.Getenv("CLOUD_SDK_SK")
  projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ScaleNodePoolRequest{
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    isAutoRenewExtendParam:= false
    isAutoPayExtendParam:= false
    extendParamBillingConfigOverride := &model.ScaleUpExtendParam{
        PeriodNum: int32(1),
        PeriodType: "month",
        IsAutoRenew: &isAutoRenewExtendParam,
        IsAutoPay: &isAutoPayExtendParam,
    }
    billingConfigOverrideOptions := &model.ScaleUpBillingConfigOverride{
        BillingMode: int32(1),
        ExtendParam: extendParamBillingConfigOverride,
    }
    optionsSpec := &model.ScaleNodePoolOptions{
        BillingConfigOverride: billingConfigOverrideOptions,
    }
    var listScaleGroupsSpec = []string{
        "default",
    }
    specbody := &model.ScaleNodePoolSpec{
        DesiredNodeCount: int32(1),
        ScaleGroups: listScaleGroupsSpec,
        Options: optionsSpec,
    }
    request.Body = &model.ScaleNodePoolRequestBody{
        Spec: specbody,
        ApiVersion: "v3",
        Kind: "NodePool",
    }
    response, err := client.ScaleNodePool(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	The node pool scaling is accepted. Nodes in the node pool will be added or deleted based on the expected number of nodes in the node pool after scaling.

Error Codes

See [Error Codes](#).

4.4.7 Synchronizing Node Configurations in a Node Pool

Function

This API is used to synchronize the configurations of existing nodes in a node pool.

 NOTE

The URL for cluster management is in the format of **https://Endpoint/uri**, where **uri** specifies the resource path for API access.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/operation/upgrade

Table 4-841 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-842 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-843 Request body parameters

Parameter	Mandatory	Type	Description
spec	Yes	UpgradeNodePoolSpec object	Definition Parameters in the request for synchronizing nodes in a node pool Constraints N/A

Table 4-844 UpgradeNodePoolSpec

Parameter	Mandatory	Type	Description
maxUnavailable	Yes	Integer	Definition The maximum number of nodes to be synchronized at a time. It also specifies the maximum number of unavailable nodes during node upgrades. When a node is reset for synchronization, the node is unavailable. Set this parameter to a proper value to prevent pod scheduling failures caused by a large number of unavailable nodes in the cluster. Constraints N/A Range 1 to 20 Default Value N/A

Parameter	Mandatory	Type	Description
nodeIds	No	Array of strings	Definition List of node IDs selected in the node pool to be synchronized. For details about how to obtain the IDs, see How to Obtain Parameters in the API URI. Constraints N/A Range N/A Default Value N/A
nodePoolID	Yes	String	Definition Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI. Constraints N/A Range N/A Default Value N/A
nodeTemplate	No	UpgradeNodePoolSpecNodeTemplate object	Definition Template parameters for synchronizing a node pool Constraints N/A Range N/A Default Value N/A

Table 4-845 UpgradeNodePoolSpecNodeTemplate

Parameter	Mandatory	Type	Description
lifeCycle	Yes	NodeLifecycleConfig object	Definition Customized node lifecycle settings Constraints N/A
login	Yes	Login object	Details: Node login mode. Either a key pair or password must be used for login. Constraints: None

Table 4-846 NodeLifecycleConfig

Parameter	Mandatory	Type	Description
preInstall	No	String	Definition Pre-installation script. The characters of both the pre-installation and post-installation scripts are centrally calculated. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Constraints The length cannot exceed 10,240 bytes. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
postInstall	No	String	Definition Pre-installation script. The characters of both the pre-installation and post-installation scripts are centrally calculated. The input value must be encoded using Base64. The method is as follows: <pre>echo -n "<content-to-be-encoded>" base64</pre> Constraints The length cannot exceed 10,240 bytes. Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
waitPostInstallFinish	No	Boolean	Definition Whether pods can be scheduled onto a node before the post-installation script has completed execution during operations such as node reset, management, and batch reset. If the node belongs to a node pool, the node pool settings apply. If this parameter is not specified or is set to false, pods can be scheduled to an available node after the Kubernetes node is ready. If this parameter is set to true, pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Constraints N/A Range • false: Pods can be scheduled to an available node after the Kubernetes node is ready. • true: Pods can be scheduled to an available node only after the Kubernetes node is ready and the post-installation script has been executed. Default Value false

Table 4-847 Login

Parameter	Mandatory	Type	Description
sshKey	No	String	Details: Name of the key pair used for login Constraints: None Options: N/A Default value: N/A
userPassword	No	UserPassword object	Details: Password used for node login Constraints: None
removeUserPassword	No	Boolean	Definition Whether to remove the password login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no user password can be specified. Range N/A Default Value false

Parameter	Mandatory	Type	Description
removeSSHKey	No	Boolean	Definition Whether to remove the key pair login setting of a node pool during the node pool update Constraints This parameter is supported only during node pool updates. If it is set to true , no SSH key can be specified. Range N/A Default Value false

Table 4-848 UserPassword

Parameter	Mandatory	Type	Description
username	No	String	Details: Login account. The default value is root . Constraints: None Options: N/A Default value: root

Parameter	Mandatory	Type	Description
password	Yes	String	Details: Login password. If a username and password are used when a node is created, this field is shielded in the response body. Constraints: The password field must be salted during node creation. For details, see Adding a Salt in the password Field When Creating a Node . Options: A password must: • Contain 8 to 26 characters. • Contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^_-=+[{ }],./? % ^ _ = + [{ }] , . / ? • Not contain the username or the username spelled backwards. Default value: N/A

Response Parameters

Status code: 200

The node pool synchronization has been accepted.

None

Example Requests

- Synchronize the node configurations in an on-premises node pool.

```
{
  "spec": {
    "maxUnavailable": 1,
    "nodeIDs": [ "aaaaaaaa-aaaa-aaaa-aaaaaaaaaaaa", "bbbbbbbb-bbbb-bbbb-bbbb-
bbbbbbbbbb" ],
    "nodePoolID": "ccccccc-cccc-cccc-cccc-cccccccccccc"
  }
}
```

- Synchronize the node configurations in the default node pool.

```
{
  "spec": {
    "maxUnavailable": 1,
```

```
"nodeIds" : [ "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa", "bbbbbbbb-bbbb-bbbb-bbbb-  
bbbbbbbbbb" ],  
"nodePoolID" : "DefaultPool",  
"nodeTemplate" : {  
  "lifeCycle" : {  
    "preInstall" : "ZWNobyAx",  
    "postInstall" : "ZWNobyAx"  
  },  
  "login" : {  
    "userPassword" : {  
      "password" : "xxxxx",  
      "username" : "root"  
    }  
  }  
}
```

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

- Synchronize the node configurations in an on-premises node pool.

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
import java.util.List;  
import java.util.ArrayList;  
  
public class UpgradeNodePoolSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before  
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local  
        // environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
        String projectId = "{project_id}";  
  
        ICredential auth = new BasicCredentials()  
            .withProjectId(projectId)  
            .withAk(ak)  
            .withSk(sk);  
  
        CceClient client = CceClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpgradeNodePoolRequest request = new UpgradeNodePoolRequest();
```

```
request.withClusterId("{cluster_id}");
request.withNodepoolId("{nodepool_id}");
UpgradeNodePool body = new UpgradeNodePool();
List<String> listSpecNodeIds = new ArrayList<>();
listSpecNodeIds.add("aaaaaaaa-aaaa-aaaa-aaaaaaaaaaaa");
listSpecNodeIds.add("bbbbbbb-bbbb-bbbb-bbbbbbbbbbb");
NodePoolUpgradeSpec specbody = new NodePoolUpgradeSpec();
specbody.withNodePoolID("ccccccc-cccc-cccc-cccccccccccc")
    .withNodeIds(listSpecNodeIds)
    .withMaxUnavailable(1);
body.withSpec(specbody);
request.withBody(body);
try {
    UpgradeNodePoolResponse response = client.upgradeNodePool(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrMsg());
}
}
```

- Synchronize the node configurations in the default node pool.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpgradeNodePoolSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before
        // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
        // environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        UpgradeNodePoolRequest request = new UpgradeNodePoolRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        UpgradeNodePool body = new UpgradeNodePool();
```



```
NodeTemplateLifeCycle lifeCycleNodeTemplate = new NodeTemplateLifeCycle();
lifeCycleNodeTemplate.withPreInstall("ZWNobyAx")
    .withPostInstall("ZWNobyAx");
NodeTemplateLoginUserPassword userPasswordLogin = new NodeTemplateLoginUserPassword();
userPasswordLogin.withUsername("root")
    .withPassword("xxxxx");
NodeTemplateLogin loginNodeTemplate = new NodeTemplateLogin();
loginNodeTemplate.withUserPassword(userPasswordLogin);
NodeTemplate nodeTemplateSpec = new NodeTemplate();
nodeTemplateSpec.withLogin(loginNodeTemplate)
    .withLifeCycle(lifeCycleNodeTemplate);
List<String> listSpecNodeIds = new ArrayList<>();
listSpecNodeIds.add("aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa");
listSpecNodeIds.add("bbbbbbb-bbbb-bbbb-bbbb-bbbbbbbbb");
NodePoolUpgradeSpec specbody = new NodePoolUpgradeSpec();
specbody.withNodePoolID("DefaultPool")
    .withNodeIds(listSpecNodeIds)
    .withNodeTemplate(nodeTemplateSpec)
    .withMaxUnavailable(1);
body.withSpec(specbody);
request.withBody(body);
try {
    UpgradeNodePoolResponse response = client.upgradeNodePool(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

- Synchronize the node configurations in an on-premises node pool.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpgradeNodePoolRequest()
```

```
request.cluster_id = "{cluster_id}"
request.nodepool_id = "{nodepool_id}"
listNodeIdsSpec = [
    "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa",
    "bbbbbb-bbbb-bbbb-bbbb-bbbbbbbbbbb"
]
specbody = NodePoolUpgradeSpec(
    node_pool_id="ccccccc-cccc-cccc-ccccccccccc",
    node_ids=listNodeIdsSpec,
    max_unavailable=1
)
request.body = UpgradeNodePool(
    spec=specbody
)
response = client.upgrade_node_pool(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

- Synchronize the node configurations in the default node pool.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    # security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    # environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before
    # running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    # environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpgradeNodePoolRequest()
        request.cluster_id = "{cluster_id}"
        request.nodepool_id = "{nodepool_id}"
        lifeCycleNodeTemplate = NodeTemplateLifecycle(
            pre_install="ZWNobyAx",
            post_install="ZWNobyAx"
        )
        userPasswordLogin = NodeTemplateLoginUserPassword(
            username="root",
            password="xxxxx"
        )
        loginNodeTemplate = NodeTemplateLogin(
            user_password=userPasswordLogin
        )
        nodeTemplateSpec = NodeTemplate(
            login=loginNodeTemplate,
            life_cycle=lifeCycleNodeTemplate
        )
        listNodeIdsSpec = [
            "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa",
```

```
        "bbbbbbb-bbbb-bbbb-bbbb-bbbbbbbbbbb"
    ]
    specbody = NodePoolUpgradeSpec(
        node_pool_id="DefaultPool",
        node_ids=listNodeIdsSpec,
        node_template=nodeTemplateSpec,
        max_unavailable=1
    )
    request.body = UpgradeNodePool(
        spec=specbody
    )
    response = client.upgrade_node_pool(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

- Synchronize the node configurations in an on-premises node pool.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpgradeNodePoolRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    var listNodeIdsSpec = []string{
        "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa",
        "bbbbbbb-bbbb-bbbb-bbbb-bbbbbbbbbbb",
    }
    maxUnavailableSpec := int32(1)
    specbody := &model.NodePoolUpgradeSpec{
        NodePoolId: "ccccccc-cccc-cccc-cccc-cccccccccc",
        NodeIds: &listNodeIdsSpec,
        MaxUnavailable: &maxUnavailableSpec,
    }
    request.Body = &model.UpgradeNodePool{
```

```
    Spec: specbody,
  }
  response, err := client.UpgradeNodePool(request)
  if err == nil {
    fmt.Printf("%v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

- Synchronize the node configurations in the default node pool.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before
    // running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local
    // environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpgradeNodePoolRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    preInstallLifeCycle := "ZWNobyAx"
    postInstallLifeCycle := "ZWNobyAx"
    lifeCycleNodeTemplate := &model.NodeTemplateLifeCycle{
        PreInstall: &preInstallLifeCycle,
        PostInstall: &postInstallLifeCycle,
    }
    usernameUserPassword := "root"
    passwordUserPassword := "xxxxx"
    userPasswordLogin := &model.NodeTemplateLoginUserPassword{
        Username: &usernameUserPassword,
        Password: &passwordUserPassword,
    }
    loginNodeTemplate := &model.NodeTemplateLogin{
        UserPassword: userPasswordLogin,
    }
    nodeTemplateSpec := &model.NodeTemplate{
        Login: loginNodeTemplate,
        LifeCycle: lifeCycleNodeTemplate,
    }
    var listNodeIdsSpec = []string{
        "aaaaaaaa-aaaa-aaaa-aaaaaaaaaaaa",
        "bbbbbbb-bbbb-bbbb-bbbb-bbbbbbbbbb",
    }
}
```

```
maxUnavailableSpec:= int32(1)
specbody := &model.NodePoolUpgradeSpec{
    NodePoolID: "DefaultPool",
    NodeIDs: &listNodeIDsSpec,
    NodeTemplate: nodeTemplateSpec,
    MaxUnavailable: &maxUnavailableSpec,
}
request.Body = &model.UpgradeNodePool{
    Spec: specbody,
}
response, err := client.UpgradeNodePool(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The node pool synchronization has been accepted.

Error Codes

See [Error Codes](#).

4.5 Storage Management

4.5.1 Creating a PVC (to be discarded)

Function

This API is used to create a PersistentVolumeClaim (PVC) in a specified namespace. A PVC is a request for PersistentVolume (PV). The PV carries details of real cloud storage such as EVS disks, SFS file systems, and OBS buckets. This API is to be deprecated. Use the corresponding Kubernetes PVC API instead.

NOTE

The URL for storage management is in the format of **https://{clusterid}.Endpoint/uri**. In the URL, *{clusterid}* indicates the cluster ID, and *uri* indicates the resource path, that is, the path for API access. If *https://Endpoint/uri* is used, the **X-Cluster-ID** parameter in the request header must be specified.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v1/namespaces/{namespace}/cloudpersistentvolumeclaims

Table 4-849 Path Parameters

Parameter	Mandatory	Type	Description
namespace	Yes	String	Namespace where the PVC is located. Constraints: • Custom namespace. You must create a namespace in the cluster before using it. • The namespace provided by the system is namespace default. • Namespaces kube-system and kube-public cannot be used.

Request Parameters

Table 4-850 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A
X-Cluster-ID	No	String	Cluster ID. This field is mandatory when the URL format https://Endpoint/uri is used. For details about how to obtain the value, see How to Obtain Parameters in the API URI .

Table 4-851 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	API version. The value is fixed at v1 .
kind	Yes	String	API type. The value is fixed at PersistentVolumeClaim .
metadata	Yes	PersistentVolumeClaimMetadata object	Cluster object metadata, which is a collection of attributes.
spec	Yes	PersistentVolumeClaimSpec object	Detailed description about the cluster object. CCE creates or updates objects by defining or updating spec .
status	No	PersistentVolumeClaimStatus object	PVC status. The request to create a PVC does not need to carry this parameter.

Table 4-852 PersistentVolumeClaimMetadata

Parameter	Mandatory	Type	Description
name	Yes	String	PVC name. Enter 1 to 253 characters starting and ending with a letter or digit. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. PVC names must be unique in a namespace.
labels	No	Map<String,String>	PVC labels, in the format of key-value pairs. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed.

Table 4-853 PersistentVolumeClaimSpec

Parameter	Mandatory	Type	Description
volumeID	Yes	String	ID of an existing storage volume. • If an SFS, EVS, or SFS Turbo volume is used, set this parameter to the ID of the volume. • If an OBS bucket is used, set this parameter to the OBS bucket name.

Parameter	Mandatory	Type	Description
storageType	Yes	String	Cloud storage class. This parameter is used together with volumeID . That means that volumeID and storageType must be configured concurrently. • bs: EVS • nfs: SFS 1.0 • obs: OBS • efs: SFS Turbo
accessModes	Yes	Array of strings	Access mode of the volume. Only the first value in all selected options is valid. • ReadWriteOnce: The volume can be mounted as read-write by a single node. NOTE This function is supported only when the cluster version is v1.13.10 and the storage-driver version is 1.0.19. • ReadOnlyMany (default): The volume can be mounted as read-only by many nodes. • ReadWriteMany: The volume can be mounted as read-write by many nodes.
storageClassName	No	String	Storage class name of the PVC.
volumeName	No	String	Name of the PV bound to the PVC.
resources	No	ResourceRequirements object	Resource requests and limits.
volumeMode	No	String	PV type specified by the PVC.

Table 4-854 ResourceRequirements

Parameter	Mandatory	Type	Description
limits	No	Map<String,String>	Resource limits. This parameter is invalid during creation.

Parameter	Mandatory	Type	Description
requests	No	Map<String,String>	Resource requests. This parameter is invalid during creation.

Table 4-855 PersistentVolumeClaimStatus

Parameter	Mandatory	Type	Description
accessModes	No	Array of strings	Access mode of the PV.
capacity	No	String	Actual resources and capacity of the PV.
phase	No	String	Status of the PVC.

Response Parameters

Status code: 201

Table 4-856 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. The value is fixed at v1 .
kind	String	API type. The value is fixed at PersistentVolumeClaim .
metadata	PersistentVolumeClaimMetadata object	Cluster object metadata, which is a collection of attributes.
spec	PersistentVolumeClaimSpec object	Detailed description about the cluster object. CCE creates or updates objects by defining or updating spec .
status	PersistentVolumeClaimStatus object	PVC status. The request to create a PVC does not need to carry this parameter.

Table 4-857 PersistentVolumeClaimMetadata

Parameter	Type	Description
name	String	PVC name. Enter 1 to 253 characters starting and ending with a letter or digit. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. PVC names must be unique in a namespace.
labels	Map<String,String>	PVC labels, in the format of key-value pairs. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed.

Table 4-858 PersistentVolumeClaimSpec

Parameter	Type	Description
volumeID	String	ID of an existing storage volume. • If an SFS, EVS, or SFS Turbo volume is used, set this parameter to the ID of the volume. • If an OBS bucket is used, set this parameter to the OBS bucket name.
storageType	String	Cloud storage class. This parameter is used together with volumeID. That means that volumeID and storageType must be configured concurrently. • bs: EVS • nfs: SFS 1.0 • obs: OBS • efs: SFS Turbo

Parameter	Type	Description
accessModes	Array of strings	Access mode of the volume. Only the first value in all selected options is valid. • ReadWriteOnce: The volume can be mounted as read-write by a single node. NOTE This function is supported only when the cluster version is v1.13.10 and the storage-driver version is 1.0.19. • ReadOnlyMany (default): The volume can be mounted as read-only by many nodes. • ReadWriteMany: The volume can be mounted as read-write by many nodes.
storageClassName	String	Storage class name of the PVC.
volumeName	String	Name of the PV bound to the PVC.
resources	ResourceRequirements object	Resource requests and limits.
volumeMode	String	PV type specified by the PVC.

Table 4-859 ResourceRequirements

Parameter	Type	Description
limits	Map<String,String>	Resource limits. This parameter is invalid during creation.
requests	Map<String,String>	Resource requests. This parameter is invalid during creation.

Table 4-860 PersistentVolumeClaimStatus

Parameter	Type	Description
accessModes	Array of strings	Access mode of the PV.
capacity	String	Actual resources and capacity of the PV.
phase	String	Status of the PVC.

Example Requests

Specifying an EVS volume ID and creating a PVC

POST /api/v1/namespaces/default/cloudpersistentvolumeclaims

```
{
  "apiVersion": "v1",
  "kind": "PersistentVolumeClaim",
  "metadata": {
    "name": "csms-dev-create",
    "namespace": "default"
  },
  "spec": {
    "volumeID": "86b29e16-23db-11e7-9c83-fa163ec08232",
    "storageType": "bs",
    "accessModes": [ "ReadWriteMany" ]
  }
}
```

Example Responses

Status code: 201

The job for creating a PVC is successfully delivered.

```
{
  "kind": "PersistentVolumeClaim",
  "apiVersion": "v1",
  "metadata": {
    "name": "csms-dev-create ",
    "namespace": "default",
    "selfLink": "/api/v1/namespaces/default/persistentvolumeclaims/db-mysql-0",
    "uid": "86b29e16-23db-11e7-9c83-fa163ec08232",
    "resourceVersion": "1793115",
    "creationTimestamp": "2017-04-18T02:05:42Z"
  },
  "spec": {
    "volumeName": "csms-dev-create ",
    "accessModes": [ "ReadWriteMany" ],
    "resources": {
      "requests": {
        "storage": "1Gi"
      }
    }
  },
  "status": {
    "phase": "Pending",
    "accessModes": [ "ReadWriteMany" ]
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Specifying an EVS volume ID and creating a PVC

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
```

```
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateCloudPersistentVolumeClaimsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        CreateCloudPersistentVolumeClaimsRequest request = new
        CreateCloudPersistentVolumeClaimsRequest();
        request.withNamespace("{namespace}");
        PersistentVolumeClaim body = new PersistentVolumeClaim();
        List<PersistentVolumeClaimSpec.AccessModesEnum> listSpecAccessModes = new ArrayList<>();

        listSpecAccessModes.add(PersistentVolumeClaimSpec.AccessModesEnum.fromValue("ReadWriteMany"));
        PersistentVolumeClaimSpec specbody = new PersistentVolumeClaimSpec();
        specbody.withVolumeID("86b29e16-23db-11e7-9c83-fa163ec08232")
            .withStorageType("bs")
            .withAccessModes(listSpecAccessModes);
        PersistentVolumeClaimMetadata metadatabody = new PersistentVolumeClaimMetadata();
        metadatabody.withName("csms-dev-create");
        body.withSpec(specbody);
        body.withMetadata(metadatabody);
        body.withKind("PersistentVolumeClaim");
        body.withApiVersion("v1");
        request.withBody(body);
        try {
            CreateCloudPersistentVolumeClaimsResponse response =
            client.createCloudPersistentVolumeClaims(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Specifying an EVS volume ID and creating a PVC

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateCloudPersistentVolumeClaimsRequest()
        request.namespace = "{namespace}"
        listAccessModesSpec = [
            "ReadWriteMany"
        ]
        specbody = PersistentVolumeClaimSpec(
            volume_id="86b29e16-23db-11e7-9c83-fa163ec08232",
            storage_type="bs",
            access_modes=listAccessModesSpec
        )
        metadatabody = PersistentVolumeClaimMetadata(
            name="csms-dev-create"
        )
        request.body = PersistentVolumeClaim(
            spec=specbody,
            metadata=metadatabody,
            kind="PersistentVolumeClaim",
            api_version="v1"
        )
        response = client.create_cloud_persistent_volume_claims(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Specifying an EVS volume ID and creating a PVC

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
```

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateCloudPersistentVolumeClaimsRequest{}
request.Namespace = "{namespace}"
var listAccessModesSpec = []model.PersistentVolumeClaimSpecAccessModes{
    model.GetPersistentVolumeClaimSpecAccessModesEnum().READ_WRITE_MANY,
}
specbody := &model.PersistentVolumeClaimSpec{
    VolumeID: "86b29e16-23db-11e7-9c83-fa163ec08232",
    StorageType: "bs",
    AccessModes: listAccessModesSpec,
}
metadatabody := &model.PersistentVolumeClaimMetadata{
    Name: "csms-dev-create",
}
request.Body = &model.PersistentVolumeClaim{
    Spec: specbody,
    Metadata: metadatabody,
    Kind: "PersistentVolumeClaim",
    ApiVersion: "v1",
}
response, err := client.CreateCloudPersistentVolumeClaims(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	The job for creating a PVC is successfully delivered.

Error Codes

See [Error Codes](#).

4.5.2 Deleting a PVC (to be discarded)

Function

This API is used to delete a PVC from a specified namespace. Using this API, you can determine whether to retain the cloud storage volume associated with the PVC. This API has been deprecated. Use the corresponding Kubernetes PV API instead.

NOTE

The URL for storage management is in the format of **https://{clusterid}.Endpoint/uri**. In the URL, *{clusterid}* indicates the cluster ID, and *uri* indicates the resource path, that is, the path for API access. If *https://Endpoint/uri* is used, the **X-Cluster-ID** parameter in the request header must be specified.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /api/v1/namespaces/{namespace}/cloudpersistentvolumeclaims/{name}

Table 4-861 Path Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the PVC to be deleted.
namespace	Yes	String	Namespace where the PVC is located.

Table 4-862 Query Parameters

Parameter	Mandatory	Type	Description
deleteVolume	No	String	Whether to delete associated backend cloud storage volume after the PVC is deleted. The value can be true (delete) or false (not delete). The value defaults to false .

Parameter	Mandatory	Type	Description
storageType	No	String	Cloud storage type. This parameter is used together with deleteVolume . This means that deleteVolume and storageType must be configured simultaneously. • bs: EVS • nfs: SFS 1.0 • obs: OBS • efs: SFS Turbo

Request Parameters

Table 4-863 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A
X-Cluster-ID	No	String	Cluster ID. This field is mandatory when the URL format https://Endpoint/uri is used. For details about how to obtain the value, see How to Obtain Parameters in the API URI .

Response Parameters

Status code: 200

Table 4-864 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. The value is fixed at v1 .
kind	String	API type. The value is fixed at PersistentVolumeClaim .
metadata	PersistentVolumeClaimMetadata object	Cluster object metadata, which is a collection of attributes.
spec	PersistentVolumeClaimSpec object	Detailed description about the cluster object. CCE creates or updates objects by defining or updating spec .
status	PersistentVolumeClaimStatus object	PVC status. The request to create a PVC does not need to carry this parameter.

Table 4-865 PersistentVolumeClaimMetadata

Parameter	Type	Description
name	String	PVC name. Enter 1 to 253 characters starting and ending with a letter or digit. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. PVC names must be unique in a namespace.
labels	Map<String,String>	PVC labels, in the format of key-value pairs. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed.

Table 4-866 PersistentVolumeClaimSpec

Parameter	Type	Description
volumeID	String	ID of an existing storage volume. • If an SFS, EVS, or SFS Turbo volume is used, set this parameter to the ID of the volume. • If an OBS bucket is used, set this parameter to the OBS bucket name.
storageType	String	Cloud storage class. This parameter is used together with volumeID. That means that volumeID and storageType must be configured concurrently. • bs: EVS • nfs: SFS 1.0 • obs: OBS • efs: SFS Turbo

Parameter	Type	Description
accessModes	Array of strings	Access mode of the volume. Only the first value in all selected options is valid. • ReadWriteOnce: The volume can be mounted as read-write by a single node. NOTE This function is supported only when the cluster version is v1.13.10 and the storage-driver version is 1.0.19. • ReadOnlyMany (default): The volume can be mounted as read-only by many nodes. • ReadWriteMany: The volume can be mounted as read-write by many nodes.
storageClassName	String	Storage class name of the PVC.
volumeName	String	Name of the PV bound to the PVC.
resources	ResourceRequirements object	Resource requests and limits.
volumeMode	String	PV type specified by the PVC.

Table 4-867 ResourceRequirements

Parameter	Type	Description
limits	Map<String,String>	Resource limits. This parameter is invalid during creation.
requests	Map<String,String>	Resource requests. This parameter is invalid during creation.

Table 4-868 PersistentVolumeClaimStatus

Parameter	Type	Description
accessModes	Array of strings	Access mode of the PV.
capacity	String	Actual resources and capacity of the PV.
phase	String	Status of the PVC.

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class DeleteCloudPersistentVolumeClaimsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteCloudPersistentVolumeClaimsRequest request = new
        DeleteCloudPersistentVolumeClaimsRequest();
        request.setName("{name}");
        request.withNamespace("{namespace}");
        try {
            DeleteCloudPersistentVolumeClaimsResponse response =
            client.deleteCloudPersistentVolumeClaims(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteCloudPersistentVolumeClaimsRequest()
        request.name = "{name}"
        request.namespace = "{namespace}"
        response = client.delete_cloud_persistent_volume_claims(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.DeleteCloudPersistentVolumeClaimsRequest{}
request.Name = "{name}"
request.Namespace = "{namespace}"
response, err := client.DeleteCloudPersistentVolumeClaims(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The job for deleting a PVC is successfully delivered.

Error Codes

See [Error Codes](#).

4.6 Add-on Management

4.6.1 Installing an Add-on Instance

Function

This API is used to install an add-on instance by using an add-on template.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/addons

Request Parameters

Table 4-869 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-870 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	API type. The value is fixed at Addon and cannot be changed. Any user-defined value is invalid.
apiVersion	Yes	String	API version. The value is fixed at v3 and cannot be changed. Any user-defined value is invalid.
metadata	Yes	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.
spec	Yes	InstanceRequestSpec object	Detailed description about add-on installation or upgrade.

Table 4-871 AddonMetadata

Parameter	Mandatory	Type	Description
uid	No	String	Unique ID
name	No	String	Add-on name
alias	No	String	Add-on alias
labels	No	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	No	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	No	String	Update time.
creationTimestamp	No	String	Creation time.

Table 4-872 InstanceRequestSpec

Parameter	Mandatory	Type	Description
version	No	String	Version of the add-on to install or upgrade, for example, 1.0.0 . - Installation: This parameter is optional. If not specified, the latest version supported by the cluster is used. - Upgrade: This parameter is mandatory. The version number must be specified.
clusterID	Yes	String	Cluster ID.
values	Yes	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). During the add-on upgrade, you need to specify all the installation parameters. If the parameters are not specified, the default values in the add-on template are used. The current add-on installation parameters can be obtained through the API for querying add-on instances.
addonTemplateName	Yes	String	Name of the add-on template to be installed, for example, coredns .

Response Parameters

Status code: 201

Table 4-873 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.

Parameter	Type	Description
spec	InstanceSpec object	Detailed description about the add-on instance.
status	AddonInstanceStatus object	Add-on instance status.

Table 4-874 AddonMetadata

Parameter	Type	Description
uid	String	Unique ID
name	String	Add-on name
alias	String	Add-on alias
labels	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	String	Update time.
creationTimestamp	String	Creation time.

Table 4-875 InstanceSpec

Parameter	Type	Description
clusterID	String	Cluster ID.
version	String	Add-on template version, for example, 1.0.0 .
addonTemplateName	String	Add-on template name, for example, coredns .
addonTemplateType	String	Add-on template type.

Parameter	Type	Description
addonTemplateLogo	String	URL for obtaining the add-on template logo.
addonTemplateLabels	Array of strings	Labels of the add-on template.
description	String	Add-on template description.
values	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). Set the parameters accordingly.

Table 4-876 AddonInstanceStatus

Parameter	Type	Description
status	String	Add-on instance status. Options: • running: All of the add-on instances are running. This specifies that the add-on runs properly. • abnormal: The add-on instances are abnormal and the add-on cannot be used. You can click the add-on name to view exceptions. • installing: The add-on is being installed. • installFailed: Installing the add-on failed. In this case, uninstall the add-on and then reinstall it. • upgrading: The add-on is being upgraded. • upgradeFailed: Upgrading the add-on failed. In this case, upgrade the add-on again or uninstall the add-on and reinstall it. • deleting: The add-on is being deleted. • deleteFailed: Deleting the add-on failed. In this case, uninstall the add-on again. • deleteSuccess: Deleting the add-on succeeded. • available: Only some instances of the add-on are running. This specifies that some functions of the add-on are available. • rollbacking: The add-on is being rolled back. • rollbackFailed: Rolling back the add-on failed. In this case, roll back the add-on again or uninstall the add-on and reinstall it. • unknown: The add-on chart instance does not exist.
Reason	String	Cause of the add-on installation failure.
message	String	Installation error details.
targetVersions	Array of strings	Cluster versions that support the add-on upgrade.

Parameter	Type	Description
currentVersion	Versions object	Information about the current add-on version.
isRollbackable	Boolean	Whether the add-on version can be rolled back to the source version.
previousVersion	String	The add-on version before upgrade or rollback

Table 4-877 Versions

Parameter	Type	Description
version	String	Add-on version.
input	Object	Add-on installation parameters.
stable	Boolean	Whether the add-on version is a stable release.
translate	Object	Translation information used by the GUI.
supportVersions	Array of SupportVersions objects	Cluster versions that support the add-on.
creationTimestamp	String	Creation time.
updateTimestamp	String	Update time.

Table 4-878 SupportVersions

Parameter	Type	Description
clusterType	String	Cluster type that supports the add-on.
clusterVersion	Array of strings	Cluster versions that support the add-on. The value is a regular expression.

Parameter	Type	Description
category	Array of strings	The cluster types to which the settings are applied Options: • CCE: CCE standard clusters • Turbo: CCE Turbo clusters • Autopilot: CCE Autopilot clusters Default value: If this parameter is left blank, CCE standard and Turbo clusters are used by default.

Example Requests

Install the coredns add-on of version 1.17.15. The add-on specification is 2500 QPS, and the number of add-on instances is 2.

```
{
  "kind" : "Addon",
  "apiVersion" : "v3",
  "metadata" : {
    "annotations" : {
      "addon.install/type" : "install"
    }
  },
  "spec" : {
    "clusterID" : "1b2ec02d-a3b2-11ec-b0d0-0255ac100099",
    "version" : "1.17.15",
    "addonTemplateName" : "coredns",
    "values" : {
      "basic" : {
        "cluster_ip" : "10.247.3.10",
        "image_version" : "1.17.15",
        "platform" : "linux-amd64",
        "swr_addr" : "<Replace_SWR_address>",
        "swr_user" : "hwofficial",
        "rbac_enabled" : true
      },
      "flavor" : {
        "name" : 2500,
        "replicas" : 2,
        "resources" : [ {
          "limitsCpu" : "500m",
          "limitsMem" : "512Mi",
          "name" : "coredns",
          "requestsCpu" : "500m",
          "requestsMem" : "512Mi"
        } ]
      },
      "custom" : {
        "stub_domains" : { },
        "upstream_nameservers" : [ ],
        "cluster_id" : "1b2ec02d-a3b2-11ec-b0d0-0255ac100099",
        "tenant_id" : "0504201b6c80256b2f08c0099f0c8fe4"
      }
    }
  }
}
```


Example Responses

Status code: 201

OK

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "uid": "b748aaea-a984-11ec-987b-0255ac1000bc",
    "name": "coredns",
    "alias": "coredns",
    "creationTimestamp": "2022-03-22T02:06:41Z",
    "updateTimestamp": "2022-03-22T02:06:41Z"
  },
  "spec": {
    "clusterID": "1b2ec02d-a3b2-11ec-b0d0-0255ac100099",
    "version": "1.17.15",
    "addonTemplateName": "coredns",
    "addonTemplateType": "helm",
    "addonTemplateLogo": "",
    "addonTemplateLabels": [ "ServiceDiscovery" ],
    "description": "CoreDNS is a DNS server that chains plugins and provides Kubernetes DNS Services",
    "values": {
      "basic": {
        "cluster_ip": "10.247.3.10",
        "image_version": "1.17.15",
        "platform": "linux-amd64",
        "rbac_enabled": true,
        "swr_addr": "",
        "swr_user": "hwofficial"
      },
      "custom": {
        "cluster_id": "1b2ec02d-a3b2-11ec-b0d0-0255ac100099",
        "stub_domains": { },
        "tenant_id": "0504201b6c80256b2f08c0099f0c8fe4",
        "upstream_nameservers": [ ]
      },
      "flavor": {
        "name": 2500,
        "replicas": 2,
        "resources": [ {
          "limitsCpu": "500m",
          "limitsMem": "512Mi",
          "name": "coredns",
          "requestsCpu": "500m",
          "requestsMem": "512Mi"
        } ]
      }
    }
  },
  "status": {
    "status": "installing",
    "Reason": "",
    "message": "",
    "targetVersions": null,
    "currentVersion": {
      "version": "1.17.15",
      "input": {
        "basic": {
          "cluster_ip": "10.247.3.10",
          "image_version": "1.17.15",
          "platform": "linux-amd64",
          "swr_addr": "",
          "swr_user": "hwofficial"
        },
        "parameters": {
          "custom": {
```

```
"stub_domains" : "",
"upstream_nameservers" : ""
},
"flavor1" : {
  "name" : 2500,
  "replicas" : 2,
  "resources" : [ {
    "limitsCpu" : "500m",
    "limitsMem" : "512Mi",
    "name" : "coredns",
    "requestsCpu" : "500m",
    "requestsMem" : "512Mi"
  } ]
},
"flavor2" : {
  "name" : 5000,
  "replicas" : 2,
  "resources" : [ {
    "limitsCpu" : "1000m",
    "limitsMem" : "1024Mi",
    "name" : "coredns",
    "requestsCpu" : "1000m",
    "requestsMem" : "1024Mi"
  } ]
},
"flavor3" : {
  "name" : 10000,
  "replicas" : 2,
  "resources" : [ {
    "limitsCpu" : "2000m",
    "limitsMem" : "2048Mi",
    "name" : "coredns",
    "requestsCpu" : "2000m",
    "requestsMem" : "2048Mi"
  } ]
},
"flavor4" : {
  "name" : 20000,
  "replicas" : 4,
  "resources" : [ {
    "limitsCpu" : "2000m",
    "limitsMem" : "2048Mi",
    "name" : "coredns",
    "requestsCpu" : "2000m",
    "requestsMem" : "2048Mi"
  } ]
}
},
"stable" : true,
"translate" : {
  "en_US" : {
    "addon" : {
      "changeLog" : "Supported CCE clusters of v1.21.",
      "description" : "CoreDNS is a DNS server that chains plugins and provides Kubernetes DNS Services"
    },
    "description" : {
      "Parameters.custom.stub_domains" : "The target nameserver may itself be a Kubernetes service. For instance, you can run your own copy of dnsmasq to export custom DNS names into the ClusterDNS namespace, a JSON map using a DNS suffix key (e.g. \"acme.local\") and a value consisting of a JSON array of DNS IPs.",
      "Parameters.custom.upstream_nameservers" : "If specified, then the values specified replace the nameservers taken by default from the node's /etc/resolv.conf. Limits:a maximum of three upstream nameservers can be specified, A JSON array of DNS IPs.",
      "Parameters.flavor1.description" : "Concurrent domain name resolution ability - External domain name: 2500 qps, Internal domain name: 10000 qps",
      "Parameters.flavor1.name" : 2500,
      "Parameters.flavor2.description" : "Concurrent domain name resolution ability - External domain name: 5000 qps, Internal domain name: 20000 qps",
```

```
"Parameters.flavor2.name" : 5000,
"Parameters.flavor3.description" : "Concurrent domain name resolution ability - External domain
name: 10000 qps, Internal domain name: 40000 qps",
"Parameters.flavor3.name" : 10000,
"Parameters.flavor4.description" : "Concurrent domain name resolution ability - External domain
name: 20000 qps, Internal domain name: 80000 qps",
"Parameters.flavor4.name" : 20000
},
"key" : {
  "Parameters.custom.stub_domains" : "stub domain",
  "Parameters.custom.upstream_nameservers" : "upstream nameservers"
}
},
"fr_FR" : {
  "addon" : {
    "changeLog" : "Prise en charge du cluster 1.21.",
    "description" : "Un serveur DNS qui enchaîne les plug-ins et fournit des services DNS Kubernetes."
  },
  "description" : {
    "Parameters.custom.stub_domains" : "Le serveur de noms cible peut lui-même être un service
Kubernetes. Par exemple, vous pouvez exécuter votre propre copie de dnsmasq pour exporter des noms
DNS personnalisés dans l'espace de noms ClusterDNS, une carte JSON à l'aide d'une clé de suffixe DNS (par
exemple, «acme.local») et une valeur constituée d'un tableau JSON d'adresses IP DNS.",
    "Parameters.custom.upstream_nameservers" : "Si spécifié, les valeurs spécifiées remplacent les
serveurs de noms pris par défaut dans le fichier /etc/resolv.conf du nœud. Limites: un maximum de trois
serveurs de noms en amont peuvent être spécifiés, un tableau JSON d'adresses IP DNS.",
    "Parameters.flavor1.description" : "Capacité de résolution de nom de domaine simultanée - Nom de
domaine externe: 2500 qps, Nom de domaine interne: 10000 qp",
    "Parameters.flavor1.name" : 2500,
    "Parameters.flavor2.description" : "Capacité de résolution de nom de domaine simultanée - Nom de
domaine externe: 5000 qps, Nom de domaine interne: 20000 qp",
    "Parameters.flavor2.name" : 5000,
    "Parameters.flavor3.description" : "Capacité de résolution de nom de domaine simultanée - Nom de
domaine externe: 10000 qps, Nom de domaine interne: 40000 qp",
    "Parameters.flavor3.name" : 10000,
    "Parameters.flavor4.description" : "Capacité de résolution de nom de domaine simultanée - Nom de
domaine externe: 20000 qps, Nom de domaine interne: 80000 qp",
    "Parameters.flavor4.name" : 20000
  },
  "key" : {
    "Parameters.custom.stub_domains" : "domaine stub",
    "Parameters.custom.upstream_nameservers" : "serveurs de noms en amont"
  }
},
"zh_CN" : {
  "addon" : {
    "changeLog" : "",
    "description" : ""
  },
  "description" : {
    "Parameters.custom.stub_domains" : "",
    "Parameters.custom.upstream_nameservers" : "",
    "Parameters.flavor1.description" : "",
    "Parameters.flavor1.name" : 2500,
    "Parameters.flavor2.description" : "",
    "Parameters.flavor2.name" : 5000,
    "Parameters.flavor3.description" : "",
    "Parameters.flavor3.name" : 10000,
    "Parameters.flavor4.description" : "",
    "Parameters.flavor4.name" : 20000
  },
  "key" : {
    "Parameters.custom.stub_domains" : "",
    "Parameters.custom.upstream_nameservers" : ""
  }
}
},
"supportVersions" : null,
"creationTimestamp" : "2021-12-14T13:43:15Z",
```

```
"updateTimestamp" : "2022-01-11T14:32:10Z"  
}  
}  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Install the coredns add-on of version 1.17.15. The add-on specification is 2500 QPS, and the number of add-on instances is 2.

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
import java.util.Map;  
import java.util.HashMap;  
  
public class CreateAddonInstanceSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        CceClient client = CceClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))  
            .build();  
  
        CreateAddonInstanceRequest request = new CreateAddonInstanceRequest();  
        InstanceRequest body = new InstanceRequest();  
        Map<String, Object> listSpecValues = new HashMap<>();  
        listSpecValues.put("basic", "{ \"rbac_enabled\": true, \"swr_user\": \"hwofficial\", \"image_version  
\": \"1.17.15\", \"cluster_ip\": \"10.247.3.10\", \"platform\": \"linux-amd64\", \"swr_addr  
\": \"<Replace_SWR_address>\" }");  
        listSpecValues.put("flavor", "{ \"replicas\": 2, \"name\": \"2500\", \"resources\": { \"limitsCpu\": \"500m  
\", \"name\": \"coredns\", \"limitsMem\": \"512Mi\", \"requestsMem\": \"512Mi\", \"requestsCpu\": \"500m\" } }");  
        listSpecValues.put("custom", "{ \"tenant_id\": \"0504201b6c80256b2f08c0099f0c8fe4\", \"cluster_id  
\": \"1b2ec02d-a3b2-11ec-b0d0-0255ac100099\", \"stub_domains\": {}, \"upstream_nameservers\": [] }");  
        InstanceRequestSpec specbody = new InstanceRequestSpec();  
        specbody.withVersion("1.17.15")  
            .withClusterID("1b2ec02d-a3b2-11ec-b0d0-0255ac100099")  
            .withValues(listSpecValues)  
            .withAddonTemplateName("coredns");  
        Map<String, String> listMetadataAnnotations = new HashMap<>();  
        listMetadataAnnotations.put("addon.install/type", "install");  
        AddonMetadata metadatabody = new AddonMetadata();  
        metadatabody.withAnnotations(listMetadataAnnotations);  
        body.withSpec(specbody);  
    }  
}
```

```
body.withMetadata(metadataBody);
body.withApiVersion("v3");
body.withKind("Addon");
request.withBody(body);
try {
    CreateAddonInstanceResponse response = client.createAddonInstance(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Install the coredns add-on of version 1.17.15. The add-on specification is 2500 QPS, and the number of add-on instances is 2.

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateAddonInstanceRequest()
        listValuesSpec = {
            "basic": {"rbac_enabled": true, "swr_user": "hwofficial", "image_version": "1.17.15", "cluster_ip": "10.247.3.10", "platform": "linux-amd64", "swr_addr": "<Replace_SWR_address>"},
            "flavor": {"replicas": 2, "name": "2500", "resources": [{"limitsCpu": "500m", "name": "coredns"}, {"limitsMem": "512Mi", "requestsMem": "512Mi", "requestsCpu": "500m"}]},
            "custom": {"tenant_id": "0504201b6c80256b2f08c0099f0c8fe4", "cluster_id": "1b2ec02d-a3b2-11ec-b0d0-0255ac100099", "stub_domains": {}, "upstream_nameservers": []}
        }
        specbody = InstanceRequestSpec(
            version="1.17.15",
            cluster_id="1b2ec02d-a3b2-11ec-b0d0-0255ac100099",
            values=listValuesSpec,
            addon_template_name="coredns"
        )
        listAnnotationsMetadata = {
            "addon.install/type": "install"
        }
    }
```

```
metadatabody = AddonMetadata(  
    annotations=listAnnotationsMetadata  
)  
request.body = InstanceRequest(  
    spec=specbody,  
    metadata=metadatabody,  
    api_version="v3",  
    kind="Addon"  
)  
response = client.create_addon_instance(request)  
print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

Install the coredns add-on of version 1.17.15. The add-on specification is 2500 QPS, and the number of add-on instances is 2.

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := cce.NewCceClient(  
        cce.CceClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.CreateAddonInstanceRequest{  
        var listValuesSpec = map[string]interface{}{  
            "basic": "{ \"rbac_enabled\":true,\"swr_user\":\"hwofficial\",\"image_version\":\"1.17.15\",\"cluster_ip  
\":\"10.247.3.10\",\"platform\":\"linux-amd64\",\"swr_addr\":\"<Replace_SWR_address>\"",  
            "flavor": "{ \"replicas\":2,\"name\":\"2500\",\"resources\":{\"limitsCpu\":\"500m\",\"name\":\"coredns  
\",\"limitsMem\":\"512Mi\",\"requestsMem\":\"512Mi\",\"requestsCpu\":\"500m\"}}",  
            "custom": "{ \"tenant_id\":\"0504201b6c80256b2f08c0099f0c8fe4\",\"cluster_id\":\"1b2ec02d-  
a3b2-11ec-b0d0-0255ac100099\",\"stub_domains\":{\"},\"upstream_nameservers\":{\"}}",  
        }  
        versionSpec:= "1.17.15"  
        specbody := &model.InstanceRequestSpec{  
            Version: &versionSpec,  
            ClusterID: "1b2ec02d-a3b2-11ec-b0d0-0255ac100099",  
            Values: listValuesSpec,  
            AddonTemplateName: "coredns",  
        }  
        var listAnnotationsMetadata = map[string]string{
```

```
"addon.install/type": "install",
}
metadatabody := &model.AddonMetadata{
    Annotations: listAnnotationsMetadata,
}
request.Body = &model.InstanceRequest{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Addon",
}
response, err := client.CreateAddonInstance(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	OK

Error Codes

See [Error Codes](#).

4.6.2 Listing Add-on Templates

Function

This API is used to query add-on templates.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/addontemplates

Table 4-879 Query Parameters

Parameter	Mandatory	Type	Description
addon_template_name	No	String	Add-on name or alias for search. If this parameter is not specified, all the add-ons will be searched.

Request Parameters

Table 4-880 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-881 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
items	Array of AddonTemplate objects	List of add-on templates.

Table 4-882 AddonTemplate

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.

Parameter	Type	Description
spec	Templatespec object	Detailed description about the add-on template.

Table 4-883 AddonMetadata

Parameter	Type	Description
uid	String	Unique ID
name	String	Add-on name
alias	String	Add-on alias
labels	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	String	Update time.
creationTimestamp	String	Creation time.

Table 4-884 Templatespec

Parameter	Type	Description
type	String	Template type. The value is helm or static .
require	Boolean	Whether the add-on is mandatory.
labels	Array of strings	Group to which the template belongs.
logoURL	String	URL of the logo image.
readmeURL	String	Add-on details and usage description.
description	String	Template description.
versions	Array of Versions objects	Template version details.

Table 4-885 Versions

Parameter	Type	Description
version	String	Add-on version.
input	Object	Add-on installation parameters.
stable	Boolean	Whether the add-on version is a stable release.
translate	Object	Translation information used by the GUI.
supportVersions	Array of SupportVersions objects	Cluster versions that support the add-on.
creationTimestamp	String	Creation time.
updateTimestamp	String	Update time.

Table 4-886 SupportVersions

Parameter	Type	Description
clusterType	String	Cluster type that supports the add-on.
clusterVersion	Array of strings	Cluster versions that support the add-on. The value is a regular expression.
category	Array of strings	The cluster types to which the settings are applied Options: • CCE: CCE standard clusters • Turbo: CCE Turbo clusters • Autopilot: CCE Autopilot clusters Default value: If this parameter is left blank, CCE standard and Turbo clusters are used by default.

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "items": [ {
    "kind": "Addon",
    "apiVersion": "v3",
    "metadata": {
      "uid": "coredns",
      "name": "coredns",
      "alias": "coredns",
      "creationTimestamp": "2018-11-04T16:15:56Z",
      "updateTimestamp": "2022-01-11T14:32:10Z"
    },
    "spec": {
      "type": "helm",
      "require": true,
      "labels": [ "ServiceDiscovery" ],
      "logoURL": "",
      "description": "CoreDNS is a DNS server that chains plugins and provides Kubernetes DNS Services",
      "versions": [ {
        "version": "1.13.6",
        "input": {
          "basic": {
            "cluster_ip": "10.247.3.10",
            "ipv6": false,
            "platform": "linux-amd64",
            "swr_addr": "100.125.16.65:20202",
            "swr_user": "hwofficial"
          },
          "parameters": {
            "custom": {
              "stub_domains": "",
              "upstream_nameservers": ""
            },
            "flavor1": {
              "name": 2500,
              "replicas": 2,
              "resources": [ {
                "limitsCpu": "500m",
                "limitsMem": "512Mi",
                "name": "coredns",
                "requestsCpu": "500m",
                "requestsMem": "512Mi"
              } ]
            },
            "flavor2": {
              "name": 5000,
              "replicas": 2,
              "resources": [ {
                "limitsCpu": "1000m",
                "limitsMem": "1024Mi",
                "name": "coredns",
                "requestsCpu": "1000m",
                "requestsMem": "1024Mi"
              } ]
            },
            "flavor3": {
              "name": 10000,
              "replicas": 2,
              "resources": [ {
                "limitsCpu": "2000m",
                "limitsMem": "2048Mi",
                "name": "coredns",
                "requestsCpu": "2000m",
                "requestsMem": "2048Mi"
              } ]
            },
            "flavor4": {
              "name": 20000,
```

```
"replicas" : 4,
"resources" : [ {
  "limitsCpu" : "2000m",
  "limitsMem" : "2048Mi",
  "name" : "coredns",
  "requestsCpu" : "2000m",
  "requestsMem" : "2048Mi"
} ]
}
},
"stable" : true,
"translate" : {
  "en_US" : {
    "addon" : {
      "changeLog" : "Support for clusters with new version",
      "description" : "CoreDNS is a DNS server that chains plugins and provides Kubernetes DNS
Services"
    },
    "description" : {
      "Parameters.custom.stub_domains" : "The target nameserver may itself be a Kubernetes service.
For instance, you can run your own copy of dnsmasq to export custom DNS names into the ClusterDNS
namespace, a JSON map using a DNS suffix key (e.g. \"acme.local\") and a value consisting of a JSON array
of DNS IPs.",
      "Parameters.custom.upstream_nameservers" : "If specified, then the values specified replace the
nameservers taken by default from the node's /etc/resolv.conf. Limits:a maximum of three upstream
nameservers can be specified, A JSON array of DNS IPs.",
      "Parameters.flavor1.description" : "Concurrent domain name resolution ability - External domain
name: 2500 qps, Internal domain name: 10000 qps",
      "Parameters.flavor1.name" : 2500,
      "Parameters.flavor2.description" : "Concurrent domain name resolution ability - External domain
name: 5000 qps, Internal domain name: 20000 qps",
      "Parameters.flavor2.name" : 5000,
      "Parameters.flavor3.description" : "Concurrent domain name resolution ability - External domain
name: 10000 qps, Internal domain name: 40000 qps",
      "Parameters.flavor3.name" : 10000,
      "Parameters.flavor4.description" : "Concurrent domain name resolution ability - External domain
name: 20000 qps, Internal domain name: 80000 qps",
      "Parameters.flavor4.name" : 20000
    },
    "key" : {
      "Parameters.custom.stub_domains" : "stub domain",
      "Parameters.custom.upstream_nameservers" : "upstream nameservers"
    }
  },
  "fr_FR" : {
    "addon" : {
      "changeLog" : "Prise en charge des clusters avec une nouvelle version",
      "description" : "Un serveur DNS qui enchaîne les plug-ins et fournit des services DNS Kubernetes."
    },
    "description" : {
      "Parameters.custom.stub_domains" : "Le serveur de noms cible peut lui-même être un service
Kubernetes. Par exemple, vous pouvez exécuter votre propre copie de dnsmasq pour exporter des noms
DNS personnalisés dans l'espace de noms ClusterDNS, une carte JSON à l'aide d'une clé de suffixe DNS (par
exemple, «acme.local») et une valeur constituée d'un tableau JSON d'adresses IP DNS.",
      "Parameters.custom.upstream_nameservers" : "Si spécifié, les valeurs spécifiées remplacent les
serveurs de noms pris par défaut dans le fichier /etc/resolv.conf du nœud. Limites: un maximum de trois
serveurs de noms en amont peuvent être spécifiés, un tableau JSON d'adresses IP DNS.",
      "Parameters.flavor1.description" : "Capacité de résolution de nom de domaine simultanée - Nom
de domaine externe: 2500 qps, Nom de domaine interne: 10000 qp",
      "Parameters.flavor1.name" : 2500,
      "Parameters.flavor2.description" : "Capacité de résolution de nom de domaine simultanée - Nom
de domaine externe: 5000 qps, Nom de domaine interne: 20000 qp",
      "Parameters.flavor2.name" : 5000,
      "Parameters.flavor3.description" : "Capacité de résolution de nom de domaine simultanée - Nom
de domaine externe: 10000 qps, Nom de domaine interne: 40000 qp",
      "Parameters.flavor3.name" : 10000,
      "Parameters.flavor4.description" : "Capacité de résolution de nom de domaine simultanée - Nom
de domaine externe: 20000 qps, Nom de domaine interne: 80000 qp",
```

```
        "Parameters.flavor4.name" : 20000
    },
    "key" : {
        "Parameters.custom.stub_domains" : "domaine stub",
        "Parameters.custom.upstream_nameservers" : "serveurs de noms en amont"
    }
},
"zh_CN" : {
    "addon" : {
        "changeLog" : "",
        "description" : ""
    },
    "description" : {
        "Parameters.custom.stub_domains" : "",
        "Parameters.custom.upstream_nameservers" : "",
        "Parameters.flavor1.description" : "",
        "Parameters.flavor1.name" : 2500,
        "Parameters.flavor2.description" : "",
        "Parameters.flavor2.name" : 5000,
        "Parameters.flavor3.description" : "",
        "Parameters.flavor3.name" : 10000,
        "Parameters.flavor4.description" : "",
        "Parameters.flavor4.name" : 20000
    },
    "key" : {
        "Parameters.custom.stub_domains" : "",
        "Parameters.custom.upstream_nameservers" : ""
    }
}
},
"supportVersions" : [ {
    "clusterType" : "VirtualMachine",
    "clusterVersion" : [ "v1.13.*" ]
}, {
    "clusterType" : "BareMetal",
    "clusterVersion" : [ "v1.13.*" ]
}, {
    "clusterType" : "ARM64",
    "clusterVersion" : [ "v1.13.*" ]
} ],
"creationTimestamp" : "2021-03-18T12:51:05Z",
"updateTimestamp" : "2021-03-18T12:51:05Z"
} ]
} ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListAddonTemplatesSolution {

    public static void main(String[] args) {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ListAddonTemplatesRequest request = new ListAddonTemplatesRequest();
try {
    ListAddonTemplatesResponse response = client.listAddonTemplates(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAddonTemplatesRequest()
        response = client.list_addon_templates(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAddonTemplatesRequest{}
    response, err := client.ListAddonTemplates(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.6.3 Updating an Add-on Instance

Function

This API is used to update an add-on instance.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/addons/{id}

Table 4-887 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Add-on instance ID.

Request Parameters

Table 4-888 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-889 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	API type. The value is fixed at Addon and cannot be changed. Any user-defined value is invalid.
apiVersion	Yes	String	API version. The value is fixed at v3 and cannot be changed. Any user-defined value is invalid.
metadata	Yes	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.
spec	Yes	InstanceRequestSpec object	Detailed description about add-on installation or upgrade.

Table 4-890 AddonMetadata

Parameter	Mandatory	Type	Description
uid	No	String	Unique ID
name	No	String	Add-on name
alias	No	String	Add-on alias

Parameter	Mandatory	Type	Description
labels	No	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	No	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	No	String	Update time.
creationTimestamp	No	String	Creation time.

Table 4-891 InstanceRequestSpec

Parameter	Mandatory	Type	Description
version	No	String	Version of the add-on to install or upgrade, for example, 1.0.0 . - Installation: This parameter is optional. If not specified, the latest version supported by the cluster is used. - Upgrade: This parameter is mandatory. The version number must be specified.
clusterID	Yes	String	Cluster ID.

Parameter	Mandatory	Type	Description
values	Yes	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). During the add-on upgrade, you need to specify all the installation parameters. If the parameters are not specified, the default values in the add-on template are used. The current add-on installation parameters can be obtained through the API for querying add-on instances.
addonTemplateName	Yes	String	Name of the add-on template to be installed, for example, coredns .

Response Parameters

Status code: 200

Table 4-892 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.
spec	InstanceSpec object	Detailed description about the add-on instance.
status	AddonInstanceStatus object	Add-on instance status.

Table 4-893 AddonMetadata

Parameter	Type	Description
uid	String	Unique ID
name	String	Add-on name

Parameter	Type	Description
alias	String	Add-on alias
labels	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	String	Update time.
creationTimestamp	String	Creation time.

Table 4-894 InstanceSpec

Parameter	Type	Description
clusterID	String	Cluster ID.
version	String	Add-on template version, for example, 1.0.0 .
addonTemplateName	String	Add-on template name, for example, coredns .
addonTemplateType	String	Add-on template type.
addonTemplateLogo	String	URL for obtaining the add-on template logo.
addonTemplateLabels	Array of strings	Labels of the add-on template.
description	String	Add-on template description.
values	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). Set the parameters accordingly.

Table 4-895 AddonInstanceStatus

Parameter	Type	Description
status	String	Add-on instance status. Options: • running: All of the add-on instances are running. This specifies that the add-on runs properly. • abnormal: The add-on instances are abnormal and the add-on cannot be used. You can click the add-on name to view exceptions. • installing: The add-on is being installed. • installFailed: Installing the add-on failed. In this case, uninstall the add-on and then reinstall it. • upgrading: The add-on is being upgraded. • upgradeFailed: Upgrading the add-on failed. In this case, upgrade the add-on again or uninstall the add-on and reinstall it. • deleting: The add-on is being deleted. • deleteFailed: Deleting the add-on failed. In this case, uninstall the add-on again. • deleteSuccess: Deleting the add-on succeeded. • available: Only some instances of the add-on are running. This specifies that some functions of the add-on are available. • rollbacking: The add-on is being rolled back. • rollbackFailed: Rolling back the add-on failed. In this case, roll back the add-on again or uninstall the add-on and reinstall it. • unknown: The add-on chart instance does not exist.
Reason	String	Cause of the add-on installation failure.
message	String	Installation error details.
targetVersions	Array of strings	Cluster versions that support the add-on upgrade.

Parameter	Type	Description
currentVersion	Versions object	Information about the current add-on version.
isRollbackable	Boolean	Whether the add-on version can be rolled back to the source version.
previousVersion	String	The add-on version before upgrade or rollback

Table 4-896 Versions

Parameter	Type	Description
version	String	Add-on version.
input	Object	Add-on installation parameters.
stable	Boolean	Whether the add-on version is a stable release.
translate	Object	Translation information used by the GUI.
supportVersions	Array of SupportVersions objects	Cluster versions that support the add-on.
creationTimestamp	String	Creation time.
updateTimestamp	String	Update time.

Table 4-897 SupportVersions

Parameter	Type	Description
clusterType	String	Cluster type that supports the add-on.
clusterVersion	Array of strings	Cluster versions that support the add-on. The value is a regular expression.

Parameter	Type	Description
category	Array of strings	The cluster types to which the settings are applied Options: • CCE: CCE standard clusters • Turbo: CCE Turbo clusters • Autopilot: CCE Autopilot clusters Default value: If this parameter is left blank, CCE standard and Turbo clusters are used by default.

Example Requests

Update the everest add-on to version 2.1.30.

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.upgrade/type": "upgrade"
    }
  },
  "spec": {
    "clusterID": "*****",
    "version": "2.1.30",
    "addonTemplateName": "everest",
    "values": {
      "basic": {
        "bms_url": "*****",
        "driver_init_image_version": "2.1.30",
        "ecsEndpoint": "*****",
        "everest_image_version": "2.1.30",
        "evs_url": "*****",
        "iam_url": "*****",
        "ims_url": "*****",
        "obs_url": "*****",
        "platform": "linux-amd64",
        "sfs30_url": "*****",
        "sfs_turbo_url": "*****",
        "sfs_url": "*****",
        "supportHcs": false,
        "swr_addr": "*****",
        "swr_user": "hwofficial",
        "rbac_enabled": true,
        "cluster_version": "v1.23"
      },
      "flavor": {
        "description": "High available",
        "name": "HA",
        "replicas": 2,
        "resources": [ {
          "limitsCpu": "250m",
          "limitsMem": "2000Mi",
          "name": "everest-csi-controller",
          "requestsCpu": "250m",
          "requestsMem": "1500Mi"
        } ], {
```



```
    "limitsCpu" : "500m",
    "limitsMem" : "300Mi",
    "name" : "everest-csi-driver",
    "requestsCpu" : "100m",
    "requestsMem" : "300Mi"
  } ],
  "category" : [ "CCE", "Turbo" ]
},
"custom" : {
  "cluster_id" : "*****",
  "csi_attacher_detach_worker_threads" : "60",
  "csi_attacher_worker_threads" : "60",
  "default_vpc_id" : "*****",
  "disable_auto_mount_secret" : false,
  "enable_node_attacher" : true,
  "flow_control" : { },
  "multiAZEnabled" : false,
  "over_subscription" : "80",
  "project_id" : "*****",
  "volume_attaching_flow_ctrl" : "0"
}
}
}
```

Example Responses

Status code: 200

OK

```
{
  "kind" : "Addon",
  "apiVersion" : "v3",
  "metadata" : {
    "uid" : "*****",
    "name" : "everest",
    "alias" : "everest",
    "creationTimestamp" : "2023-07-03T10:57:43Z",
    "updateTimestamp" : "2023-07-03T11:24:05Z"
  },
  "spec" : {
    "clusterID" : "*****",
    "version" : "2.1.30",
    "addonTemplateName" : "everest",
    "addonTemplateType" : "helm",
    "addonTemplateLogo" : "*****",
    "addonTemplateLabels" : [ "Storage" ],
    "description" : "",
    "values" : {
      "basic" : {
        "bms_url" : "*****",
        "cluster_version" : "v1.23",
        "driver_init_image_version" : "2.1.30",
        "ecsEndpoint" : "*****",
        "everest_image_version" : "2.1.30",
        "evs_url" : "*****",
        "iam_url" : "*****",
        "ims_url" : "*****",
        "obs_url" : "*****",
        "platform" : "linux-amd64",
        "rbac_enabled" : true,
        "sfs30_url" : "*****",
        "sfs_turbo_url" : "*****",
        "sfs_url" : "*****",
        "supportHcs" : false,
        "swr_addr" : "*****",
        "swr_user" : "hwofficial"
      },
    }
  },
}
```

```
"custom" : {
  "cluster_id" : "*****",
  "cluster_version" : "v1.23.8-r0",
  "csi_attacher_detach_worker_threads" : "60",
  "csi_attacher_worker_threads" : "60",
  "default_vpc_id" : "*****",
  "disable_auto_mount_secret" : false,
  "enable_node_attacher" : false,
  "flow_control" : { },
  "multiAZEnabled" : false,
  "over_subscription" : "80",
  "project_id" : "*****",
  "volume_attaching_flow_ctrl" : "0"
},
"flavor" : {
  "category" : [ "CCE", "Turbo" ],
  "description" : "Has only one instance",
  "name" : "Single",
  "replicas" : 1,
  "resources" : [ {
    "limitsCpu" : "250m",
    "limitsMem" : "600Mi",
    "name" : "everest-csi-controller",
    "requestsCpu" : "250m",
    "requestsMem" : "600Mi"
  }, {
    "limitsCpu" : "100m",
    "limitsMem" : "300Mi",
    "name" : "everest-csi-driver",
    "requestsCpu" : "100m",
    "requestsMem" : "300Mi"
  } ]
},
"systemAutoInject" : {
  "cluster" : {
    "clusterID" : "*****",
    "clusterNetworkMode" : "vpc-router",
    "clusterVersion" : "v1.23.8-r0"
  },
  "user" : {
    "projectID" : "*****"
  }
}
},
"status" : {
  "status" : "upgrading",
  "Reason" : "addon upgrading",
  "message" : "",
  "targetVersions" : null,
  "isRollbackable" : false,
  "currentVersion" : {
    "version" : "2.1.30",
    "input" : {
      "basic" : {
        "bms_url" : "*****",
        "driver_init_image_version" : "2.1.30",
        "ecsEndpoint" : "*****",
        "everest_image_version" : "2.1.30",
        "evs_url" : "*****",
        "iam_url" : "*****",
        "ims_url" : "*****",
        "obs_url" : "*****",
        "platform" : "*****",
        "sfs30_url" : "*****",
        "sfs_turbo_url" : "*****",
        "sfs_url" : "*****",
        "supportHcs" : false,
        "swr_addr" : "*****",
```

```
"swr_user" : "hwofficial"
},
"parameters" : {
  "common" : {
    "defaultVPCId" : 1234567
  },
  "custom" : {
    "cluster_id" : "",
    "csi_attacher_detach_worker_threads" : "60",
    "csi_attacher_worker_threads" : "60",
    "default_vpc_id" : "",
    "disable_auto_mount_secret" : false,
    "enable_node_attacher" : false,
    "flow_control" : { },
    "multiAZEnabled" : false,
    "over_subscription" : "80",
    "project_id" : "",
    "volume_attaching_flow_ctrl" : "0"
  },
  "flavor1" : {
    "description" : "High available",
    "name" : "HA",
    "replicas" : 2,
    "resources" : [ {
      "limitsCpu" : "250m",
      "limitsMem" : "1500Mi",
      "name" : "everest-csi-controller",
      "requestsCpu" : "250m",
      "requestsMem" : "600Mi"
    }, {
      "limitsCpu" : "500m",
      "limitsMem" : "300Mi",
      "name" : "everest-csi-driver",
      "requestsCpu" : "100m",
      "requestsMem" : "300Mi"
    } ]
  },
  "flavor2" : {
    "description" : "Has only one instance",
    "name" : "Single",
    "replicas" : 1,
    "resources" : [ {
      "limitsCpu" : "250m",
      "limitsMem" : "600Mi",
      "name" : "everest-csi-controller",
      "requestsCpu" : "250m",
      "requestsMem" : "600Mi"
    }, {
      "limitsCpu" : "100m",
      "limitsMem" : "300Mi",
      "name" : "everest-csi-driver",
      "requestsCpu" : "100m",
      "requestsMem" : "300Mi"
    } ]
  },
  "flavor3" : {
    "description" : "custom resources",
    "name" : "custom-resources",
    "replicas" : 2,
    "resources" : [ {
      "limitsCpu" : "250m",
      "limitsMem" : "2000Mi",
      "name" : "everest-csi-controller",
      "requestsCpu" : "250m",
      "requestsMem" : "1500Mi"
    }, {
      "limitsCpu" : "500m",
      "limitsMem" : "300Mi",
      "name" : "everest-csi-driver",
```

```
        "requestsCpu" : "100m",
        "requestsMem" : "300Mi"
    } ]
    }
},
"stable" : true,
"translate" : {
    "en_US" : {
        "addon" : {
            "changeLog" : "",
            "description" : ""
        },
        "description" : {
            "Parameters.flavor1.description" : "Deploy the add-on with two instances, delivering high
availability but requiring more compute resources.",
            "Parameters.flavor1.name" : "HA",
            "Parameters.flavor2.description" : "Deploy the add-on with one instance.",
            "Parameters.flavor2.name" : "Standalone",
            "Parameters.flavor3.name" : "Custom"
        }
    },
    "fr_FR" : {
        "addon" : {
            "changeLog" : "",
            "description" : ""
        },
        "description" : {
            "Parameters.flavor1.description" : "Déployez avec deux instances, haute disponibilité.",
            "Parameters.flavor1.name" : "HA",
            "Parameters.flavor2.description" : "Déployez avec une seule instance.",
            "Parameters.flavor2.name" : "Célibataire",
            "Parameters.flavor3.name" : "Douane"
        }
    },
    "zh_CN" : {
        "addon" : {
            "changeLog" : "",
            "description" : ""
        },
        "description" : {
            "Parameters.flavor1.description" : "Dual-instance deployment features high availability but requires
more compute resources.",
            "Parameters.flavor1.name" : "High availability",
            "Parameters.flavor2.description" : "Single-instance deployment",
            "Parameters.flavor2.name" : "Single instance",
            "Parameters.flavor3.description" : "Custom resource specifications for deployment",
            "Parameters.flavor3.name" : "Custom"
        }
    }
},
"supportVersions" : null,
"creationTimestamp" : "2023-05-12T16:10:05Z",
"updateTimestamp" : "2023-05-12T16:10:05Z"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Update the everest add-on to version 2.1.30.

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.Map;
import java.util.HashMap;

public class UpdateAddonInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdateAddonInstanceRequest request = new UpdateAddonInstanceRequest();
        request.withId("{id}");
        InstanceRequest body = new InstanceRequest();
        Map<String, Object> listSpecValues = new HashMap<>();
        listSpecValues.put("basic", "{\n\"evs_url\": \"*****\",\n\"obs_url\": \"*****\",\n\"cluster_version\n\\\": \"v1.23\\\", \"supportHcs\": false,\n\"iam_url\": \"*****\",\n\"rbac_enabled\": true,\n\"ecsEndpoint\": \"*****\",\n\"sfs_url\n\\\": \"*****\",\n\"platform\": \"linux-amd64\",\n\"ims_url\": \"*****\",\n\"driver_init_image_version\n\\\": \"2.1.30\\\", \"sfs30_url\": \"*****\",\n\"sfs_turbo_url\": \"*****\",\n\"swr_user\": \"hwofficial\\\", \"bms_url\": \"*****\n\\\", \"everest_image_version\": \"2.1.30\\\", \"swr_addr\": \"*****\"}");
        listSpecValues.put("flavor", "{\n\"replicas\": 2,\n\"name\": \"HA\\\", \"description\": \"High available\n\\\", \"resources\": [{\n\"limitsCpu\": \"250m\\\", \"name\": \"everest-csi-controller\\\", \"limitsMem\": \"2000Mi\n\\\", \"requestsMem\": \"1500Mi\\\", \"requestsCpu\": \"250m\\\"}], {\n\"limitsCpu\": \"500m\\\", \"name\": \"everest-csi-driver\\\", \"limitsMem\": \"300Mi\\\", \"requestsMem\": \"300Mi\\\", \"requestsCpu\": \"100m\\\"}], \"category\": [\"CCE\n\\\", \"Turbo\"]});");
        listSpecValues.put("custom", "{\n\"csi_attacher_worker_threads\": \"60\\\", \"cluster_id\": \"*****\n\\\", \"csi_attacher_detach_worker_threads\": \"60\\\", \"disable_auto_mount_secret\": false,\n\"over_subscription\n\\\": \"80\\\", \"project_id\": \"*****\\\", \"enable_node_attacher\": true,\n\"volume_attaching_flow_ctrl\n\\\": \"0\\\", \"multiAZEnabled\": false,\n\"flow_control\": {}, \"default_vpc_id\": \"*****\"}");
        InstanceRequestSpec specbody = new InstanceRequestSpec();
        specbody.withVersion("2.1.30")
            .withClusterID("*****")
            .withValues(listSpecValues)
            .withAddonTemplateName("everest");
        Map<String, String> listMetadataAnnotations = new HashMap<>();
        listMetadataAnnotations.put("addon.upgrade/type", "upgrade");
        AddonMetadata metadatabody = new AddonMetadata();
        metadatabody.withAnnotations(listMetadataAnnotations);
        body.withSpec(specbody);
        body.withMetadata(metadatabody);
        body.withApiVersion("v3");
        body.withKind("Addon");
        request.withBody(body);
        try {
            UpdateAddonInstanceResponse response = client.updateAddonInstance(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
```

```
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

Update the everest add-on to version 2.1.30.

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateAddonInstanceRequest()
        request.id = "{id}"
        listValuesSpec = {
            "basic": {"\evs_url\":"*****","\obs_url\":"*****","\cluster_version\":"v1.23","\supportHcs
\":"false","\iam_url\":"*****","\rbac_enabled\":"true","\ecsEndpoint\":"*****","\sfs_url\":"*****","\platform
\":"linux-amd64","\ims_url\":"*****","\driver_init_image_version\":"2.1.30","\sfs30_url\":"*****
","\sfs_turbo_url\":"*****","\swr_user\":"hwofficial","\bms_url\":"*****","\everest_image_version
\":"2.1.30","\swr_addr\":"*****"}},
            "flavor": {"\replicas\":"2","\name\":"HA","\description\":"High available","\resources\":"
[{"limitsCpu\":"250m","\name\":"everest-csi-controller","\limitsMem\":"2000Mi","\requestsMem
\":"1500Mi","\requestsCpu\":"250m"}, {"limitsCpu\":"500m","\name\":"everest-csi-driver","\limitsMem
\":"300Mi","\requestsMem\":"300Mi","\requestsCpu\":"100m"}]}, {"category\":"CCE","\Turbo\":"}],
            "custom": {"\csi_attacher_worker_threads\":"60","\cluster_id\":"*****
","\csi_attacher_detach_worker_threads\":"60","\disable_auto_mount_secret\":"false","\over_subscription
\":"80","\project_id\":"*****","\enable_node_attacher\":"true","\volume_attaching_flow_ctrl
\":"0","\multiAZEnabled\":"false","\flow_control\":"{}","\default_vpc_id\":"*****"}
        }
        specbody = InstanceRequestSpec(
            version="2.1.30",
            cluster_id="*****",
            values=listValuesSpec,
            addon_template_name="everest"
        )
        listAnnotationsMetadata = {
            "addon.upgrade/type": "upgrade"
        }
        metadatabody = AddonMetadata(
            annotations=listAnnotationsMetadata
        )
    }
```

```
request.body = InstanceRequest(
    spec=specbody,
    metadata=metadatabody,
    api_version="v3",
    kind="Addon"
)
response = client.update_addon_instance(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Update the everest add-on to version 2.1.30.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateAddonInstanceRequest{}
    request.Id = "{id}"
    var listValuesSpec = map[string]interface{}{
        "basic": "{\n\"evs_url\": \"*****\", \"obs_url\": \"*****\", \"cluster_version\": \"v1.23\", \"supportHcs\n\n\": false, \"iam_url\": \"*****\", \"rbac_enabled\": true, \"ecsEndpoint\": \"*****\", \"sfs_url\": \"*****\", \"platform\n\n\": \"linux-amd64\", \"ims_url\": \"*****\", \"driver_init_image_version\": \"2.1.30\", \"sfs30_url\": \"*****\n\n\", \"sfs_turbo_url\": \"*****\", \"swr_user\": \"hwofficial\", \"bms_url\": \"*****\", \"everest_image_version\n\n\": \"2.1.30\", \"swr_addr\": \"*****\"}",
        "flavor": "{\n\"replicas\": 2, \"name\": \"HA\", \"description\": \"High available\", \"resources\": {\n\n\"limitsCpu\n\n\": \"250m\", \"name\": \"everest-csi-controller\", \"limitsMem\": \"2000Mi\", \"requestsMem\": \"1500Mi\n\n\", \"requestsCpu\": \"250m\"}, {\n\n\"limitsCpu\n\n\": \"500m\", \"name\": \"everest-csi-driver\", \"limitsMem\": \"300Mi\n\n\", \"requestsMem\": \"300Mi\", \"requestsCpu\": \"100m\"}, {\n\n\"category\n\n\": \"CCE\", \"Turbo\n\n\": \"\"},\n\n\"custom\": {\n\n\"csi_attacher_worker_threads\n\n\": \"60\", \"cluster_id\n\n\": \"*****\n\n\", \"csi_attacher_detach_worker_threads\n\n\": \"60\", \"disable_auto_mount_secret\n\n\": false, \"over_subscription\n\n\": \"80\", \"project_id\n\n\": \"*****\", \"enable_node_attacher\n\n\": true, \"volume_attaching_flow_ctrl\n\n\": \"0\", \"multiAZEnabled\n\n\": false, \"flow_control\n\n\": {}, \"default_vpc_id\n\n\": \"*****\"}",
    }
    versionSpec := "2.1.30"
    specbody := &model.InstanceRequestSpec{
        Version: &versionSpec,
        ClusterID: "*****",
    }
}
```

```
    Values: listValuesSpec,
    AddonTemplateName: "everest",
  }
  var listAnnotationsMetadata = map[string]string{
    "addon.upgrade/type": "upgrade",
  }
  metadatabody := &model.AddonMetadata{
    Annotations: listAnnotationsMetadata,
  }
  request.Body = &model.InstanceRequest{
    Spec: specbody,
    Metadata: metadatabody,
    ApiVersion: "v3",
    Kind: "Addon",
  }
  response, err := client.UpdateAddonInstance(request)
  if err == nil {
    fmt.Printf("%v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.6.4 Rolling Back an Add-on Instance

Function

Roll back the add-on instance version to the source version. Only when the **status.isRollbackable** parameter is set to **true**, and the add-on instance is in the **running**, **available**, **abnormal**, **upgradeFailed**, or **rollbackFailed** state, the version can be rolled back to the source version.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/addons/{id}/operation/rollback

Table 4-898 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Add-on instance ID

Request Parameters

Table 4-899 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-900 Request body parameters

Parameter	Mandatory	Type	Description
clusterID	Yes	String	Cluster ID

Response Parameters

Status code: 200

Table 4-901 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.

Parameter	Type	Description
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.
spec	InstanceSpec object	Detailed description about the add-on instance.
status	AddonInstanceStatus object	Add-on instance status.

Table 4-902 AddonMetadata

Parameter	Type	Description
uid	String	Unique ID
name	String	Add-on name
alias	String	Add-on alias
labels	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	String	Update time.
creationTimestamp	String	Creation time.

Table 4-903 InstanceSpec

Parameter	Type	Description
clusterID	String	Cluster ID.
version	String	Add-on template version, for example, 1.0.0 .

Parameter	Type	Description
addonTemplateName	String	Add-on template name, for example, coredns .
addonTemplateType	String	Add-on template type.
addonTemplateLogo	String	URL for obtaining the add-on template logo.
addonTemplateLabels	Array of strings	Labels of the add-on template.
description	String	Add-on template description.
values	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). Set the parameters accordingly.

Table 4-904 AddonInstanceStatus

Parameter	Type	Description
status	String	Add-on instance status. Options: • running: All of the add-on instances are running. This specifies that the add-on runs properly. • abnormal: The add-on instances are abnormal and the add-on cannot be used. You can click the add-on name to view exceptions. • installing: The add-on is being installed. • installFailed: Installing the add-on failed. In this case, uninstall the add-on and then reinstall it. • upgrading: The add-on is being upgraded. • upgradeFailed: Upgrading the add-on failed. In this case, upgrade the add-on again or uninstall the add-on and reinstall it. • deleting: The add-on is being deleted. • deleteFailed: Deleting the add-on failed. In this case, uninstall the add-on again. • deleteSuccess: Deleting the add-on succeeded. • available: Only some instances of the add-on are running. This specifies that some functions of the add-on are available. • rollbacking: The add-on is being rolled back. • rollbackFailed: Rolling back the add-on failed. In this case, roll back the add-on again or uninstall the add-on and reinstall it. • unknown: The add-on chart instance does not exist.
Reason	String	Cause of the add-on installation failure.
message	String	Installation error details.
targetVersions	Array of strings	Cluster versions that support the add-on upgrade.

Parameter	Type	Description
currentVersion	Versions object	Information about the current add-on version.
isRollbackable	Boolean	Whether the add-on version can be rolled back to the source version.
previousVersion	String	The add-on version before upgrade or rollback

Table 4-905 Versions

Parameter	Type	Description
version	String	Add-on version.
input	Object	Add-on installation parameters.
stable	Boolean	Whether the add-on version is a stable release.
translate	Object	Translation information used by the GUI.
supportVersions	Array of SupportVersions objects	Cluster versions that support the add-on.
creationTimestamp	String	Creation time.
updateTimestamp	String	Update time.

Table 4-906 SupportVersions

Parameter	Type	Description
clusterType	String	Cluster type that supports the add-on.
clusterVersion	Array of strings	Cluster versions that support the add-on. The value is a regular expression.

Parameter	Type	Description
category	Array of strings	The cluster types to which the settings are applied Options: • CCE: CCE standard clusters • Turbo: CCE Turbo clusters • Autopilot: CCE Autopilot clusters Default value: If this parameter is left blank, CCE standard and Turbo clusters are used by default.

Example Requests

```
{
  "clusterID" : "*****"
}
```

Example Responses

Status code: 200

Add-on instance version is rolled back.

```
{
  "kind" : "Addon",
  "apiVersion" : "v3",
  "metadata" : {
    "uid" : "*****",
    "name" : "everest",
    "alias" : "everest",
    "creationTimestamp" : "2023-03-15T02:48:01Z",
    "updateTimestamp" : "2023-03-15T04:18:45Z"
  },
  "spec" : {
    "clusterID" : "*****",
    "version" : "2.1.16",
    "addonTemplateName" : "everest",
    "addonTemplateType" : "helm",
    "addonTemplateLogo" : "*****",
    "addonTemplateLabels" : [ "Storage" ],
    "description" : "Everest is a cloud native container storage system based on CSI, used\nto support cloud storages services for Kubernetes",
    "values" : {
      "basic" : {
        "base_image" : "euleros",
        "bms_url" : "*****",
        "cluster_version" : "v1.25",
        "driver_init_image_version" : "2.1.16",
        "ecsEndpoint" : "*****",
        "euleros_version" : "2.2.5",
        "everest_image_version" : "2.1.16",
        "evs_url" : "*****",
        "iam_url" : "*****",
        "ims_url" : "*****",
        "obs_url" : "*****",
        "platform" : "linux-amd64",
        "rbac_enabled" : true,

```

```
"sfs30_url" : "*****",
"sfs_turbo_url" : "*****",
"sfs_url" : "*****",
"supportHcs" : false,
"swr_addr" : "*****",
"swr_user" : "*****"
},
"custom" : {
  "cluster_id" : "*****",
  "cluster_version" : "v1.25.3-r0",
  "csi_attacher_detach_worker_threads" : "60",
  "csi_attacher_worker_threads" : "60",
  "default_vpc_id" : "*****",
  "disable_auto_mount_secret" : false,
  "enable_node_attacher" : false,
  "flow_control" : { },
  "multiAZEnabled" : false,
  "over_subscription" : "80",
  "project_id" : "*****",
  "volume_attaching_flow_ctrl" : "0"
},
"flavor" : {
  "category" : [ "CCE", "Turbo" ],
  "description" : "High available",
  "name" : "HA",
  "replicas" : 2,
  "resources" : [ {
    "limitsCpu" : "250m",
    "limitsMem" : "1500Mi",
    "name" : "everest-csi-controller",
    "requestsCpu" : "250m",
    "requestsMem" : "600Mi"
  }, {
    "limitsCpu" : "500m",
    "limitsMem" : "300Mi",
    "name" : "everest-csi-driver",
    "requestsCpu" : "100m",
    "requestsMem" : "300Mi"
  } ]
},
"multiAZPreferred" : {
  "podAntiAffinity" : {
    "preferredDuringSchedulingIgnoredDuringExecution" : [ {
      "podAffinityTerm" : {
        "labelSelector" : {
          "matchExpressions" : [ {
            "key" : "app",
            "operator" : "In",
            "values" : [ "everest-csi-controller" ]
          } ]
        }
      }
    ]
  },
  "topologyKey" : "topology.kubernetes.io/zone"
},
"weight" : 100
} ]
},
"multiAZRequired" : {
  "podAntiAffinity" : {
    "requiredDuringSchedulingIgnoredDuringExecution" : [ {
      "labelSelector" : {
        "matchExpressions" : [ {
          "key" : "app",
          "operator" : "In",
          "values" : [ "everest-csi-controller" ]
        } ]
      }
    ]
  },
  "topologyKey" : "topology.kubernetes.io/zone"
} ]
}
```

```
}
},
"systemAutoInject" : {
  "cluster" : {
    "category" : "CCE",
    "clusterID" : "*****",
    "clusterNetworkMode" : "vpc-router",
    "clusterVersion" : "v1.25.3-r0"
  },
  "user" : {
    "projectID" : "*****"
  }
},
"tolerations" : [ {
  "effect" : "NoExecute",
  "key" : "node.kubernetes.io/not-ready",
  "operator" : "Exists",
  "tolerationSeconds" : 60
}, {
  "effect" : "NoExecute",
  "key" : "node.kubernetes.io/unreachable",
  "operator" : "Exists",
  "tolerationSeconds" : 60
} ]
}
},
"status" : {
  "status" : "rollbacking",
  "Reason" : "Rollback to 4",
  "message" : "",
  "targetVersions" : [ "2.1.18", "2.1.19" ],
  "isRollbackable" : false,
  "previousVersion" : "2.1.19",
  "currentVersion" : {
    "version" : "2.1.16",
    "input" : {
      "basic" : {
        "bms_url" : "*****",
        "driver_init_image_version" : "2.1.16",
        "ecsEndpoint" : "*****",
        "everest_image_version" : "2.1.16",
        "evs_url" : "*****",
        "iam_url" : "*****",
        "ims_url" : "*****",
        "obs_url" : "*****",
        "platform" : "linux-amd64",
        "sfs30_url" : "*****",
        "sfs_turbo_url" : "*****",
        "sfs_url" : "*****",
        "supportHcs" : false,
        "swr_addr" : "*****",
        "swr_user" : "*****"
      },
      "parameters" : {
        "common" : {
          "defaultVPCId" : 0
        },
        "custom" : {
          "cluster_id" : "",
          "csi_attacher_detach_worker_threads" : "60",
          "csi_attacher_worker_threads" : "60",
          "default_vpc_id" : "",
          "disable_auto_mount_secret" : false,
          "enable_node_attacher" : false,
          "flow_control" : { },
          "multiAZEnabled" : false,
          "over_subscription" : "80",
          "project_id" : "",
          "volume_attaching_flow_ctrl" : "0"
        }
      }
    }
  }
}
```



```
,
"flavor1" : {
  "description" : "High available",
  "name" : "HA",
  "replicas" : 2,
  "resources" : [ {
    "limitsCpu" : "250m",
    "limitsMem" : "1500Mi",
    "name" : "everest-csi-controller",
    "requestsCpu" : "250m",
    "requestsMem" : "600Mi"
  }, {
    "limitsCpu" : "500m",
    "limitsMem" : "300Mi",
    "name" : "everest-csi-driver",
    "requestsCpu" : "100m",
    "requestsMem" : "300Mi"
  } ]
},
"flavor2" : {
  "description" : "Has only one instance",
  "name" : "Single",
  "replicas" : 1,
  "resources" : [ {
    "limitsCpu" : "250m",
    "limitsMem" : "600Mi",
    "name" : "everest-csi-controller",
    "requestsCpu" : "250m",
    "requestsMem" : "600Mi"
  }, {
    "limitsCpu" : "100m",
    "limitsMem" : "300Mi",
    "name" : "everest-csi-driver",
    "requestsCpu" : "100m",
    "requestsMem" : "300Mi"
  } ]
},
"flavor3" : {
  "description" : "custom resources",
  "name" : "custom-resources",
  "replicas" : 2,
  "resources" : [ {
    "limitsCpu" : "250m",
    "limitsMem" : "2000Mi",
    "name" : "everest-csi-controller",
    "requestsCpu" : "250m",
    "requestsMem" : "1500Mi"
  }, {
    "limitsCpu" : "500m",
    "limitsMem" : "300Mi",
    "name" : "everest-csi-driver",
    "requestsCpu" : "100m",
    "requestsMem" : "300Mi"
  } ]
}
},
"stable" : true,
"translate" : {
  "en_US" : {
    "addon" : {
      "changeLog" : "*****",
      "description" : "*****"
    },
    "description" : {
      "Parameters.flavor1.description" : "*****",
      "Parameters.flavor1.name" : "*****",
      "Parameters.flavor2.description" : "*****",
      "Parameters.flavor2.name" : "*****",
```

```
        "Parameters.flavor3.name" : "*****"
    },
    "zh_CN" : {
        "addon" : {
            "changeLog" : "*****",
            "description" : "*****"
        },
        "description" : {
            "Parameters.flavor1.description" : "*****",
            "Parameters.flavor1.name" : "*****",
            "Parameters.flavor2.description" : "*****",
            "Parameters.flavor2.name" : "*****",
            "Parameters.flavor3.description" : "*****",
            "Parameters.flavor3.name" : "*****"
        }
    }
},
"supportVersions" : null,
"creationTimestamp" : "2023-02-21T16:29:02Z",
"updateTimestamp" : "2023-02-22T06:49:50Z"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class RollbackAddonInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        RollbackAddonInstanceRequest request = new RollbackAddonInstanceRequest();
        request.withId("{id}");
        AddonInstanceRollbackRequest body = new AddonInstanceRollbackRequest();
        body.withClusterID("*****");
        request.withBody(body);
        try {
```

```
RollbackAddonInstanceResponse response = client.rollbackAddonInstance(request);
System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RollbackAddonInstanceRequest()
        request.id = "{id}"
        request.body = AddonInstanceRollbackRequest(
            cluster_id="*****"
        )
        response = client.rollback_addon_instance(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.RollbackAddonInstanceRequest{}
request.Id = "{id}"
request.Body = &model.AddonInstanceRollbackRequest{
    ClusterID: "*****",
}
response, err := client.RollbackAddonInstance(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Add-on instance version is rolled back.

Error Codes

See [Error Codes](#).

4.6.5 Deleting an Add-on Instance

Function

This API is used to delete an add-on instance.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /api/v3/addons/{id}

Table 4-907 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Add-on instance ID.

Table 4-908 Query Parameters

Parameter	Mandatory	Type	Description
cluster_id	No	String	Cluster ID (deprecated). For details about how to obtain the cluster ID, see How Do I Obtain Parameters in the API URI .

Request Parameters

Table 4-909 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-910 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.
spec	InstanceSpec object	Detailed description about the add-on instance.
status	AddonInstanceStatus object	Add-on instance status.

Table 4-911 AddonMetadata

Parameter	Type	Description
uid	String	Unique ID
name	String	Add-on name

Parameter	Type	Description
alias	String	Add-on alias
labels	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	String	Update time.
creationTimestamp	String	Creation time.

Table 4-912 InstanceSpec

Parameter	Type	Description
clusterID	String	Cluster ID.
version	String	Add-on template version, for example, 1.0.0 .
addonTemplateName	String	Add-on template name, for example, coredns .
addonTemplateType	String	Add-on template type.
addonTemplateLogo	String	URL for obtaining the add-on template logo.
addonTemplateLabels	Array of strings	Labels of the add-on template.
description	String	Add-on template description.
values	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). Set the parameters accordingly.

Table 4-913 AddonInstanceStatus

Parameter	Type	Description
status	String	Add-on instance status. Options: • running: All of the add-on instances are running. This specifies that the add-on runs properly. • abnormal: The add-on instances are abnormal and the add-on cannot be used. You can click the add-on name to view exceptions. • installing: The add-on is being installed. • installFailed: Installing the add-on failed. In this case, uninstall the add-on and then reinstall it. • upgrading: The add-on is being upgraded. • upgradeFailed: Upgrading the add-on failed. In this case, upgrade the add-on again or uninstall the add-on and reinstall it. • deleting: The add-on is being deleted. • deleteFailed: Deleting the add-on failed. In this case, uninstall the add-on again. • deleteSuccess: Deleting the add-on succeeded. • available: Only some instances of the add-on are running. This specifies that some functions of the add-on are available. • rollbacking: The add-on is being rolled back. • rollbackFailed: Rolling back the add-on failed. In this case, roll back the add-on again or uninstall the add-on and reinstall it. • unknown: The add-on chart instance does not exist.
Reason	String	Cause of the add-on installation failure.
message	String	Installation error details.
targetVersions	Array of strings	Cluster versions that support the add-on upgrade.

Parameter	Type	Description
currentVersion	Versions object	Information about the current add-on version.
isRollbackable	Boolean	Whether the add-on version can be rolled back to the source version.
previousVersion	String	The add-on version before upgrade or rollback

Table 4-914 Versions

Parameter	Type	Description
version	String	Add-on version.
input	Object	Add-on installation parameters.
stable	Boolean	Whether the add-on version is a stable release.
translate	Object	Translation information used by the GUI.
supportVersions	Array of SupportVersions objects	Cluster versions that support the add-on.
creationTimestamp	String	Creation time.
updateTimestamp	String	Update time.

Table 4-915 SupportVersions

Parameter	Type	Description
clusterType	String	Cluster type that supports the add-on.
clusterVersion	Array of strings	Cluster versions that support the add-on. The value is a regular expression.

Parameter	Type	Description
category	Array of strings	The cluster types to which the settings are applied Options: • CCE: CCE standard clusters • Turbo: CCE Turbo clusters • Autopilot: CCE Autopilot clusters Default value: If this parameter is left blank, CCE standard and Turbo clusters are used by default.

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "uid": "*****",
    "name": "everest",
    "alias": "everest",
    "creationTimestamp": "2023-07-03T10:57:43Z",
    "updateTimestamp": "2023-07-03T11:24:05Z"
  },
  "spec": {
    "clusterID": "*****",
    "version": "2.1.30",
    "addonTemplateName": "everest",
    "addonTemplateType": "helm",
    "addonTemplateLogo": "*****",
    "addonTemplateLabels": [ "Storage" ],
    "description": "",
    "values": {
      "basic": {
        "bms_url": "*****",
        "cluster_version": "v1.23",
        "driver_init_image_version": "2.1.30",
        "ecsEndpoint": "*****",
        "everest_image_version": "2.1.30",
        "evs_url": "*****",
        "iam_url": "*****",
        "ims_url": "*****",
        "obs_url": "*****",
        "platform": "linux-amd64",
        "rbac_enabled": true,
        "sfs30_url": "*****",
        "sfs_turbo_url": "*****",
        "sfs_url": "*****",
        "supportHcs": false,
        "swr_addr": "*****",

```

```
"swr_user" : "hwofficial"
},
"custom" : {
  "cluster_id" : "*****",
  "cluster_version" : "v1.23.8-r0",
  "csi_attacher_detach_worker_threads" : "60",
  "csi_attacher_worker_threads" : "60",
  "default_vpc_id" : "*****",
  "disable_auto_mount_secret" : false,
  "enable_node_attacher" : false,
  "flow_control" : { },
  "multiAZEnabled" : false,
  "over_subscription" : "80",
  "project_id" : "*****",
  "volume_attaching_flow_ctrl" : "0"
},
"flavor" : {
  "category" : [ "CCE", "Turbo" ],
  "description" : "Has only one instance",
  "name" : "Single",
  "replicas" : 1,
  "resources" : [ {
    "limitsCpu" : "250m",
    "limitsMem" : "600Mi",
    "name" : "everest-csi-controller",
    "requestsCpu" : "250m",
    "requestsMem" : "600Mi"
  }, {
    "limitsCpu" : "100m",
    "limitsMem" : "300Mi",
    "name" : "everest-csi-driver",
    "requestsCpu" : "100m",
    "requestsMem" : "300Mi"
  } ]
},
"systemAutoInject" : {
  "cluster" : {
    "clusterID" : "*****",
    "clusterNetworkMode" : "vpc-router",
    "clusterVersion" : "v1.23.8-r0"
  },
  "user" : {
    "projectID" : "*****"
  }
}
},
"status" : {
  "status" : "deleting",
  "message" : "",
  "targetVersions" : null,
  "isRollbackable" : false,
  "currentVersion" : {
    "version" : "2.1.30",
    "input" : {
      "basic" : {
        "bms_url" : "*****",
        "driver_init_image_version" : "2.1.30",
        "ecsEndpoint" : "*****",
        "everest_image_version" : "2.1.30",
        "evs_url" : "*****",
        "iam_url" : "*****",
        "ims_url" : "*****",
        "obs_url" : "*****",
        "platform" : "*****",
        "sfs30_url" : "*****",
        "sfs_turbo_url" : "*****",
        "sfs_url" : "*****",
        "supportHcs" : false,
```

```
"swr_addr" : "*****",
"swr_user" : "hwofficial"
},
"parameters" : {
  "common" : {
    "defaultVPCId" : 1234567
  },
  "custom" : {
    "cluster_id" : "",
    "csi_attacher_detach_worker_threads" : "60",
    "csi_attacher_worker_threads" : "60",
    "default_vpc_id" : "",
    "disable_auto_mount_secret" : false,
    "enable_node_attacher" : false,
    "flow_control" : { },
    "multiAZEnabled" : false,
    "over_subscription" : "80",
    "project_id" : "",
    "volume_attaching_flow_ctrl" : "0"
  },
  "flavor1" : {
    "description" : "High available",
    "name" : "HA",
    "replicas" : 2,
    "resources" : [ {
      "limitsCpu" : "250m",
      "limitsMem" : "1500Mi",
      "name" : "everest-csi-controller",
      "requestsCpu" : "250m",
      "requestsMem" : "600Mi"
    }, {
      "limitsCpu" : "500m",
      "limitsMem" : "300Mi",
      "name" : "everest-csi-driver",
      "requestsCpu" : "100m",
      "requestsMem" : "300Mi"
    } ]
  },
  "flavor2" : {
    "description" : "Has only one instance",
    "name" : "Single",
    "replicas" : 1,
    "resources" : [ {
      "limitsCpu" : "250m",
      "limitsMem" : "600Mi",
      "name" : "everest-csi-controller",
      "requestsCpu" : "250m",
      "requestsMem" : "600Mi"
    }, {
      "limitsCpu" : "100m",
      "limitsMem" : "300Mi",
      "name" : "everest-csi-driver",
      "requestsCpu" : "100m",
      "requestsMem" : "300Mi"
    } ]
  },
  "flavor3" : {
    "description" : "custom resources",
    "name" : "custom-resources",
    "replicas" : 2,
    "resources" : [ {
      "limitsCpu" : "250m",
      "limitsMem" : "2000Mi",
      "name" : "everest-csi-controller",
      "requestsCpu" : "250m",
      "requestsMem" : "1500Mi"
    }, {
      "limitsCpu" : "500m",
      "limitsMem" : "300Mi",

```

```
        "name" : "everest-csi-driver",
        "requestsCpu" : "100m",
        "requestsMem" : "300Mi"
    } ]
}
},
"stable" : true,
"translate" : {
    "en_US" : {
        "addon" : {
            "changeLog" : "",
            "description" : ""
        },
        "description" : {
            "Parameters.flavor1.description" : "Deploy the add-on with two instances, delivering high
availability but requiring more compute resources.",
            "Parameters.flavor1.name" : "HA",
            "Parameters.flavor2.description" : "Deploy the add-on with one instance.",
            "Parameters.flavor2.name" : "Standalone",
            "Parameters.flavor3.name" : "Custom"
        }
    },
    "fr_FR" : {
        "addon" : {
            "changeLog" : "",
            "description" : ""
        },
        "description" : {
            "Parameters.flavor1.description" : "Déployez avec deux instances, haute disponibilité.",
            "Parameters.flavor1.name" : "HA",
            "Parameters.flavor2.description" : "Déployez avec une seule instance.",
            "Parameters.flavor2.name" : "Célibataire",
            "Parameters.flavor3.name" : "Douane"
        }
    },
    "zh_CN" : {
        "addon" : {
            "changeLog" : "",
            "description" : ""
        },
        "description" : {
            "Parameters.flavor1.description" : "Dual-instance deployment features high availability but requires
more compute resources.",
            "Parameters.flavor1.name" : "High availability",
            "Parameters.flavor2.description" : "Single-instance deployment",
            "Parameters.flavor2.name" : "Single instance",
            "Parameters.flavor3.description" : "Custom resource specifications for deployment",
            "Parameters.flavor3.name" : "Custom"
        }
    }
},
"supportVersions" : null,
"creationTimestamp" : "2023-05-12T16:10:05Z",
"updateTimestamp" : "2023-05-12T16:10:05Z"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
```

```
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class DeleteAddonInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteAddonInstanceRequest request = new DeleteAddonInstanceRequest();
        request.withId("{id}");
        try {
            DeleteAddonInstanceResponse response = client.deleteAddonInstance(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
```

```
.with_credentials(credentials) \
.with_region(CceRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = DeleteAddonInstanceRequest()
    request.id = "{id}"
    response = client.delete_addon_instance(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteAddonInstanceRequest{}
    request.Id = "{id}"
    response, err := client.DeleteAddonInstance(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.6.6 Querying an Add-on Instance

Function

This API is used to obtain details about an add-on instance.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/addons/{id}

Table 4-916 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Add-on instance ID.

Table 4-917 Query Parameters

Parameter	Mandatory	Type	Description
cluster_id	No	String	Cluster ID (deprecated). For details about how to obtain the cluster ID, see How Do I Obtain Parameters in the API URI .

Request Parameters

Table 4-918 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-919 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.
spec	InstanceSpec object	Detailed description about the add-on instance.
status	AddonInstanceStatus object	Add-on instance status.

Table 4-920 AddonMetadata

Parameter	Type	Description
uid	String	Unique ID
name	String	Add-on name
alias	String	Add-on alias
labels	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	String	Update time.
creationTimestamp	String	Creation time.

Table 4-921 InstanceSpec

Parameter	Type	Description
clusterID	String	Cluster ID.

Parameter	Type	Description
version	String	Add-on template version, for example, 1.0.0 .
addonTemplateName	String	Add-on template name, for example, coredns .
addonTemplateType	String	Add-on template type.
addonTemplateLogo	String	URL for obtaining the add-on template logo.
addonTemplateLabels	Array of strings	Labels of the add-on template.
description	String	Add-on template description.
values	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). Set the parameters accordingly.

Table 4-922 AddonInstanceStatus

Parameter	Type	Description
status	String	Add-on instance status. Options: • running: All of the add-on instances are running. This specifies that the add-on runs properly. • abnormal: The add-on instances are abnormal and the add-on cannot be used. You can click the add-on name to view exceptions. • installing: The add-on is being installed. • installFailed: Installing the add-on failed. In this case, uninstall the add-on and then reinstall it. • upgrading: The add-on is being upgraded. • upgradeFailed: Upgrading the add-on failed. In this case, upgrade the add-on again or uninstall the add-on and reinstall it. • deleting: The add-on is being deleted. • deleteFailed: Deleting the add-on failed. In this case, uninstall the add-on again. • deleteSuccess: Deleting the add-on succeeded. • available: Only some instances of the add-on are running. This specifies that some functions of the add-on are available. • rollbacking: The add-on is being rolled back. • rollbackFailed: Rolling back the add-on failed. In this case, roll back the add-on again or uninstall the add-on and reinstall it. • unknown: The add-on chart instance does not exist.
Reason	String	Cause of the add-on installation failure.
message	String	Installation error details.
targetVersions	Array of strings	Cluster versions that support the add-on upgrade.

Parameter	Type	Description
currentVersion	Versions object	Information about the current add-on version.
isRollbackable	Boolean	Whether the add-on version can be rolled back to the source version.
previousVersion	String	The add-on version before upgrade or rollback

Table 4-923 Versions

Parameter	Type	Description
version	String	Add-on version.
input	Object	Add-on installation parameters.
stable	Boolean	Whether the add-on version is a stable release.
translate	Object	Translation information used by the GUI.
supportVersions	Array of SupportVersions objects	Cluster versions that support the add-on.
creationTimestamp	String	Creation time.
updateTimestamp	String	Update time.

Table 4-924 SupportVersions

Parameter	Type	Description
clusterType	String	Cluster type that supports the add-on.
clusterVersion	Array of strings	Cluster versions that support the add-on. The value is a regular expression.

Parameter	Type	Description
category	Array of strings	The cluster types to which the settings are applied Options: • CCE: CCE standard clusters • Turbo: CCE Turbo clusters • Autopilot: CCE Autopilot clusters Default value: If this parameter is left blank, CCE standard and Turbo clusters are used by default.

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "uid": "24b23108-55c0-11e9-926f-0255ac101a31",
    "name": "gpu-beta",
    "alias": "gpu",
    "creationTimestamp": "2019-04-03T03:25:34Z",
    "updateTimestamp": "2019-04-03T03:25:34Z"
  },
  "spec": {
    "clusterID": "0c0e4a63-5539-11e9-95f7-0255ac10177e",
    "version": "1.0.0",
    "addonTemplateName": "gpu-beta",
    "addonTemplateType": "helm",
    "addonTemplateLogo": "",
    "addonTemplateLabels": [ "Accelerator" ],
    "description": "A device plugin for nvidia.com/gpu resource on nvidia driver",
    "values": {
      "basic": {
        "rbac_enabled": true,
        "swr_addr": "100.125.6.246:20202",
        "swr_user": "hwofficial"
      }
    }
  },
  "status": {
    "status": "installing",
    "Reason": "",
    "message": "",
    "targetVersions": null,
    "currentVersion": {
      "version": "1.0.0",
      "input": {
        "basic": {
          "swr_addr": "100.125.6.246:20202",
```

```
    "swr_user" : "hwofficial"
  },
  "parameters" : { }
},
"stable" : true,
"translate" : {
  "en_US" : {
    "addon" : {
      "changeLog" : "A device plugin for nvidia.com/gpu resource on nvidia driver",
      "description" : "A device plugin for nvidia.com/gpu resource on nvidia driver"
    }
  },
  "zh_CN" : {
    "addon" : {
      "changeLog" : "",
      "description" : ""
    }
  }
},
"supportVersions" : null,
"creationTimestamp" : "2018-10-23T13:14:55Z",
"updateTimestamp" : "2018-12-07T09:40:24Z"
}
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowAddonInstanceSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowAddonInstanceRequest request = new ShowAddonInstanceRequest();
        request.withId("{id}");
        try {
            ShowAddonInstanceResponse response = client.showAddonInstance(request);
            System.out.println(response.toString());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowAddonInstanceRequest()
        request.id = "{id}"
        response = client.show_addon_instance(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```



```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowAddonInstanceRequest{}
request.Id = "{id}"
response, err := client.ShowAddonInstance(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.6.7 Listing Add-on Instances

Function

This API is used to list all add-on instances in the cluster.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/addons

Table 4-925 Query Parameters

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID. For details about how to obtain a cluster ID, see How to Obtain Parameters in the API URI .

Request Parameters

Table 4-926 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-927 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
items	Array of AddonInstance objects	Add-on instance list.

Table 4-928 AddonInstance

Parameter	Type	Description
kind	String	API type. The value is fixed at Addon and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	AddonMetadata object	Basic information about the object. Metadata is a collection of attributes.

Parameter	Type	Description
spec	InstanceSpec object	Detailed description about the add-on instance.
status	AddonInstanceStatus object	Add-on instance status.

Table 4-929 AddonMetadata

Parameter	Type	Description
uid	String	Unique ID
name	String	Add-on name
alias	String	Add-on alias
labels	Map<String,String>	Add-on labels in key-value pairs. This is a reserved field and does not take effect.
annotations	Map<String,String>	Add-on annotations in the format of key-value pairs. - For add-on installation, the value is fixed at {"addon.install/type":"install"}. - For add-on upgrade, the value is fixed at {"addon.upgrade/type":"upgrade"}.
updateTimestamp	String	Update time.
creationTimestamp	String	Creation time.

Table 4-930 InstanceSpec

Parameter	Type	Description
clusterID	String	Cluster ID.
version	String	Add-on template version, for example, 1.0.0 .
addonTemplateName	String	Add-on template name, for example, coredns .
addonTemplateType	String	Add-on template type.

Parameter	Type	Description
addonTemplateLogo	String	URL for obtaining the add-on template logo.
addonTemplateLabels	Array of strings	Labels of the add-on template.
description	String	Add-on template description.
values	Map<String,Object>	Add-on template installation parameters (varying depending on the add-on). Set the parameters accordingly.

Table 4-931 AddonInstanceStatus

Parameter	Type	Description
status	String	Add-on instance status. Options: • running: All of the add-on instances are running. This specifies that the add-on runs properly. • abnormal: The add-on instances are abnormal and the add-on cannot be used. You can click the add-on name to view exceptions. • installing: The add-on is being installed. • installFailed: Installing the add-on failed. In this case, uninstall the add-on and then reinstall it. • upgrading: The add-on is being upgraded. • upgradeFailed: Upgrading the add-on failed. In this case, upgrade the add-on again or uninstall the add-on and reinstall it. • deleting: The add-on is being deleted. • deleteFailed: Deleting the add-on failed. In this case, uninstall the add-on again. • deleteSuccess: Deleting the add-on succeeded. • available: Only some instances of the add-on are running. This specifies that some functions of the add-on are available. • rollbacking: The add-on is being rolled back. • rollbackFailed: Rolling back the add-on failed. In this case, roll back the add-on again or uninstall the add-on and reinstall it. • unknown: The add-on chart instance does not exist.
Reason	String	Cause of the add-on installation failure.
message	String	Installation error details.
targetVersions	Array of strings	Cluster versions that support the add-on upgrade.

Parameter	Type	Description
currentVersion	Versions object	Information about the current add-on version.
isRollbackable	Boolean	Whether the add-on version can be rolled back to the source version.
previousVersion	String	The add-on version before upgrade or rollback

Table 4-932 Versions

Parameter	Type	Description
version	String	Add-on version.
input	Object	Add-on installation parameters.
stable	Boolean	Whether the add-on version is a stable release.
translate	Object	Translation information used by the GUI.
supportVersions	Array of SupportVersions objects	Cluster versions that support the add-on.
creationTimestamp	String	Creation time.
updateTimestamp	String	Update time.

Table 4-933 SupportVersions

Parameter	Type	Description
clusterType	String	Cluster type that supports the add-on.
clusterVersion	Array of strings	Cluster versions that support the add-on. The value is a regular expression.

Parameter	Type	Description
category	Array of strings	The cluster types to which the settings are applied Options: • CCE: CCE standard clusters • Turbo: CCE Turbo clusters • Autopilot: CCE Autopilot clusters Default value: If this parameter is left blank, CCE standard and Turbo clusters are used by default.

Example Requests

None

Example Responses

Status code: 200

ok

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "items": [ {
    "kind": "Addon",
    "apiVersion": "v3",
    "metadata": {
      "uid": "8ca259cc-553b-11e9-926f-0255ac101a31",
      "name": "storage-driver",
      "alias": "storage-driver",
      "creationTimestamp": "2019-04-02T11:36:26Z",
      "updateTimestamp": "2019-04-02T11:36:26Z"
    },
    "spec": {
      "clusterID": "0c0e4a63-5539-11e9-95f7-0255ac10177e",
      "version": "1.0.10",
      "addonTemplateName": "storage-driver",
      "addonTemplateType": "helm",
      "addonTemplateLogo": "https://192.149.48.66/cce-addon-southchina-aw1hz2u/storage-driverlogo.svg",
      "addonTemplateLabels": [ "Storage" ],
      "description": "A kubernetes FlexVolume Driver used to support cloud storage",
      "values": {
        "basic": {
          "addon_version": "1.0.10",
          "euleros_version": "2.2.5",
          "obs_url": "",
          "platform": "linux-amd64",
          "swr_addr": "100.125.6.246:20202",
          "swr_user": "hwofficial"
        },
        "flavor": {
          "replicas": 1
        },
        "parameters": { }
      }
    }
  } ],
}
```



```
"status" : {
  "status" : "running",
  "Reason" : "Install complete",
  "message" : "",
  "targetVersions" : null,
  "currentVersion" : {
    "version" : "1.0.10",
    "input" : {
      "basic" : {
        "euleros_version" : "2.2.5",
        "obs_url" : "",
        "swr_addr" : "100.125.6.246:20202",
        "swr_user" : "hwofficial"
      },
      "parameters" : { }
    },
    "stable" : true,
    "translate" : {
      "en_US" : {
        "addon" : {
          "changeLog" : "The plug-in is upgraded to enhance the storage plug-in function.",
          "description" : "A kubernetes FlexVolume Driver used to support cloud storage"
        }
      },
      "zh_CN" : {
        "addon" : {
          "changeLog" : "",
          "description" : ""
        }
      }
    },
    "supportVersions" : null,
    "creationTimestamp" : "2019-03-29T13:45:37Z",
    "updateTimestamp" : "2019-03-29T13:45:37Z"
  }
}
}]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListAddonInstancesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```

```
ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ListAddonInstancesRequest request = new ListAddonInstancesRequest();
try {
    ListAddonInstancesResponse response = client.listAddonInstances(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAddonInstancesRequest()
        response = client.list_addon_instances(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```

```
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAddonInstancesRequest{}
    response, err := client.ListAddonInstances(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

4.7 Cluster Upgrade

4.7.1 Upgrading a Cluster

Function

Cluster upgrade

 NOTE

- Cluster upgrade involves many operations on components. You are advised to upgrade your cluster on the CCE console, a more interactive and intuitive way to reduce operational risks.
- Currently, cluster upgrade APIs are available only upon request.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade

Table 4-934 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-935 Request body parameters

Parameter	Mandatory	Type	Description
metadata	Yes	UpgradeClusterRequestMetadata object	Details: Cluster upgrade metadata information Constraints: None
spec	Yes	UpgradeSpec object	Details: Upgrade settings. CCE upgrades a cluster based on spec . Constraints: None

Table 4-936 UpgradeClusterRequestMetadata

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Details: API version Constraints: The value is fixed. Options: v3
kind	Yes	String	Details: API type Constraints: The value is fixed. Options: UpgradeTask

Table 4-937 UpgradeSpec

Parameter	Mandatory	Type	Description
clusterUpgradeAction	No	ClusterUpgradeAction object	Details: Detailed configuration information about the cluster upgrade Constraints: None

Table 4-938 ClusterUpgradeAction

Parameter	Mandatory	Type	Description
addons	No	Array of UpgradeAddonConfig objects	Details: Add-on configuration list. CCE upgrades add-ons based on the configuration during the cluster upgrade. Constraints: None
nodeOrder	No	Map<String,Array< NodePriority >>	Details: Upgrade sequence of nodes in a node pool. key indicates the node pool ID. The value for the default node pool is DefaultPool . Constraints: None
nodePoolOrder	No	Map<String,Integer>	Details: Upgrade sequence of a node pool, in key-value pairs. key indicates the node pool ID. The value for the default node pool is DefaultPool . value indicates the node pool priority. The default value is 0 , which indicates the lowest priority. A larger value indicates a higher priority. Constraints: None

Parameter	Mandatory	Type	Description
strategy	Yes	UpgradeStrategy object	Details: Cluster upgrade policy Constraints: None
targetVersion	Yes	String	Details: Target cluster version, for example, v1.23 Constraints: You can only upgrade a cluster to a later version. Do not enter a value that is equal to or earlier than the current cluster version. Options: Supported cluster versions

Table 4-939 UpgradeAddonConfig

Parameter	Mandatory	Type	Description
addonTemplateName	Yes	String	Details: CCE add-on name Constraints: None Options: Names of the add-ons installed in the cluster.
operation	Yes	String	Details: Action for upgrading an add-on Constraints: None Options: patch: specifies that the add-on version will be upgraded.

Parameter	Mandatory	Type	Description
version	Yes	String	Details: Target add-on version Constraints: The target add-on version must match the target cluster version. Options: N/A
values	No	Map<String,Object>	Details: Add-on parameter list, in key-value pairs Constraints: None

Table 4-940 NodePriority

Parameter	Mandatory	Type	Description
nodeSelector	Yes	NodeSelector object	Details: Node label selector, which selects a batch of nodes Constraints: Only labels on the nodes are allowed.
priority	Yes	Integer	Details: Priority of the current batch of nodes. A larger value indicates a higher priority. Constraints: None Options: Positive integers Default value: 0

Table 4-941 NodeSelector

Parameter	Mandatory	Type	Description
key	Yes	String	Details: Key Constraints: None Options: N/A
value	No	Array of strings	Details: Tag value list Constraints: None Options: N/A
operator	Yes	String	Details: Logical operators of labels Constraints: None Options: • in • notin • exists • ! • gt • lt

Table 4-942 UpgradeStrategy

Parameter	Mandatory	Type	Description
type	Yes	String	Details: Upgrade policy type Constraints: None Options: • inPlaceRollingUpdate: in-place upgrade

Parameter	Mandatory	Type	Description
inPlaceRollingUpdate	No	InPlaceRollingUpdate object	Details: Detailed configuration of an in-place upgrade policy Constraints: This parameter is mandatory when the in-place upgrade policy type is specified.

Table 4-943 InPlaceRollingUpdate

Parameter	Mandatory	Type	Description
userDefinedStep	No	Integer	Details: Maximum number of nodes to be upgraded in a batch. Node pools will be upgraded in sequence. Nodes in node pools will be upgraded in batches. One node is upgraded in the first batch, two nodes in the second batch, and the number of nodes to be upgraded in each subsequent batch increases by a power of 2 until the maximum number of nodes to be upgraded in each batch is reached. The next cluster is upgraded after the previous one is upgraded. Constraints: None Options: 1 to 60 Default value: 20

Parameter	Mandatory	Type	Description
scope	No	String	Details: Scope of the node upgrade batch Constraints: None Options: Cluster: If the scope is set to a cluster, the upgrade batch will remain unchanged throughout the entire upgrade process. NodePool: If the scope is set to node pools, the upgrade batch will be reset for each individual node pool. Default value: Cluster

Response Parameters

Status code: 200

Table 4-944 Response body parameters

Parameter	Type	Description
metadata	UpgradeClusterResponseMetadata object	Upgrade task metadata
spec	UpgradeResponseSpec object	Upgrade settings

Table 4-945 UpgradeClusterResponseMetadata

Parameter	Type	Description
uid	String	Upgrade task ID. You can obtain the progress by calling the API for obtaining cluster upgrade task details.

Table 4-946 UpgradeResponseSpec

Parameter	Type	Description
clusterUpgradeAction	ClusterUpgradeResponseAction object	Cluster upgrade settings

Table 4-947 ClusterUpgradeResponseAction

Parameter	Type	Description
version	String	Current cluster version
targetVersion	String	Target cluster version, for example, v1.23.
targetPlatformVersion	String	Platform version of the target cluster, which is an internal version of the cluster version and cannot be specified.
strategy	UpgradeStrategyResponse object	Upgrade policies.
config	Object	Cluster configuration specified during an upgrade.

Table 4-948 UpgradeStrategyResponse

Parameter	Type	Description
type	String	Upgrade policy type.
inPlaceRollingUpdate	InPlaceRollingUpdateResponse object	Detailed configuration of an in-place upgrade policy.

Table 4-949 InPlaceRollingUpdateResponse

Parameter	Type	Description
userDefinedStep	Integer	Maximum number of nodes to be upgraded in a batch.
scope	String	Scope of the node upgrade batch.

Example Requests

Upgrade the cluster to v1.23 and set the node upgrade step to 20.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade

{
  "metadata": {
    "apiVersion": "v3",
    "kind": "UpgradeTask"
  },
  "spec": {
    "clusterUpgradeAction": {
      "strategy": {
        "type": "inPlaceRollingUpdate",
        "inPlaceRollingUpdate": {
          "userDefinedStep": 20
        }
      }
    },
    "targetVersion": "v1.23"
  }
}
```

Example Responses

Status code: 200

Cluster upgrade requested.

```
{
  "metadata": {
    "uid": "976a33e2-f545-11ed-87af-0255ac1002c2"
  },
  "spec": {
    "clusterUpgradeAction": {
      "version": "v1.19.16-r20",
      "targetVersion": "v1.23.8-r0",
      "targetPlatformVersion": "cce.10",
      "strategy": {
        "type": "inPlaceRollingUpdate",
        "inPlaceRollingUpdate": {
          "userDefinedStep": 20
        }
      }
    },
    "config": { }
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Upgrade the cluster to v1.23 and set the node upgrade step to 20.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
```

```
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpgradeClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        UpgradeClusterRequest request = new UpgradeClusterRequest();
        request.withClusterId("{cluster_id}");
        UpgradeClusterRequestBody body = new UpgradeClusterRequestBody();
        InPlaceRollingUpdate inPlaceRollingUpdateStrategy = new InPlaceRollingUpdate();
        inPlaceRollingUpdateStrategy.withUserDefinedStep(20);
        UpgradeStrategy strategyClusterUpgradeAction = new UpgradeStrategy();
        strategyClusterUpgradeAction.withType("inPlaceRollingUpdate")
            .withInPlaceRollingUpdate(inPlaceRollingUpdateStrategy);
        ClusterUpgradeAction clusterUpgradeActionSpec = new ClusterUpgradeAction();
        clusterUpgradeActionSpec.withStrategy(strategyClusterUpgradeAction)
            .withTargetVersion("v1.23");
        UpgradeSpec specbody = new UpgradeSpec();
        specbody.withClusterUpgradeAction(clusterUpgradeActionSpec);
        UpgradeClusterRequestMetadata metadatabody = new UpgradeClusterRequestMetadata();
        metadatabody.withApiVersion("v3")
            .withKind("UpgradeTask");
        body.withSpec(specbody);
        body.withMetadata(metadatabody);
        request.withBody(body);
        try {
            UpgradeClusterResponse response = client.upgradeCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Upgrade the cluster to v1.23 and set the node upgrade step to 20.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
```

```
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpgradeClusterRequest()
        request.cluster_id = "{cluster_id}"
        inplaceRollingUpdateStrategy = InPlaceRollingUpdate(
            user_defined_step=20
        )
        strategyClusterUpgradeAction = UpgradeStrategy(
            type="inPlaceRollingUpdate",
            in_place_rolling_update=inplaceRollingUpdateStrategy
        )
        clusterUpgradeActionSpec = ClusterUpgradeAction(
            strategy=strategyClusterUpgradeAction,
            target_version="v1.23"
        )
        specbody = UpgradeSpec(
            cluster_upgrade_action=clusterUpgradeActionSpec
        )
        metabodbody = UpgradeClusterRequestMetadata(
            api_version="v3",
            kind="UpgradeTask"
        )
        request.body = UpgradeClusterRequestBody(
            spec=specbody,
            metadata=metabodbody
        )
        response = client.upgrade_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Upgrade the cluster to v1.23 and set the node upgrade step to 20.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")
projectId := "{project_id}"

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpgradeClusterRequest{}
request.ClusterId = "{cluster_id}"
userDefinedStepInPlaceRollingUpdate := int32(20)
inPlaceRollingUpdateStrategy := &model.InPlaceRollingUpdate{
    UserDefinedStep: &userDefinedStepInPlaceRollingUpdate,
}
strategyClusterUpgradeAction := &model.UpgradeStrategy{
    Type: "inPlaceRollingUpdate",
    InPlaceRollingUpdate: inPlaceRollingUpdateStrategy,
}
clusterUpgradeActionSpec := &model.ClusterUpgradeAction{
    Strategy: strategyClusterUpgradeAction,
    TargetVersion: "v1.23",
}
specbody := &model.UpgradeSpec{
    ClusterUpgradeAction: clusterUpgradeActionSpec,
}
metadatabody := &model.UpgradeClusterRequestMetadata{
    ApiVersion: "v3",
    Kind: "UpgradeTask",
}
request.Body = &model.UpgradeClusterRequestBody{
    Spec: specbody,
    Metadata: metadatabody,
}
response, err := client.UpgradeCluster(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Cluster upgrade requested.

Error Codes

See [Error Codes](#).

4.7.2 Obtaining Cluster Upgrade Task Details

Function

Upgrade task ID, obtained from the **uid** field in the response body after the cluster upgrade API is called.

 NOTE

- Cluster upgrade involves many operations on components. You are advised to upgrade your cluster on the CCE console, a more interactive and intuitive way to reduce operational risks.
- Currently, cluster upgrade APIs are available only upon request.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade/tasks/{task_id}

Table 4-950 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
task_id	Yes	String	Details: Upgrade task ID, obtained from the uid field in the response body after the cluster upgrade API is called. Constraints: None Options: Upgrade task IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-951 Response body parameters

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3 .
kind	String	Resource type, which defaults to UpgradeTask .
metadata	UpgradeTaskMetadata object	Upgrade task metadata.

Parameter	Type	Description
spec	UpgradeTaskSpec object	Upgrade task information.
status	UpgradeTaskStatus object	Upgrade task status.

Table 4-952 UpgradeTaskMetadata

Parameter	Type	Description
uid	String	Upgrade task ID.
creationTimestamp	String	Time when a task was created.
updateTimestamp	String	Time when a task is updated.

Table 4-953 UpgradeTaskSpec

Parameter	Type	Description
version	String	Source cluster version.
targetVersion	String	Target cluster version to upgrade to.
items	Object	Additional information about the upgrade task.

Table 4-954 UpgradeTaskStatus

Parameter	Type	Description
phase	String	Status of the upgrade task. Init: initializing NOTE Queuing: waitingPause: pausedFailed: failed • Running: The driver is running. • Success: successful.
progress	String	Upgrade task progress.
completionTime	String	End time of the upgrade task.

Example Requests

None

Example Responses

Status code: 200

Cluster upgrade task details obtained.

```
{
  "kind" : "UpgradeTask",
  "apiVersion" : "v3",
  "metadata" : {
    "uid" : "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "creationTimestamp" : "2022-12-16 13:40:20.75671 +0800 CST",
    "updateTimestamp" : "2022-12-16 13:40:20.756712 +0800 CST"
  },
  "spec" : {
    "version" : "v1.19.16-r4",
    "targetVersion" : "v1.23.5-r0"
  },
  "status" : {
    "phase" : "Init",
    "progress" : "0.00",
    "completionTime" : "2022-12-16 13:40:20.756712 +0800 CST"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowUpgradeClusterTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
```

```
        .withRegion(CceRegion.valueOf("<YOUR REGION>"))
        .build();
ShowUpgradeClusterTaskRequest request = new ShowUpgradeClusterTaskRequest();
request.withClusterId("{cluster_id}");
request.withTaskId("{task_id}");
try {
    ShowUpgradeClusterTaskResponse response = client.showUpgradeClusterTask(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.valueOf("<YOUR REGION>")) \
        .build()

    try:
        request = ShowUpgradeClusterTaskRequest()
        request.cluster_id = "{cluster_id}"
        request.task_id = "{task_id}"
        response = client.show_upgrade_cluster_task(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowUpgradeClusterTaskRequest{}
    request.ClusterId = "{cluster_id}"
    request.TaskId = "{task_id}"
    response, err := client.ShowUpgradeClusterTask(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Cluster upgrade task details obtained.

Error Codes

See [Error Codes](#).

4.7.3 Retrying a Cluster Upgrade Task

Function

Execute the failed cluster upgrade task again.

 NOTE

- Cluster upgrade involves many operations on components. You are advised to upgrade your cluster on the CCE console, a more interactive and intuitive way to reduce operational risks.
- Currently, cluster upgrade APIs are available only upon request.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade/retry

Table 4-955 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Cluster upgrade retry requested.

None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class RetryUpgradeClusterTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        RetryUpgradeClusterTaskRequest request = new RetryUpgradeClusterTaskRequest();
        request.withClusterId("{cluster_id}");
        try {
            RetryUpgradeClusterTaskResponse response = client.retryUpgradeClusterTask(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```



```
    } catch (ServiceResponseException e) {  
        e.printStackTrace();  
        System.out.println(e.getHttpStatusCode());  
        System.out.println(e.getRequestId());  
        System.out.println(e.getErrorCode());  
        System.out.println(e.getErrorMsg());  
    }  
}  
}
```

Python

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = CceClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(CceRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = RetryUpgradeClusterTaskRequest()  
        request.cluster_id = "{cluster_id}"  
        response = client.retry_upgrade_cluster_task(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.RetryUpgradeClusterTaskRequest{}
request.ClusterId = "{cluster_id}"
response, err := client.RetryUpgradeClusterTask(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Cluster upgrade retry requested.

Error Codes

See [Error Codes](#).

4.7.4 Suspending a Cluster Upgrade Task (Deprecated)

Function

Suspend the cluster upgrade task.

NOTE

- Cluster upgrade involves many operations on components. You are advised to upgrade your cluster on the CCE console, a more interactive and intuitive way to reduce operational risks.
- Currently, cluster upgrade APIs are available only upon request.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade/pause

Table 4-956 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Cluster upgrade suspension requested.

None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class PauseUpgradeClusterTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        PauseUpgradeClusterTaskRequest request = new PauseUpgradeClusterTaskRequest();
        request.withClusterId("{cluster_id}");
        try {
            PauseUpgradeClusterTaskResponse response = client.pauseUpgradeClusterTask(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = PauseUpgradeClusterTaskRequest()
        request.cluster_id = "{cluster_id}"
        response = client.pause_upgrade_cluster_task(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.PauseUpgradeClusterTaskRequest{}
    request.ClusterId = "{cluster_id}"
```

```
response, err := client.PauseUpgradeClusterTask(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Cluster upgrade suspension requested.

Error Codes

See [Error Codes](#).

4.7.5 Continuing to Execute a Cluster Upgrade Task (Deprecated)

Function

Resume the suspended cluster upgrade task.

NOTE

- Cluster upgrade involves many operations on components. You are advised to upgrade your cluster on the CCE console, a more interactive and intuitive way to reduce operational risks.
- Currently, cluster upgrade APIs are available only upon request.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade/continue

Table 4-957 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200
Cluster upgrade resumption requested.
None

Example Requests

None

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ContinueUpgradeClusterTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ContinueUpgradeClusterTaskRequest request = new ContinueUpgradeClusterTaskRequest();
        request.withClusterId("{cluster_id}");
        try {
            ContinueUpgradeClusterTaskResponse response = client.continueUpgradeClusterTask(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```



```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ContinueUpgradeClusterTaskRequest()
        request.cluster_id = "{cluster_id}"
        response = client.continue_upgrade_cluster_task(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ContinueUpgradeClusterTaskRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ContinueUpgradeClusterTask(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Cluster upgrade resumption requested.

Error Codes

See [Error Codes](#).

4.7.6 Obtaining a List of Cluster Upgrade Task Details

Function

Obtain a list of cluster upgrade task details.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade/tasks

Table 4-958 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-959 Response body parameters

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3
kind	String	Resource type
metadata	UpgradeTaskMetadata object	Metadata
items	Array of UpgradeTaskResponseBody objects	Cluster upgrade tasks

Table 4-960 UpgradeTaskResponseBody

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3 .
kind	String	Resource type, which defaults to UpgradeTask .

Parameter	Type	Description
metadata	UpgradeTaskMetadata object	Upgrade task metadata.
spec	UpgradeTaskSpec object	Upgrade task information.
status	UpgradeTaskStatus object	Upgrade task status.

Table 4-961 UpgradeTaskMetadata

Parameter	Type	Description
uid	String	Upgrade task ID.
creationTimestamp	String	Time when a task was created.
updateTimestamp	String	Time when a task is updated.

Table 4-962 UpgradeTaskSpec

Parameter	Type	Description
version	String	Source cluster version.
targetVersion	String	Target cluster version to upgrade to.
items	Object	Additional information about the upgrade task.

Table 4-963 UpgradeTaskStatus

Parameter	Type	Description
phase	String	Status of the upgrade task. Init: initializing NOTE Queuing: waitingPause: pausedFailed: failed • Running: The driver is running. • Success: successful.
progress	String	Upgrade task progress.
completionTime	String	End time of the upgrade task.

Example Requests

None

Example Responses

Status code: 200

The list of cluster upgrade task details is obtained.

```
{
  "kind" : "List",
  "apiVersion" : "v3",
  "metadata" : { },
  "items" : [ {
    "kind" : "UpgradeTask",
    "apiVersion" : "v3",
    "metadata" : {
      "uid" : "f40cafed-7bf1-4c3b-b619-80113b4bbb18",
      "creationTimestamp" : "2023-11-24 16:41:12.09236 +0800 CST",
      "updateTimestamp" : "2023-11-24 16:44:05.634206 +0800 CST"
    },
    "spec" : {
      "version" : "v1.17.17-r0",
      "targetVersion" : "v1.19.16-r80"
    },
    "status" : {
      "phase" : "Success",
      "completionTime" : "2023-11-24 16:44:05.634206 +0800 CST"
    }
  }, {
    "kind" : "UpgradeTask",
    "apiVersion" : "v3",
    "metadata" : {
      "uid" : "91755b96-5fd8-4a6a-bda1-983de9055996",
      "creationTimestamp" : "2023-11-24 19:54:35.194306 +0800 CST",
      "updateTimestamp" : "2023-11-24 20:14:35.194306 +0800 CST"
    },
    "spec" : {
      "version" : "v1.19.16-r80",
      "targetVersion" : "v1.23.8-r10"
    },
    "status" : {
      "phase" : "Success",
      "completionTime" : "2023-11-24 20:14:35.194306 +0800 CST"
    }
  }
]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;
```

```
public class ListUpgradeClusterTasksSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ListUpgradeClusterTasksRequest request = new ListUpgradeClusterTasksRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListUpgradeClusterTasksResponse response = client.listUpgradeClusterTasks(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
```

```
request = ListUpgradeClusterTasksRequest()
request.cluster_id = "{cluster_id}"
response = client.list_upgrade_cluster_tasks(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListUpgradeClusterTasksRequest{
        request.ClusterId = "{cluster_id}"
    }
    response, err := client.ListUpgradeClusterTasks(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The list of cluster upgrade task details is obtained.

Error Codes

See [Error Codes](#).

4.7.7 Performing a Pre-upgrade Check

Function

Pre-upgrade check

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/precheck

Table 4-964 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-965 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Details: API version Constraints: The value is fixed. Options: v3
kind	Yes	String	Details: API type Constraints: The value is fixed. Options: PreCheckTask
spec	Yes	PrecheckSpec object	Details: spec is an element type of the collection class. The main body of the configuration that needs to be checked before an upgrade is provided in spec . CCE checks the configuration based on spec . Constraints: None

Table 4-966 PrecheckSpec

Parameter	Mandatory	Type	Description
clusterID	Yes	String	Details: Cluster ID Constraints: None Options: N/A

Parameter	Mandatory	Type	Description
clusterVersion	Yes	String	Details: Cluster version. The value is the patch version of the current cluster. You can log in to the console and view the version on the Overview page. Constraints: None Options: N/A
targetVersion	Yes	String	Details: Target version. If you enter a major version, the latest patch version is automatically selected. Constraints: None Options: Available cluster versions later than the current cluster version
skippedCheckItemtemList	No	Array of skippedCheckItemtemList objects	Details: Skipped check items Constraints: None Options: N/A

Table 4-967 skippedCheckItemtemList

Parameter	Mandatory	Type	Description
name	No	String	Details: Name of the skipped checked item Constraints: None Options: N/A

Parameter	Mandatory	Type	Description
resourceSelector	No	resourceSelector object	Details: Resource tag selector Constraints: This parameter is available only for node check, but not for cluster check or add-on check. Options: N/A

Table 4-968 resourceSelector

Parameter	Mandatory	Type	Description
key	Yes	String	Details: Tag key Constraints: None Options: • node.uid : node UID
values	No	Array of strings	Details: Tag value list Constraints: None Options: N/A
operator	Yes	String	Details: Logical operators of labels Constraints: None Options: • In

Response Parameters

Status code: 200

Table 4-969 Response body parameters

Parameter	Type	Description
apiVersion	String	API version
kind	String	Resource type
metadata	PrecheckClusterResponseMetadata object	Pre-upgrade check metadata
spec	PrecheckClusterResponseSpec object	spec is an element type of the collection class. The main body of the configuration that needs to be checked before an upgrade is provided in spec . CCE checks the configuration based on the spec description.
status	PrecheckStatus object	Pre-upgrade check status of a cluster

Table 4-970 PrecheckClusterResponseMetadata

Parameter	Type	Description
uid	String	Check the task ID.

Table 4-971 PrecheckClusterResponseSpec

Parameter	Type	Description
clusterID	String	Cluster ID
clusterVersion	String	Current cluster version
targetVersion	String	Target version
skippedCheckItemList	Array of skippedCheckItemListResponse objects	Skipped check items

Table 4-972 skippedCheckItemListResponse

Parameter	Type	Description
name	String	Name of the item that is skipped to be checked

Parameter	Type	Description
resourceSelector	resourceSelector-Response object	Resource tag selector. This parameter is available only for node check, but not for cluster check or add-on check.

Table 4-973 resourceSelectorResponse

Parameter	Type	Description
key	String	Tag key
values	Array of strings	Tag value list
operator	String	Tag value

Table 4-974 PrecheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed • Error
expireTimeStamp	String	Check result expiration time
message	String	Information, which typically indicates the log for execution errors
clusterCheckStatus	clusterCheckStatus object	Status of a cluster restrictions check
addonCheckStatus	addonCheckStatus object	Status of an add-on check
nodeCheckStatus	nodeCheckStatus object	Node check status

Table 4-975 clusterCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-976 addonCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-977 nodeCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
nodeStageStatus	Array of NodeStageStatus objects	Node check status

Table 4-978 NodeStageStatus

Parameter	Type	Description
nodeInfo	NodeInfo object	Node information
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-979 NodeInfo

Parameter	Type	Description
uid	String	Node UID
name	String	Node name
status	String	Status
nodeType	String	Node type

Table 4-980 PreCheckItemStatus

Parameter	Type	Description
name	String	Check item
kind	String	Check item type. Options: • Exception: exceptions that need to be resolved by you • Risk: risks that can be skipped after you confirm them
group	String	Check item group. Options: • LimitCheck: cluster limit check • MasterCheck: master node check • NodeCheck: worker node check • AddonCheck: add-on check • ExecuteException: process error check
level	String	Check item severity. Options: • Info: information • Warning • Fatal: critical

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
message	String	Information
riskSource	riskSource object	Risk item
errorCodes	Array of strings	Error code set

Table 4-981 riskSource

Parameter	Type	Description
configurationRisks	Array of configurationRisks objects	Configuration risk
deprecatedAPIRisks	Array of deprecatedAPIRisks objects	Deprecated API risk
nodeRisks	Array of nodeRisks objects	Node risk
addonRisks	Array of addonRisks objects	Add-on risk

Table 4-982 configurationRisks

Parameter	Type	Description
package	String	Component
sourceFile	String	How to Obtain
nodeMsg	String	Node information
field	String	Parameter Value
operation	String	Change an operation type.
originalValue	String	Original Value
value	String	Current Value

Table 4-983 deprecatedAPIRisks

Parameter	Type	Description
url	String	Request path, for example, /apis/policy/v1beta1/podsecuritypolicies
userAgent	String	Client information

Table 4-984 nodeRisks

Parameter	Type	Description
NodeID	String	Worker node ID

Table 4-985 addonRisks

Parameter	Type	Description
addonTemplateName	String	Name of the add-on template
alias	String	Add-on alias

Example Requests

Request body for a cluster pre-upgrade check

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/precheck
```

```
{
  "kind": "PreCheckTask",
  "apiVersion": "v3",
  "spec": {
    "clusterID": "8978deaa-1743-11ee-8e46-0255ac10004c",
    "clusterVersion": "v1.15.11-r1",
    "targetVersion": "v1.19.16-r80",
    "skippedCheckItemList": [ ]
  }
}
```

Example Responses

Status code: 200

Cluster pre-upgrade check succeeded.

```
{
  "kind": "PreCheckTask",
  "apiVersion": "v3",
  "metadata": {
    "uid": "9991b45e-a2be-4b49-aca4-50a25fa6f81e"
  },
  "spec": {
    "clusterID": "8978deaa-1743-11ee-8e46-0255ac10004c",
```

```
{
  "clusterVersion" : "v1.15.11-r1",
  "targetVersion" : "v1.19.16-r80"
},
"status" : {
  "phase" : "Init",
  "clusterCheckStatus" : {
    "phase" : "Init"
  },
  "addonCheckStatus" : {
    "phase" : "Init"
  },
  "nodeCheckStatus" : {
    "phase" : "Init"
  }
}
```

Status Codes

Status Code	Description
200	Cluster pre-upgrade check succeeded.

Error Codes

See [Error Codes](#).

4.7.8 Obtaining Details About a Pre-upgrade Check Task of a Cluster

Function

Obtain details about a pre-upgrade check task of a cluster. The task ID is obtained from the **uid** field in the response body after the cluster check API is called.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/precheck/tasks/{task_id}

Table 4-986 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
task_id	Yes	String	Details: Upgrade task ID, obtained from the uid field in the response body after the cluster upgrade API is called. Constraints: None Options: Upgrade task IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-987 Response body parameters

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3
kind	String	Resource type, which defaults to PreCheckTask
metadata	PrecheckTaskMetadata object	Metadata of a pre-upgrade check task
spec	PrecheckClusterResponseSpec object	Information of a pre-upgrade check task
status	PrecheckStatus object	Status of a pre-upgrade check task

Table 4-988 PrecheckTaskMetadata

Parameter	Type	Description
uid	String	Task ID
creationTimestamp	String	Time when the task was created
updateTimestamp	String	Time when the task was updated.

Table 4-989 PrecheckClusterResponseSpec

Parameter	Type	Description
clusterID	String	Cluster ID
clusterVersion	String	Current cluster version
targetVersion	String	Target version
skippedCheckItemList	Array of skippedCheckItemListResponse objects	Skipped check items

Table 4-990 skippedCheckItemListResponse

Parameter	Type	Description
name	String	Name of the item that is skipped to be checked
resourceSelector	resourceSelectorResponse object	Resource tag selector. This parameter is available only for node check, but not for cluster check or add-on check.

Table 4-991 resourceSelectorResponse

Parameter	Type	Description
key	String	Tag key
values	Array of strings	Tag value list
operator	String	Tag value

Table 4-992 PrecheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed • Error
expireTimeStamp	String	Check result expiration time
message	String	Information, which typically indicates the log for execution errors
clusterCheckStatus	clusterCheckStatus object	Status of a cluster restrictions check
addonCheckStatus	addonCheckStatus object	Status of an add-on check
nodeCheckStatus	nodeCheckStatus object	Node check status

Table 4-993 clusterCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-994 addonCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-995 nodeCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
nodeStageStatus	Array of NodeStageStatus objects	Node check status

Table 4-996 NodeStageStatus

Parameter	Type	Description
nodeInfo	NodeInfo object	Node information
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-997 NodeInfo

Parameter	Type	Description
uid	String	Node UID
name	String	Node name
status	String	Status
nodeType	String	Node type

Table 4-998 PreCheckItemStatus

Parameter	Type	Description
name	String	Check item
kind	String	Check item type. Options: • Exception: exceptions that need to be resolved by you • Risk: risks that can be skipped after you confirm them
group	String	Check item group. Options: • LimitCheck: cluster limit check • MasterCheck: master node check • NodeCheck: worker node check • AddonCheck: add-on check • ExecuteException: process error check
level	String	Check item severity. Options: • Info: information • Warning • Fatal: critical

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
message	String	Information
riskSource	riskSource object	Risk item
errorCodes	Array of strings	Error code set

Table 4-999 riskSource

Parameter	Type	Description
configurationRisks	Array of configurationRisks objects	Configuration risk
deprecatedAPIRisks	Array of deprecatedAPIRisks objects	Deprecated API risk
nodeRisks	Array of nodeRisks objects	Node risk
addonRisks	Array of addonRisks objects	Add-on risk

Table 4-1000 configurationRisks

Parameter	Type	Description
package	String	Component
sourceFile	String	How to Obtain
nodeMsg	String	Node information
field	String	Parameter Value
operation	String	Change an operation type.
originalValue	String	Original Value
value	String	Current Value

Table 4-1001 deprecatedAPIRisks

Parameter	Type	Description
url	String	Request path, for example, /apis/policy/v1beta1/podsecuritypolicies
userAgent	String	Client information

Table 4-1002 nodeRisks

Parameter	Type	Description
NodeID	String	Worker node ID

Table 4-1003 addonRisks

Parameter	Type	Description
addonTemplateName	String	Name of the add-on template
alias	String	Add-on alias

Example Requests

None

Example Responses

Status code: 200

The details about a pre-upgrade check task of a cluster are obtained.

```
{
  "kind": "PreCheckTask",
  "apiVersion": "v3",
  "metadata": {
    "uid": "f61e008c-1600-41c0-9bde-121de5a30660",
    "creationTimestamp": "2023-11-25 07:20:04.592972 +0000 UTC",
    "updateTimestamp": "2023-11-25 07:21:05.518966 +0000 UTC"
  },
  "spec": {
    "clusterVersion": "v1.19.16-r4",
    "targetVersion": "v1.23.5-r0"
  },
  "status": {
    "phase": "Success",
    "expireTimestamp": "2023-11-25 08:21:05.518966 +0000 UTC",
    "clusterCheckStatus": {
      "phase": "Success",
      "itemsStatus": [ {
        "name": "DeprecatedApiCheck",
        "kind": "Risk",
        "group": "LimitCheck",
```

```
{
  "level": "Info",
  "phase": "Success",
  "message": "check item succeed",
  "riskSource": { }
}, {
  "name": "NodeContainerdPodRestartRisk",
  "kind": "Risk",
  "group": "LimitCheck",
  "level": "Warning",
  "phase": "Success",
  "message": "check item succeed",
  "riskSource": { }
}, {
  "name": "ResiduePackageVersion",
  "kind": "Exception",
  "group": "LimitCheck",
  "level": "Fatal",
  "phase": "Success",
  "message": "check item succeed",
  "riskSource": { }
}]
}, {
  "addonCheckStatus": {
    "phase": "Success",
    "itemsStatus": [ {
      "name": "AddonLimit",
      "kind": "Exception",
      "group": "AddonCheck",
      "level": "Warning",
      "phase": "Success",
      "message": "check item succeed",
      "riskSource": { }
    }, {
      "name": "CoreDNSConfLimit",
      "kind": "Exception",
      "group": "AddonCheck",
      "level": "Fatal",
      "phase": "Success",
      "message": "check item succeed",
      "riskSource": { }
    }
  ]
}, {
  "nodeCheckStatus": {
    "phase": "Success"
  }
}
}
```

Status Codes

Status Code	Description
200	The details about a pre-upgrade check task of a cluster are obtained.

Error Codes

See [Error Codes](#).

4.7.9 Obtaining a List of Pre-upgrade Check Tasks of a Cluster

Function

Obtain a list of pre-upgrade check tasks of a cluster.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/precheck/tasks

Table 4-1004 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-1005 Response body parameters

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3
kind	String	Type
metadata	Metadata object	Basic information about the object. Metadata is a collection of attributes.
items	Array of PrecheckCluster-Task objects	Cluster check tasks

Table 4-1006 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-1007 PrecheckClusterTask

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3
kind	String	Resource type, which defaults to PreCheckTask
metadata	PrecheckTaskMetadata object	Metadata of a pre-upgrade check task
spec	PrecheckClusterResponseSpec object	Information of a pre-upgrade check task
status	PrecheckStatus object	Status of a pre-upgrade check task

Table 4-1008 PrecheckTaskMetadata

Parameter	Type	Description
uid	String	Task ID
creationTimestamp	String	Time when the task was created
updateTimestamp	String	Time when the task was updated.

Table 4-1009 PrecheckClusterResponseSpec

Parameter	Type	Description
clusterID	String	Cluster ID
clusterVersion	String	Current cluster version
targetVersion	String	Target version
skippedCheckItemList	Array of skippedCheckItemResponse objects	Skipped check items

Table 4-1010 skippedCheckItemListResponse

Parameter	Type	Description
name	String	Name of the item that is skipped to be checked

Parameter	Type	Description
resourceSelector	resourceSelectorResponse object	Resource tag selector. This parameter is available only for node check, but not for cluster check or add-on check.

Table 4-1011 resourceSelectorResponse

Parameter	Type	Description
key	String	Tag key
values	Array of strings	Tag value list
operator	String	Tag value

Table 4-1012 PrecheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed • Error
expireTimeStamp	String	Check result expiration time
message	String	Information, which typically indicates the log for execution errors
clusterCheckStatus	clusterCheckStatus object	Status of a cluster restrictions check
addonCheckStatus	addonCheckStatus object	Status of an add-on check
nodeCheckStatus	nodeCheckStatus object	Node check status

Table 4-1013 clusterCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-1014 addonCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-1015 nodeCheckStatus

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
nodeStageStatus	Array of NodeStageStatus objects	Node check status

Table 4-1016 NodeStageStatus

Parameter	Type	Description
nodeInfo	NodeInfo object	Node information
itemsStatus	Array of PreCheckItemStatus objects	Compliance set by check item

Table 4-1017 NodeInfo

Parameter	Type	Description
uid	String	Node UID
name	String	Node name
status	String	Status
nodeType	String	Node type

Table 4-1018 PreCheckItemStatus

Parameter	Type	Description
name	String	Check item
kind	String	Check item type. Options: • Exception: exceptions that need to be resolved by you • Risk: risks that can be skipped after you confirm them
group	String	Check item group. Options: • LimitCheck: cluster limit check • MasterCheck: master node check • NodeCheck: worker node check • AddonCheck: add-on check • ExecuteException: process error check
level	String	Check item severity. Options: • Info: information • Warning • Fatal: critical

Parameter	Type	Description
phase	String	Status. Options: • Init: initializing • Running • Success • Failed
message	String	Information
riskSource	riskSource object	Risk item
errorCodes	Array of strings	Error code set

Table 4-1019 riskSource

Parameter	Type	Description
configurationRisks	Array of configurationRisks objects	Configuration risk
deprecatedAPIRisks	Array of deprecatedAPIRisks objects	Deprecated API risk
nodeRisks	Array of nodeRisks objects	Node risk
addonRisks	Array of addonRisks objects	Add-on risk

Table 4-1020 configurationRisks

Parameter	Type	Description
package	String	Component
sourceFile	String	How to Obtain
nodeMsg	String	Node information
field	String	Parameter Value
operation	String	Change an operation type.
originalValue	String	Original Value
value	String	Current Value

Table 4-1021 deprecatedAPIRisks

Parameter	Type	Description
url	String	Request path, for example, /apis/policy/v1beta1/podsecuritypolicies
userAgent	String	Client information

Table 4-1022 nodeRisks

Parameter	Type	Description
NodeID	String	Worker node ID

Table 4-1023 addonRisks

Parameter	Type	Description
addonTemplateName	String	Name of the add-on template
alias	String	Add-on alias

Example Requests

None

Example Responses

Status code: 200

The list of pre-upgrade check tasks of a cluster is obtained.

```
{
  "kind" : "List",
  "apiVersion" : "v3",
  "metadata" : { },
  "items" : [ {
    "kind" : "PreCheckTask",
    "apiVersion" : "v3",
    "metadata" : {
      "uid" : "10b52d23-080a-4b7d-bf83-64b4687ca786",
      "creationTimestamp" : "2023-12-16 07:07:11.099111 +0000 UTC",
      "updateTimestamp" : "2023-12-16 07:09:10.425622 +0000 UTC"
    },
    "spec" : {
      "clusterVersion" : "v1.23.5-r0",
      "targetVersion" : "v1.23.11-r0"
    },
    "status" : {
      "phase" : "Failed",
      "clusterCheckStatus" : {
        "phase" : "Success",
        "itemsStatus" : [ {
```

```
"name" : "DeprecatedApiCheck",
"kind" : "Risk",
"group" : "LimitCheck",
"level" : "Info",
"phase" : "Success",
"message" : "check item succeed",
"riskSource" : { }
}, {
  "name" : "BlackLimit",
  "kind" : "Exception",
  "group" : "LimitCheck",
  "level" : "Fatal",
  "phase" : "Success",
  "message" : "check item succeed",
  "riskSource" : { }
}, {
  "name" : "MasterSSH",
  "kind" : "Exception",
  "group" : "LimitCheck",
  "level" : "Fatal",
  "phase" : "Success",
  "message" : "check item succeed",
  "riskSource" : { }
}, {
  "name" : "ReleaseLimit",
  "kind" : "Exception",
  "group" : "LimitCheck",
  "level" : "Warning",
  "phase" : "Success",
  "message" : "check item succeed",
  "riskSource" : { }
}, {
  "name" : "ClusterNoArm",
  "kind" : "Exception",
  "group" : "LimitCheck",
  "level" : "Warning",
  "phase" : "Success",
  "message" : "check item succeed",
  "riskSource" : { }
}
}],
"addonCheckStatus" : {
  "phase" : "Failed",
  "itemsStatus" : [ {
    "name" : "AddonLimit",
    "kind" : "Exception",
    "group" : "AddonCheck",
    "level" : "Warning",
    "phase" : "Failed",
    "message" : "addon [ CoreDNS,CCE Container Storage (Everest) ] status is abnormal, check and try again",
    "riskSource" : {
      "addonRisks" : [ {
        "addonTemplateName" : "coredns",
        "alias" : "CoreDNS"
      }, {
        "addonTemplateName" : "everest",
        "alias" : "CCE Container Storage (Everest)"
      } ]
    }
  } ],
  "riskSource" : { }
}, {
  "name" : "CoreDNSConfLimit",
  "kind" : "Exception",
  "group" : "AddonCheck",
  "level" : "Fatal",
  "phase" : "Success",
  "message" : "check item succeed",
  "riskSource" : { }
}, {
```

```
{
  "name": "EverestLimitHungVersion",
  "kind": "Risk",
  "group": "AddonCheck",
  "level": "Fatal",
  "phase": "Success",
  "message": "check item succeed",
  "riskSource": { }
}]
},
"nodeCheckStatus": {
  "phase": "Success"
}
}
}]
}
```

Status Codes

Status Code	Description
200	The list of pre-upgrade check tasks of a cluster is obtained.

Error Codes

See [Error Codes](#).

4.7.10 Performing a Post-upgrade Check

Function

Post-upgrade check, which is used by the customer to check the cluster status after an upgrade and provide feedback after services are restored. Use this API with the console.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/postcheck

Table 4-1024 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1025 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	Details: API version Constraints: The value is fixed. Options: v3

Parameter	Mandatory	Type	Description
kind	Yes	String	Details: API type Constraints: The value is fixed. Options: PostCheckTask
spec	Yes	PostcheckResponseSpec object	spec indicates the configuration confirmed after an upgrade.

Table 4-1026 PostcheckResponseSpec

Parameter	Mandatory	Type	Description
clusterID	No	String	Cluster ID
clusterVersion	No	String	Source cluster version
targetVersion	No	String	Current cluster version

Response Parameters

Status code: 200

Table 4-1027 Response body parameters

Parameter	Type	Description
apiVersion	String	API version
kind	String	Resource type
metadata	PostcheckClusterResponseMetadata object	Post-upgrade check metadata
spec	PostcheckSpec object	Configuration confirmed after a cluster upgrade
status	status object	Status confirmed after a cluster upgrade

Table 4-1028 PostcheckCluserResponseMetadata

Parameter	Type	Description
uid	String	Task ID

Table 4-1029 PostcheckSpec

Parameter	Type	Description
clusterID	String	Details: Cluster ID Constraints: None Options: N/A
clusterVersion	String	Details: Source cluster version Constraints: None Options: N/A
targetVersion	String	Details: Current cluster version Constraints: None Options: Supported cluster versions

Table 4-1030 status

Parameter	Type	Description
phase	String	Status. Options: • Success • Failed • Error

Example Requests

Post-upgrade check


```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/postcheck

{
  "kind": "PostCheckTask",
  "apiVersion": "v3",
  "spec": {
    "clusterID": "8978deaa-1743-11ee-8e46-0255ac10004c",
    "clusterVersion": "v1.15.11-r1",
    "targetVersion": "v1.19.16-r80"
  }
}
```

Example Responses

Status code: 200

Cluster post-upgrade check succeeded.

```
{
  "kind": "PostCheckTask",
  "apiVersion": "v3",
  "metadata": {
    "uid": "e99fedf8-348c-4084-b0fd-81bf187df4e0"
  },
  "spec": {
    "clusterID": "8978deaa-1743-11ee-8e46-0255ac10004c",
    "clusterVersion": "v1.15.11-r1",
    "targetVersion": "v1.19.16-r80"
  },
  "status": {
    "phase": "Success"
  }
}
```

Status Codes

Status Code	Description
200	Cluster post-upgrade check succeeded.

Error Codes

See [Error Codes](#).

4.7.11 Backing Up a Cluster

Function

Cluster backup

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3.1/projects/{project_id}/clusters/{cluster_id}/operation/snapshot

Table 4-1031 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-1032 Response body parameters

Parameter	Type	Description
uid	String	Task ID
metadata	SnapshotClusterResponseMetadata object	Backup metadata

Table 4-1033 SnapshotCluserResponseMetadata

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3.1
kind	String	Task type

Example Requests

Example of a cluster upgrade backup request

POST /api/v3.1/projects/{project_id}/clusters/{cluster_id}/operation/snapshot

Example Responses

Status code: 200

The cluster backup task is created.

```
{
  "uid" : "15376f1b-daa6-4e2d-96a6-e9d5d7caeea2",
  "metadata" : {
    "kind" : "Snapshot",
    "apiVersion" : "v3.1"
  }
}
```

Status Codes

Status Code	Description
200	The cluster backup task is created.

Error Codes

See [Error Codes](#).

4.7.12 Obtaining a List of Cluster Backup Task Details

Function

Obtain a list of cluster backup task details.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3.1/projects/{project_id}/clusters/{cluster_id}/operation/snapshot/tasks

Table 4-1034 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200**Table 4-1035** Response body parameters

Parameter	Type	Description
apiVersion	String	API version, which defaults to v3.1
kind	String	Task type
metadata	SnapshotTaskMetadata object	Backup metadata

Parameter	Type	Description
items	Array of SnapshotTask objects	Backup tasks
status	SnapshotTaskStatus object	Backup task status

Table 4-1036 SnapshotTask

Parameter	Type	Description
kind	String	Task type
apiVersion	String	API version
metadata	SnapshotTaskMetadata object	Backup metadata
spec	SnapshotSpec object	Backing Up Task Configurations (To Be Deprecated)
status	SnapshotStatus object	Backup task status

Table 4-1037 SnapshotTaskMetadata

Parameter	Type	Description
uid	String	Task ID
creationTimestamp	String	Time when the task was created
updateTimestamp	String	Time when the task was updated.

Table 4-1038 SnapshotSpec

Parameter	Type	Description
items	Array of SnapshotSpecItems objects	Backup task details

Table 4-1039 SnapshotSpecItems

Parameter	Type	Description
id	String	Subtask ID
type	String	Subtask type
status	String	Status
creationTimestamp	String	Time when the task was created
updateTimestamp	String	Time when the task was updated.
message	String	Information

Table 4-1040 SnapshotStatus

Parameter	Type	Description
phase	String	Task status
progress	String	Task progress
completionTime	String	Completion time

Table 4-1041 SnapshotTaskStatus

Parameter	Type	Description
latestBackupTime	String	Time of the latest backup

Example Requests

None

Example Responses

Status code: 200

The list of cluster backup task details is obtained.

```
{
  "kind" : "List",
  "apiVersion" : "v3.1",
  "metadata" : { },
  "items" : [ {
    "kind" : "SnapshotTask",
    "apiVersion" : "v3.1",
    "metadata" : {
      "uid" : "87d326f9-46b0-486e-a4ba-1f82ec9315ed",
      "creationTimestamp" : "2023-11-25 17:03:46.739012 +0800 CST",
      "updateTimestamp" : "2023-11-25 17:03:46.739027 +0800 CST"
    }
  }
]
```

```
    },
    "spec" : { },
    "status" : {
      "phase" : "Running",
      "progress" : "67",
      "completionTime" : "2023-11-25 17:03:46.739027 +0800 CST"
    }
  },
  "status" : {
    "latestBackupTime" : "2023-11-25 17:03:47.980844 +0800 CST"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListClusterMasterSnapshotTasksSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ListClusterMasterSnapshotTasksRequest request = new ListClusterMasterSnapshotTasksRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListClusterMasterSnapshotTasksResponse response = client.listClusterMasterSnapshotTasks(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}
```

Python

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = CceClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(CceRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ListClusterMasterSnapshotTasksRequest()  
        request.cluster_id = "{cluster_id}"  
        response = client.list_cluster_master_snapshot_tasks(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()
```



```
client := cce.NewCceClient(  
    cce.CceClientBuilder().  
        WithRegion(region.ValueOf("<YOUR REGION>")).  
        WithCredential(auth).  
        Build())  
  
request := &model.ListClusterMasterSnapshotTasksRequest{  
    request.ClusterId = "{cluster_id}"  
}  
response, err := client.ListClusterMasterSnapshotTasks(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The list of cluster backup task details is obtained.

Error Codes

See [Error Codes](#).

4.7.13 Obtaining the Cluster Upgrade Information

Function

Obtain the cluster upgrade information.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/upgradeinfo

Table 4-1042 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

None

Response Parameters

Status code: 200**Table 4-1043** Response body parameters

Parameter	Type	Description
kind	String	Type
apiVersion	String	API version
metadata	Metadata object	Basic information about the object. Metadata is a collection of attributes.

Parameter	Type	Description
spec	UpgradeInfoSpec object	Upgrade settings
status	UpgradeInfoStatus object	Upgrade status

Table 4-1044 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-1045 UpgradeInfoSpec

Parameter	Type	Description
lastUpgradeInfo	UpgradeInfoStatus object	Last cluster upgrade
versionInfo	UpgradeVersionInfo object	Version
upgradeFeatureGates	UpgradeFeatureGates object	Cluster upgrade feature gates

Table 4-1046 UpgradeVersionInfo

Parameter	Type	Description
release	String	Official version, for example, v1.19.10
patch	String	Patch version, for example, r0
suggestPatch	String	Recommended target patch version, for example, r0
targetVersions	Array of strings	Target versions for the upgrade

Table 4-1047 UpgradeFeatureGates

Parameter	Type	Description
supportUpgradePageV4	Boolean	Whether the cluster upgrade console supports v4. This field is used by the CCE console.

Table 4-1048 UpgradeInfoStatus

Parameter	Type	Description
phase	String	Status of the upgrade task • Init: initializing • Running • Pause • Success • Failed
progress	String	Upgrade task progress
completionTime	String	End time of the upgrade task

Example Requests

None

Example Responses

Status code: 200

The cluster upgrade information is obtained.

```
{
  "kind": "UpgradeInfo",
  "apiVersion": "v3",
  "metadata": { },
  "spec": {
```

```
"lastUpgradeInfo" : {
  "phase" : "Success",
  "completionTime" : "2023-11-25 11:18:54.478926 +0800 CST"
},
"versionInfo" : {
  "release" : "v1.27.2",
  "patch" : "r0",
  "suggestPatch" : "r0",
  "targetVersions" : [ "v1.27.3-r0" ]
},
"upgradeFeatureGates" : {
  "supportUpgradePageV4" : true
}
},
"status" : {
  "phase" : "Success",
  "completionTime" : "2023-11-25 11:18:54.478926 +0800 CST"
}
}
```

Status Codes

Status Code	Description
200	The cluster upgrade information is obtained.

Error Codes

See [Error Codes](#).

4.7.14 Obtaining a Cluster Upgrade Path

Function

Obtain the cluster upgrade path.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/clusterupgradepaths

Request Parameters

Table 4-1049 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1050 Response body parameters

Parameter	Type	Description
apiVersion	String	API version
kind	String	Resource type
metadata	Metadata object	Basic information about the object. Metadata is a collection of attributes.
upgradePaths	Array of UpgradePath objects	Upgrade paths

Table 4-1051 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-1052 UpgradePath

Parameter	Type	Description
version	String	Cluster version. For example, v1.19.16-r20 for clusters v1.19 or earlier, and v1.21 for clusters v1.21 or later. For details, see CCE Cluster Versions.
platformVersion	String	CCE cluster platform version, which is an internal version under the cluster version (version). Platform versions are used to trace iterations in a major cluster version. They are unique within a major cluster version and recounted when the major cluster version changes.The format of platformVersion is cce.X.Y.- X : an internal feature version. When features or OSs in a cluster version are modified or patches are released in the cluster, the value of this parameter increases monotonically from 1.- Y : a patch version of an internal feature version. This parameter is only used for updating a software package for a released feature version. The value of this parameter increases monotonically from 0.
targetVersions	Array of strings	Target versions that can be upgraded

Example Requests

None

Example Responses

Status code: 200

The cluster upgrade path is obtained.

```
{
  "kind": "ClusterUpgradePaths",
  "apiVersion": "v3",
  "metadata": { },
  "upgradePaths": [ {
    "version": "v1.25",
    "platformVersion": "cce.5.0",
    "targetVersions": [ "v1.25.6-r0", "v1.27.3-r0" ]
  }, {
    "version": "v1.25",
    "platformVersion": "cce.4.0",
    "targetVersions": [ "v1.25.6-r0", "v1.27.3-r0" ]
  }, {
    "version": "v1.23",
    "platformVersion": "cce.10.0",
    "targetVersions": [ "v1.23.6-r0", "v1.25.6-r0" ]
  } ]
}
```

```
"targetVersions" : [ "v1.23.11-r0", "v1.25.6-r0", "v1.27.3-r0" ]  
}, {  
  "version" : "v1.23",  
  "platformVersion" : "cce.9.0",  
  "targetVersions" : [ "v1.23.11-r0", "v1.25.6-r0", "v1.27.3-r0" ]  
}]  
}
```

Status Codes

Status Code	Description
200	The cluster upgrade path is obtained.

Error Codes

See [Error Codes](#).

4.7.15 Obtaining the Configuration of Cluster Upgrade Feature Gates

Function

Obtain the configuration of cluster upgrade feature gates.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/clusterupgradefeaturegates

Request Parameters

Table 4-1053 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1054 Response body parameters

Parameter	Type	Description
apiVersion	String	API version
kind	String	Resource type
metadata	Metadata object	Basic information about the object. Metadata is a collection of attributes.
upgradeFeatureGates	Map<String,String>	Feature gates information, which is in the format of a key-value pair. • Key: can be DisplayPreCheckDetail (displaying details about all cluster check items before an upgrade), EvsSnapshot (using EVS snapshots to back up clusters), LabelForSkippedNode (labeling the nodes skipped during a cluster upgrade), or UpgradeStrategy (specifying cluster upgrade policies) • Value: can be Support, Disable, or Default, which is determined using default CCE rules.

Table 4-1055 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

The cluster upgrade path is obtained.

```
{
  "kind": "ClusterUpgradeFeatureGates",
  "apiVersion": "v3",
  "metadata": { },
  "upgradeFeatureGates": {
    "DisplayPreCheckDetail": "Support",
    "EvsSnapshot": "Support",
    "LabelForSkippedNode": "Support",
    "UpgradeStrategy": "Support"
  }
}
```

Status Codes

Status Code	Description
200	The cluster upgrade path is obtained.

Error Codes

See [Error Codes](#).

4.7.16 Enabling the Cluster Upgrade Process Booting Task

Function

This API is used to create a cluster upgrade booting task. After the booting task is created by calling this API, start the check through the cluster pre-upgrade check.

The upgrade process task is used to control the execution process of the cluster upgrade task. The execution process is as follows: Pre-upgrade check -> Cluster upgrade -> Post-upgrade check.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgradeworkflows

Table 4-1056 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1057 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1058 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	Details: API type Constraints: The value is fixed. Options: WorkflowTask
apiVersion	Yes	String	Details: API version Constraints: The value is fixed. Options: v3
spec	Yes	WorkflowSpec object	Details: Element type of the collection class. The main body of the cluster upgrade process is provided in spec . CCE creates or updates objects by defining or updating spec . Constraints: None

Table 4-1059 WorkflowSpec

Parameter	Mandatory	Type	Description
clusterID	Yes	String	Details: Cluster ID Constraints: None Options: N/A
clusterVersion	No	String	Details: Current cluster version Constraints: None Options: N/A

Parameter	Mandatory	Type	Description
targetVersion	Yes	String	Details: Target cluster version Constraints: None Options: N/A

Response Parameters

Status code: 201

Table 4-1060 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at WorkFlowTask and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	Metadata object	Upgrade process metadata
spec	WorkflowResponseSpec object	Element type of the collection class. The main body of the cluster upgrade process is provided in spec . CCE creates or updates objects by defining or updating spec .
status	WorkflowStatus object	Element type of the collection class, which is used to record the status of the cluster upgrade process, including the execution status of each process in the cluster upgrade process.

Table 4-1061 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A
labels	Map<String,String >	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String >	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-1062 WorkFlowResponseSpec

Parameter	Type	Description
clusterID	String	Cluster ID
clusterVersion	String	Current cluster version
targetVersion	String	Target cluster version

Table 4-1063 WorkFlowStatus

Parameter	Type	Description
phase	String	Execution status of the cluster upgrade process. Options: Init: No task is running in the upgrade process. Running: Some tasks in the upgrade process are being executed. Pending: A task in the upgrade process fails to be executed. Success: All tasks in the upgrade process have been executed. Cancel: The upgrade process has been canceled.
pointStatuses	Array of PointStatus objects	Execution status of each task in the upgrade process
lineStatuses	Array of LineStatus objects	Task execution path of the upgrade process

Table 4-1064 PointStatus

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster: a cluster upgrade task PreCheck: a cluster pre-upgrade check task Rollback: a cluster rollback task Snapshot: a cluster upgrade snapshot task PostCheck: a cluster post-upgrade check task
taskId	String	Upgrade task ID
status	String	Cluster upgrade status. Options: Init: The task is in the initial state. Queuing: The task is in the execution queue. Running: The task is being executed. Success: The task has been executed. Failed: Executing the task failed.

Parameter	Type	Description
startTimeStamp	String	Start time of the upgrade task
endTimeStamp	String	End time of the upgrade task
expireTimeStamp	String	Expiration time of the upgrade task (only for pre-upgrade check tasks)

Table 4-1065 LineStatus

Parameter	Type	Description
startPoint	Point object	Path start point
endPoint	Point object	Path end point
critical	String	Whether it is a key path. An upgrade can be canceled only after the key path is executed.

Table 4-1066 Point

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster : a cluster upgrade task PreCheck : a cluster pre-upgrade check task Rollback : a cluster rollback task Snapshot : a cluster upgrade snapshot task PostCheck : a cluster post-upgrade check task

Example Requests

Enable the process of upgrading a cluster to v1.23.

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgradeworkflows
{
  "kind": "WorkFlowTask",
  "apiVersion": "v3",
  "spec": {
    "targetVersion": "v1.23",
    "clusterID": "b4b9e60f-8aa2-11ee-af09-0255ac10004f"
  }
}
```

Example Responses

Status code: 201

The upgrade process is created in the specified cluster.

```
{
  "kind" : "WorkflowTask",
  "apiVersion" : "v3",
  "metadata" : {
    "uid" : "5ddfdffe-87db-11ec-b5e5-0255ac111914"
  },
  "spec" : {
    "clusterID" : "b4b9e60f-8aa2-11ee-af09-0255ac10004f",
    "clusterVersion" : "v1.17.17-r0",
    "targetVersion" : "v1.19.16-r80"
  },
  "status" : {
    "pointStatuses" : [ {
      "taskType" : "PreCheck"
    }, {
      "taskType" : "Snapshot"
    }, {
      "taskType" : "Cluster"
    }, {
      "taskType" : "PostCheck"
    } ],
    "lineStatuses" : [ {
      "startPoint" : {
        "taskType" : "PreCheck"
      },
      "endPoint" : {
        "taskType" : "Cluster"
      }
    }, {
      "startPoint" : {
        "taskType" : "Cluster"
      },
      "endPoint" : {
        "taskType" : "PostCheck"
      }
    } ]
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Enable the process of upgrading a cluster to v1.23.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class CreateUpgradeWorkflowSolution {
```



```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");
    String projectId = "{project_id}";

    ICredential auth = new BasicCredentials()
        .withProjectId(projectId)
        .withAk(ak)
        .withSk(sk);

    CceClient client = CceClient.newBuilder()
        .withCredential(auth)
        .withRegion(CceRegion.valueOf("<YOUR REGION>"))
        .build();
    CreateUpgradeWorkFlowRequest request = new CreateUpgradeWorkFlowRequest();
    request.withClusterId("{cluster_id}");
    CreateUpgradeWorkFlowRequestBody body = new CreateUpgradeWorkFlowRequestBody();
    WorkFlowSpec specbody = new WorkFlowSpec();
    specbody.withClusterID("b4b9e60f-8aa2-11ee-af09-0255ac10004f")
        .withTargetVersion("v1.23");
    body.withSpec(specbody);
    body.withApiVersion("v3");
    body.withKind("WorkFlowTask");
    request.withBody(body);
    try {
        CreateUpgradeWorkFlowResponse response = client.createUpgradeWorkFlow(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

Enable the process of upgrading a cluster to v1.23.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)
```

```
client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateUpgradeWorkFlowRequest()
    request.cluster_id = "{cluster_id}"
    specbody = WorkFlowSpec(
        cluster_id="b4b9e60f-8aa2-11ee-af09-0255ac10004f",
        target_version="v1.23"
    )
    request.body = CreateUpgradeWorkFlowRequestBody(
        spec=specbody,
        api_version="v3",
        kind="WorkFlowTask"
    )
    response = client.create_upgrade_work_flow(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Enable the process of upgrading a cluster to v1.23.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateUpgradeWorkFlowRequest{
        request.ClusterId = "{cluster_id}"
        clusterIDSpec := "b4b9e60f-8aa2-11ee-af09-0255ac10004f"
        specbody := &model.WorkFlowSpec{
            ClusterID: &clusterIDSpec,
            TargetVersion: "v1.23",
        }
    }
```

```
request.Body = &model.CreateUpgradeWorkFlowRequestBody{
    Spec: specbody,
    ApiVersion: "v3",
    Kind: "WorkFlowTask",
}
response, err := client.CreateUpgradeWorkFlow(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	The upgrade process is created in the specified cluster.

Error Codes

See [Error Codes](#).

4.7.17 Obtaining a List of Upgrade Workflows

Function

Obtain a list of historical cluster upgrade booting tasks.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgradeworkflows

Table 4-1067 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1068 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1069 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at List and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
items	UpgradeWorkflow object	Upgrade workflow list

Table 4-1070 UpgradeWorkflow

Parameter	Type	Description
kind	String	API type. The value is fixed at WorkflowTask and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	Metadata object	Upgrade process metadata
spec	WorkflowResponseSpec object	Element type of the collection class. The main body of the cluster upgrade process is provided in spec . CCE creates or updates objects by defining or updating spec .
status	WorkflowStatus object	Element type of the collection class, which is used to record the status of the cluster upgrade process, including the execution status of each process in the cluster upgrade process.

Table 4-1071 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-1072 WorkflowResponseSpec

Parameter	Type	Description
clusterID	String	Cluster ID
clusterVersion	String	Current cluster version
targetVersion	String	Target cluster version

Table 4-1073 WorkflowStatus

Parameter	Type	Description
phase	String	Execution status of the cluster upgrade process. Options: Init: No task is running in the upgrade process. Running: Some tasks in the upgrade process are being executed. Pending: A task in the upgrade process fails to be executed. Success: All tasks in the upgrade process have been executed. Cancel: The upgrade process has been canceled.
pointStatuses	Array of PointStatus objects	Execution status of each task in the upgrade process
lineStatuses	Array of LineStatus objects	Task execution path of the upgrade process

Table 4-1074 PointStatus

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster : a cluster upgrade task PreCheck : a cluster pre-upgrade check task Rollback : a cluster rollback task Snapshot : a cluster upgrade snapshot task PostCheck : a cluster post-upgrade check task
taskID	String	Upgrade task ID
status	String	Cluster upgrade status. Options: Init : The task is in the initial state. Queuing : The task is in the execution queue. Running : The task is being executed. Success : The task has been executed. Failed : Executing the task failed.
startTimeStamp	String	Start time of the upgrade task
endTimeStamp	String	End time of the upgrade task
expireTimeStamp	String	Expiration time of the upgrade task (only for pre-upgrade check tasks)

Table 4-1075 LineStatus

Parameter	Type	Description
startPoint	Point object	Path start point
endPoint	Point object	Path end point
critical	String	Whether it is a key path. An upgrade can be canceled only after the key path is executed.

Table 4-1076 Point

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster : a cluster upgrade task PreCheck : a cluster pre-upgrade check task Rollback : a cluster rollback task Snapshot : a cluster upgrade snapshot task PostCheck : a cluster post-upgrade check task

Example Requests

None

Example Responses

Status code: 200

The list of historical cluster upgrade booting tasks is obtained.

```
{
  "apiVersion" : "v3",
  "kind" : "List",
  "items" : {
    "kind" : "WorkFlowTask",
    "apiVersion" : "v3",
    "metadata" : {
      "uid" : "730f5577-38ef-448c-b4a7-c6878fbefdda",
      "creationTimestamp" : "2023-11-24 08:39:15.894417 +0000 UTC",
      "updateTimestamp" : "2023-11-25 02:57:25.718567 +0000 UTC"
    },
    "spec" : {
      "clusterID" : "b4b9e60f-8aa2-11ee-af09-0255ac10004f",
      "clusterVersion" : "v1.17.17-r0",
      "targetVersion" : "v1.19.16-r80"
    },
    "status" : {
      "phase" : "Cancel",
      "pointStatuses" : [ {
        "taskType" : "PreCheck"
      }, {
        "taskType" : "Snapshot"
      }, {
        "taskType" : "Cluster"
      }, {
        "taskType" : "PostCheck"
      } ],
      "lineStatuses" : [ {
        "startPoint" : {
          "taskType" : "PreCheck"
        },
        "endPoint" : {
          "taskType" : "Cluster"
        }
      }, {
        "startPoint" : {
```

```
        "taskType" : "Cluster"
      },
      "endPoint" : {
        "taskType" : "PostCheck"
      }
    }
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListUpgradeWorkFlowsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ListUpgradeWorkFlowsRequest request = new ListUpgradeWorkFlowsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListUpgradeWorkFlowsResponse response = client.listUpgradeWorkFlows(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListUpgradeWorkFlowsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_upgrade_work_flows(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
```

```
WithCredential(auth).  
Build())  
  
request := &model.ListUpgradeWorkFlowsRequest{  
    request.ClusterId = "{cluster_id}"  
    response, err := client.ListUpgradeWorkFlows(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The list of historical cluster upgrade booting tasks is obtained.

Error Codes

See [Error Codes](#).

4.7.18 Obtaining Details About a Specified Cluster Upgrade Task

Function

This API is used to obtain details about an upgrade booting task based on the task ID.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgradeworkflows/{upgrade_workflow_id}

Table 4-1077 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
upgrade_workflow_id	Yes	String	Bootimg process ID of a cluster upgrade task. For details about how to obtain the ID, see Obtaining API URI Parameters .

Request Parameters

Table 4-1078 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1079 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at WorkflowTask and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	Metadata object	Upgrade process metadata
spec	WorkflowResponseSpec object	Element type of the collection class. The main body of the cluster upgrade process is provided in spec . CCE creates or updates objects by defining or updating spec .
status	WorkflowStatus object	Element type of the collection class, which is used to record the status of the cluster upgrade process, including the execution status of each process in the cluster upgrade process.

Table 4-1080 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-1081 WorkflowResponseSpec

Parameter	Type	Description
clusterID	String	Cluster ID
clusterVersion	String	Current cluster version
targetVersion	String	Target cluster version

Table 4-1082 WorkflowStatus

Parameter	Type	Description
phase	String	Execution status of the cluster upgrade process. Options: Init : No task is running in the upgrade process. Running : Some tasks in the upgrade process are being executed. Pending : A task in the upgrade process fails to be executed. Success : All tasks in the upgrade process have been executed. Cancel : The upgrade process has been canceled.
pointStatuses	Array of PointStatus objects	Execution status of each task in the upgrade process
lineStatuses	Array of LineStatus objects	Task execution path of the upgrade process

Table 4-1083 PointStatus

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster : a cluster upgrade task PreCheck : a cluster pre-upgrade check task Rollback : a cluster rollback task Snapshot : a cluster upgrade snapshot task PostCheck : a cluster post-upgrade check task

Parameter	Type	Description
taskID	String	Upgrade task ID
status	String	Cluster upgrade status. Options: Init : The task is in the initial state. Queuing : The task is in the execution queue. Running : The task is being executed. Success : The task has been executed. Failed : Executing the task failed.
startTimeStamp	String	Start time of the upgrade task
endTimeStamp	String	End time of the upgrade task
expireTimeStamp	String	Expiration time of the upgrade task (only for pre-upgrade check tasks)

Table 4-1084 LineStatus

Parameter	Type	Description
startPoint	Point object	Path start point
endPoint	Point object	Path end point
critical	String	Whether it is a key path. An upgrade can be canceled only after the key path is executed.

Table 4-1085 Point

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster : a cluster upgrade task PreCheck : a cluster pre-upgrade check task Rollback : a cluster rollback task Snapshot : a cluster upgrade snapshot task PostCheck : a cluster post-upgrade check task

Example Requests

None

Example Responses

Status code: 200

Details about the cluster upgrade task are obtained.

```
{
  "kind": "WorkFlowTask",
  "apiVersion": "v3",
  "metadata": {
    "uid": "c271e39e-1a6e-4d3d-8fa8-2a36329c68d1",
    "creationTimestamp": "2023-11-25 06:32:34.923248 +0000 UTC",
    "updateTimestamp": "2023-11-25 07:49:30.281911 +0000 UTC"
  },
  "spec": {
    "clusterID": "b4b9e60f-8aa2-11ee-af09-0255ac10004f",
    "clusterVersion": "v1.17.17-r0",
    "targetVersion": "v1.19.16-r80"
  },
  "status": {
    "phase": "Pending",
    "pointStatuses": [ {
      "taskType": "PreCheck",
      "taskID": "f61e008c-1600-41c0-9bde-121de5a30660",
      "status": "Success",
      "startTimeStamp": "2023-11-25 07:20:04.592972 +0000 UTC",
      "endTimeStamp": "2023-11-25 07:21:05.518966 +0000 UTC",
      "expireTimeStamp": "2023-11-25 08:21:05.518966 +0000 UTC"
    }, {
      "taskType": "Snapshot"
    }, {
      "taskType": "Cluster",
      "taskID": "6d799ff6-3afe-4242-80b4-6f0a0fa746cb",
      "status": "Failed",
      "startTimeStamp": "2023-11-25 07:49:30.283459 +0000 UTC",
      "endTimeStamp": "2023-11-25 07:58:35.507243 +0000 UTC"
    }, {
      "taskType": "PostCheck"
    } ],
    "lineStatuses": [ {
      "startPoint": {
        "taskType": "PreCheck"
      },
      "endPoint": {
        "taskType": "Cluster"
      }
    }, {
      "startPoint": {
        "taskType": "Cluster"
      },
      "endPoint": {
        "taskType": "PostCheck"
      }
    } ]
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowUpgradeWorkFlowSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowUpgradeWorkFlowRequest request = new ShowUpgradeWorkFlowRequest();
        request.withClusterId("{cluster_id}");
        request.withUpgradeWorkflowId("{upgrade_workflow_id}");
        try {
            ShowUpgradeWorkFlowResponse response = client.showUpgradeWorkFlow(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
```

```
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ShowUpgradeWorkFlowRequest()
    request.cluster_id = "{cluster_id}"
    request.upgrade_workflow_id = "{upgrade_workflow_id}"
    response = client.show_upgrade_work_flow(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowUpgradeWorkFlowRequest{}
    request.ClusterId = "{cluster_id}"
    request.UpgradeWorkflowId = "{upgrade_workflow_id}"
    response, err := client.ShowUpgradeWorkFlow(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Details about the cluster upgrade task are obtained.

Error Codes

See [Error Codes](#).

4.7.19 Updating the Status of a Specified Cluster Upgrade Booting Task

Function

This API is used to update the status of a specified cluster upgrade booting task. This API applies only to the upgrade cancellation process.

When this API is called, the status of the upgrade process booting task cannot be **running**, **success**, or **cancel**, and the status of the upgrade subtask cannot be **running**, **init**, **pause**, or **queue**.

Calling Method

For details, see [Calling APIs](#).

URI

PATCH /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgradeworkflows/{upgrade_workflow_id}

Table 4-1086 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
upgrade_workflow_id	Yes	String	Bootimg process ID of a cluster upgrade task. For details about how to obtain the ID, see Obtaining API URI Parameters .

Request Parameters

Table 4-1087 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1088 Request body parameters

Parameter	Mandatory	Type	Description
status	No	status object	Details: Updated workflow status (Only Cancel is supported.) Constraints: None

Table 4-1089 status

Parameter	Mandatory	Type	Description
phase	No	String	Details: Execution status of the cluster upgrade process Constraints: None Options: • Cancel: The upgrade is canceled.

Response Parameters

Status code: 200

Table 4-1090 Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at WorkFlowTask and cannot be changed.
apiVersion	String	API version. The value is fixed at v3 and cannot be changed.
metadata	Metadata object	Upgrade process metadata
spec	WorkFlowResponseSpec object	Element type of the collection class. The main body of the cluster upgrade process is provided in spec . CCE creates or updates objects by defining or updating spec .

Parameter	Type	Description
status	WorkflowStatus object	Element type of the collection class, which is used to record the status of the cluster upgrade process, including the execution status of each process in the cluster upgrade process.

Table 4-1091 Metadata

Parameter	Type	Description
uid	String	Definition Unique ID Constraints N/A Range N/A Default Value N/A
name	String	Definition Resource name Constraints N/A Range N/A Default Value N/A
labels	Map<String,String>	Definition Resource labels in key-value pairs. This is a reserved field. Inputting values will not work. Constraints N/A Range N/A Default Value N/A

Parameter	Type	Description
annotations	Map<String,String>	Definition Resource annotations in the format of key-value pairs Constraints N/A Range N/A Default Value N/A
updateTimestamp	String	Definition Update time Constraints N/A Range N/A Default Value N/A
creationTimestamp	String	Definition Creation time Constraints N/A Range N/A Default Value N/A

Table 4-1092 WorkflowResponseSpec

Parameter	Type	Description
clusterID	String	Cluster ID
clusterVersion	String	Current cluster version
targetVersion	String	Target cluster version

Table 4-1093 WorkFlowStatus

Parameter	Type	Description
phase	String	Execution status of the cluster upgrade process. Options: Init: No task is running in the upgrade process. Running: Some tasks in the upgrade process are being executed. Pending: A task in the upgrade process fails to be executed. Success: All tasks in the upgrade process have been executed. Cancel: The upgrade process has been canceled.
pointStatuses	Array of PointStatus objects	Execution status of each task in the upgrade process
lineStatuses	Array of LineStatus objects	Task execution path of the upgrade process

Table 4-1094 PointStatus

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster: a cluster upgrade task PreCheck: a cluster pre-upgrade check task Rollback: a cluster rollback task Snapshot: a cluster upgrade snapshot task PostCheck: a cluster post-upgrade check task
taskId	String	Upgrade task ID
status	String	Cluster upgrade status. Options: Init: The task is in the initial state. Queuing: The task is in the execution queue. Running: The task is being executed. Success: The task has been executed. Failed: Executing the task failed.

Parameter	Type	Description
startTimeStamp	String	Start time of the upgrade task
endTimeStamp	String	End time of the upgrade task
expireTimeStamp	String	Expiration time of the upgrade task (only for pre-upgrade check tasks)

Table 4-1095 LineStatus

Parameter	Type	Description
startPoint	Point object	Path start point
endPoint	Point object	Path end point
critical	String	Whether it is a key path. An upgrade can be canceled only after the key path is executed.

Table 4-1096 Point

Parameter	Type	Description
taskType	String	Cluster upgrade task type. Options: Cluster : a cluster upgrade task PreCheck : a cluster pre-upgrade check task Rollback : a cluster rollback task Snapshot : a cluster upgrade snapshot task PostCheck : a cluster post-upgrade check task

Example Requests

Cancel the upgrade process.

```
PATCH /api/v3/projects/47eb1d64cbeb45cfa01ae20af4f4b563/clusters/
f9960c6b-8e60-11ee-9754-0255ac100b05/operation/upgradeworkflows/
d0b7e319-8172-424c-86ea-543cd23f9756

{
  "status" : {
    "phase" : "Cancel"
  }
}
```

Example Responses

Status code: 200

The status of the cluster upgrade booting task is updated.

```
{
  "kind": "WorkFlowTask",
  "apiVersion": "v3",
  "metadata": {
    "uid": "c271e39e-1a6e-4d3d-8fa8-2a36329c68d1",
    "creationTimestamp": "2023-11-25 06:32:34.923248 +0000 UTC",
    "updateTimestamp": "2023-11-25 07:49:30.281911 +0000 UTC"
  },
  "spec": {
    "clusterID": "b4b9e60f-8aa2-11ee-af09-0255ac10004f",
    "clusterVersion": "v1.17.17-r0",
    "targetVersion": "v1.19.16-r80"
  },
  "status": {
    "phase": "Cancel",
    "pointStatuses": [ {
      "taskType": "PreCheck",
      "taskID": "f61e008c-1600-41c0-9bde-121de5a30660",
      "status": "Success",
      "startTimeStamp": "2023-11-25 07:20:04.592972 +0000 UTC",
      "endTimeStamp": "2023-11-25 07:21:05.518966 +0000 UTC",
      "expireTimeStamp": "2023-11-25 08:21:05.518966 +0000 UTC"
    }, {
      "taskType": "Snapshot"
    }, {
      "taskType": "Cluster",
      "taskID": "6d799ff6-3afe-4242-80b4-6f0a0fa746cb",
      "status": "Failed",
      "startTimeStamp": "2023-11-25 07:49:30.283459 +0000 UTC",
      "endTimeStamp": "2023-11-25 07:58:35.507243 +0000 UTC"
    }, {
      "taskType": "PostCheck"
    } ],
    "lineStatuses": [ {
      "startPoint": {
        "taskType": "PreCheck"
      },
      "endPoint": {
        "taskType": "Cluster"
      }
    }, {
      "startPoint": {
        "taskType": "Cluster"
      },
      "endPoint": {
        "taskType": "PostCheck"
      }
    } ]
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Cancel the upgrade process.

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpgradeWorkFlowUpdateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR_REGION>"))
            .build();

        UpgradeWorkFlowUpdateRequest request = new UpgradeWorkFlowUpdateRequest();
        request.withClusterId("{cluster_id}");
        request.withUpgradeWorkflowId("{upgrade_workflow_id}");
        UpgradeWorkFlowUpdateRequestBody body = new UpgradeWorkFlowUpdateRequestBody();
        UpgradeWorkFlowUpdateRequestBodyStatus statusbody = new
        UpgradeWorkFlowUpdateRequestBodyStatus();
        statusbody.withPhase(UpgradeWorkFlowUpdateRequestBodyStatus.PhaseEnum.fromValue("Cancel"));
        body.withStatus(statusbody);
        request.withBody(body);
        try {
            UpgradeWorkFlowUpdateResponse response = client.upgradeWorkFlowUpdate(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

Cancel the upgrade process.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *
```



```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpgradeWorkFlowUpdateRequest()
        request.cluster_id = "{cluster_id}"
        request.upgrade_workflow_id = "{upgrade_workflow_id}"
        statusbody = UpgradeWorkFlowUpdateRequestBodyStatus(
            phase="Cancel"
        )
        request.body = UpgradeWorkFlowUpdateRequestBody(
            status=statusbody
        )
        response = client.upgrade_work_flow_update(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Cancel the upgrade process.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

```
Build()  
  
request := &model.UpgradeWorkFlowUpdateRequest{  
    request.ClusterId = "{cluster_id}"  
    request.UpgradeWorkflowId = "{upgrade_workflow_id}"  
    phaseStatus:= model.GetUpgradeWorkFlowUpdateRequestBodyStatusPhaseEnum().CANCEL  
    statusbody := &model.UpgradeWorkFlowUpdateRequestBodyStatus{  
        Phase: &phaseStatus,  
    }  
    request.Body = &model.UpgradeWorkFlowUpdateRequestBody{  
        Status: statusbody,  
    }  
    response, err := client.UpgradeWorkFlowUpdate(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The status of the cluster upgrade booting task is updated.

Error Codes

See [Error Codes](#).

4.8 Quota Management

4.8.1 Querying Resource Quotas

Function

This API is used to query CCE resource quotas.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/quotas

Table 4-1097 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Request Parameters

Table 4-1098 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1099 Response body parameters

Parameter	Type	Description
quotas	Array of QuotaResource objects	Resources

Table 4-1100 QuotaResource

Parameter	Type	Description
quotaKey	String	Details: Resource type Constraints: None Options: cluster: cluster quota Default value: N/A
quotaLimit	Integer	Quota

Parameter	Type	Description
used	Integer	Number of created resources
unit	String	Unit
regionId	String	Region ID. This parameter is not returned if not involved.
availabilityZoneId	String	AZ ID. This parameter is not returned if not involved.

Example Requests

None

Example Responses

Status code: 200

The resource quotas are obtained successfully.

```
{
  "quotas": [ {
    "quotaKey": "cluster",
    "quotaLimit": 20,
    "used": 13,
    "unit": "count"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowQuotasSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";
```

```
ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ShowQuotasRequest request = new ShowQuotasRequest();
try {
    ShowQuotasResponse response = client.showQuotas(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowQuotasRequest()
        response = client.show_quotas(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
```

```
"fmt"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowQuotasRequest{}
    response, err := client.ShowQuotas(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The resource quotas are obtained successfully.

Error Codes

See [Error Codes](#).

4.9 API Versions

4.9.1 Obtaining API Versions

Function

This API is used to query the list of API versions supported by CCE.

Calling Method

For details, see [Calling APIs](#).

URI

GET /

Request Parameters

None

Response Parameters

Status code: 200

Table 4-1101 Response body parameters

Parameter	Type	Description
versions	Array of APIVersionDetail objects	API version list.

Table 4-1102 APIVersionDetail

Parameter	Type	Description
id	String	API version ID. Example: v3
links	Array of APIVersionLink objects	URL of the API version.
min_version	String	Microversion. If the APIs of this version support microversions, set this parameter to the supported minimum microversion. If no microversion is supported, leave this parameter blank.

Parameter	Type	Description
status	String	API version status. Possible values: • CURRENT: preferred API version • SUPPORTED: old API version that is still supported • DEPRECATED: deprecated API version that will be deleted
updated	String	Time when the API was released (UTC). For example, if the API version is v3, the value is '2018-09-15 00:00:00Z'.
version	String	Microversion. If the APIs of this version support microversions, set this parameter to the supported maximum microversion. If no microversion is supported, leave this parameter blank.

Table 4-1103 APIVersionLink

Parameter	Type	Description
href	String	URL of the API version.
rel	String	Link attributes. self : A self link contains a versioned link to the resource. Use these links immediately after linking.

Example Requests

None

Example Responses

Status code: 200

The API version list is queried successfully.

```
{
  "versions": [ {
    "id": "v3",
    "links": [ {
      "href": "https://cce.region.***.com/v3",
      "rel": "self"
    } ],
    "min_version": "",
    "status": "CURRENT",
    "updated": "2018-09-15 00:00:00Z",
    "version": ""
  } ],
  "min_version": "",
  "status": "CURRENT",
  "updated": "2018-09-15 00:00:00Z",
  "version": ""
}
```

```
    }  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
public class ShowVersionSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        CceClient client = CceClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ShowVersionRequest request = new ShowVersionRequest();  
        try {  
            ShowVersionResponse response = client.showVersion(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *
```

```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowVersionRequest()
        response = client.show_version(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowVersionRequest{}
    response, err := client.ShowVersion(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The API version list is queried successfully.

Error Codes

See [Error Codes](#).

4.10 Tag Management

4.10.1 Adding Resource Tags to a Specified Cluster in Batches

Function

This API is used to add resource tags for a specified cluster in batches.

NOTE

- Each cluster supports a maximum of 20 resource tags.
- This API is idempotent. If the to-be-added tag has the same tag key and tag value as an existing tag, the tag will be added. If the to-be-added tag has the same key but different value as an existing tag, the tag will overwrite the existing one.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/tags/create

Table 4-1104 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1105 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1106 Request body parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of ResourceTag objects	List of cluster resource tags to be created. Each cluster supports a maximum of 20 resource tags.

Table 4-1107 ResourceTag

Parameter	Mandatory	Type	Description
key	No	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: <code>_.:=+@</code> • The value cannot start with <code>_sys_</code>. Default value: N/A

Parameter	Mandatory	Type	Description
value	No	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: _./=+-@ Default value: N/A

Response Parameters

Status code: 204

No Content

None

Example Requests

Adding Resource Tags to a Specified Cluster in Batches

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/tags/create
{
  "tags": [ {
    "key": "key1",
    "value": "value1"
  }, {
    "key": "key2",
    "value": "value3"
  } ]
}
```

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

Adding Resource Tags to a Specified Cluster in Batches

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchCreateClusterTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        BatchCreateClusterTagsRequest request = new BatchCreateClusterTagsRequest();
        request.withClusterId("{cluster_id}");
        BatchCreateClusterTagsRequestBody body = new BatchCreateClusterTagsRequestBody();
        List<ResourceTag> listbodyTags = new ArrayList<>();
        listbodyTags.add(
            new ResourceTag()
                .withKey("key1")
                .withValue("value1")
        );
        listbodyTags.add(
            new ResourceTag()
                .withKey("key2")
                .withValue("value3")
        );
        body.withTags(listbodyTags);
        request.withBody(body);
        try {
            BatchCreateClusterTagsResponse response = client.batchCreateClusterTags(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}
```

Python

Adding Resource Tags to a Specified Cluster in Batches

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
    projectId = "{project_id}"  
  
    credentials = BasicCredentials(ak, sk, projectId)  
  
    client = CceClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(CceRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = BatchCreateClusterTagsRequest()  
        request.cluster_id = "{cluster_id}"  
        listTagsbody = [  
            ResourceTag(  
                key="key1",  
                value="value1"  
            ),  
            ResourceTag(  
                key="key2",  
                value="value3"  
            )  
        ]  
        request.body = BatchCreateClusterTagsRequestBody(  
            tags=listTagsbody  
        )  
        response = client.batch_create_cluster_tags(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

Adding Resource Tags to a Specified Cluster in Batches

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"  
)
```

```
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := cce.NewCceClient(  
        cce.CceClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.BatchCreateClusterTagsRequest{}  
    request.ClusterId = "{cluster_id}"  
    keyTags := "key1"  
    valueTags := "value1"  
    keyTags1 := "key2"  
    valueTags1 := "value3"  
    var listTagsbody = []model.ResourceTag{  
        {  
            Key: &keyTags,  
            Value: &valueTags,  
        },  
        {  
            Key: &keyTags1,  
            Value: &valueTags1,  
        },  
    }  
    request.Body = &model.BatchCreateClusterTagsRequestBody{  
        Tags: listTagsbody,  
    }  
    response, err := client.BatchCreateClusterTags(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
204	No Content

Error Codes

See [Error Codes](#).

4.10.2 Deleting Resource Tags of a Specified Cluster in Batches

Function

This API is used to delete resource tags of a specified cluster in batches.

NOTE

- This API is idempotent. If the tag key of the to-be-deleted tag does not exist, the tag will be deleted.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/projects/{project_id}/clusters/{cluster_id}/tags/delete

Table 4-1108 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1109 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1110 Request body parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of ResourceDeleteTag objects	List of cluster resource tags to be deleted

Table 4-1111 ResourceDeleteTag

Parameter	Mandatory	Type	Description
key	No	String	The key value of the resource tag - It cannot be null. It can contain a maximum of 128 characters. - It can contain Chinese characters, letters, digits, and spaces in UTF-8 format. - It can contain the following special characters: <code>._:/=+@.</code> - It cannot start with <code>_sys_</code>.

Response Parameters

Status code: 204

No Content

None

Example Requests

Deleting Resource Tags of a Specified Cluster in Batches

```
POST /api/v3/projects/{project_id}/clusters/{cluster_id}/tags/delete
{
  "tags" : [ {
    "key" : "key1"
  }, {
    "key" : "key2"
  } ]
}
```

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

Deleting Resource Tags of a Specified Cluster in Batches

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchDeleteClusterTagsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);
```

```
CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
BatchDeleteClusterTagsRequest request = new BatchDeleteClusterTagsRequest();
request.withClusterId("{cluster_id}");
BatchDeleteClusterTagsRequestBody body = new BatchDeleteClusterTagsRequestBody();
List<ResourceDeleteTag> listbodyTags = new ArrayList<>();
listbodyTags.add(
    new ResourceDeleteTag()
        .withKey("key1")
);
listbodyTags.add(
    new ResourceDeleteTag()
        .withKey("key2")
);
body.withTags(listbodyTags);
request.withBody(body);
try {
    BatchDeleteClusterTagsResponse response = client.batchDeleteClusterTags(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

Deleting Resource Tags of a Specified Cluster in Batches

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchDeleteClusterTagsRequest()
        request.cluster_id = "{cluster_id}"
        listTagsbody = [
```



```
        ResourceDeleteTag(  
            key="key1"  
        ),  
        ResourceDeleteTag(  
            key="key2"  
        )  
    ]  
    request.body = BatchDeleteClusterTagsRequestBody(  
        tags=listTagsbody  
    )  
    response = client.batch_delete_cluster_tags(request)  
    print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

Deleting Resource Tags of a Specified Cluster in Batches

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
    projectId := "{project_id}"  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
  
    client := cce.NewCceClient(  
        cce.CceClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.BatchDeleteClusterTagsRequest{}  
    request.ClusterId = "{cluster_id}"  
    keyTags := "key1"  
    keyTags1 := "key2"  
    var listTagsbody = []model.ResourceDeleteTag{  
        {  
            Key: &keyTags,  
        },  
        {  
            Key: &keyTags1,  
        },  
    }  
    request.Body = &model.BatchDeleteClusterTagsRequestBody{  
        Tags: listTagsbody,  
    }  
}
```

```
response, err := client.BatchDeleteClusterTags(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
204	No Content

Error Codes

See [Error Codes](#).

4.10.3 Obtaining Resource Tags

Function

This API is used to obtain resource tags.

Calling Method

For details, see [Calling APIs](#).

URI

GET /cce/v1/{project_id}/{resource_type}/{resource_id}/tags

Table 4-1112 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
resource_type	Yes	String	Definition Resource type Constraints N/A Range • cce-cluster: cluster Default Value N/A
resource_id	Yes	String	Definition Resource ID, for example, cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints N/A Range N/A Default Value N/A

Request Parameters

Table 4-1113 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1114 Response body parameters

Parameter	Type	Description
tags	Array of ResourceTag objects	Definition Custom tags Constraints N/A
sys_tags	Array of ResourceTag objects	Definition System tags Constraints N/A

Table 4-1115 ResourceTag

Parameter	Type	Description
key	String	Details: Key Constraints: None Options: • The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. • The value can contain letters, digits, and spaces in UTF-8 format. • The value can contain the following special characters: _.:=-@ • The value cannot start with _sys_. Default value: N/A

Parameter	Type	Description
value	String	Details: Value Constraints: None Options: • The value can be null but not the default. Max characters: 255 • Letters, digits, and spaces in UTF-8 format are supported. • The value can contain the following special characters: _./=+-@ Default value: N/A

Example Requests

None

Example Responses

Status code: 200

The resource tags have been obtained.

```
{
  "tags": [ {
    "key": "key1",
    "value": "value1"
  } ],
  "sys_tags": [ {
    "key": "key2",
    "value": "value2"
  } ]
}
```

Status Codes

Status Code	Description
200	The resource tags have been obtained.

Error Codes

See [Error Codes](#).

4.10.4 Obtaining Custom Tags

Function

This API is used to obtain custom tags.

Calling Method

For details, see [Calling APIs](#).

URI

GET /cce/v1/{project_id}/{resource_type}/tags

Table 4-1116 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
resource_type	Yes	String	Definition Resource type Constraints N/A Range • cce-cluster: cluster Default Value N/A

Request Parameters

Table 4-1117 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1118 Response body parameters

Parameter	Type	Description
tags	Array of CustomizeResourceTag objects	Definition Custom tags Constraints N/A

Table 4-1119 CustomizeResourceTag

Parameter	Type	Description
key	String	Definition Key Constraints N/A Range - The value cannot be empty and cannot start or end with spaces. A maximum of 128 characters is supported. - The value can contain letters, digits, and spaces in UTF-8 format. - The value can contain the following special characters: _.:=-@ - The value cannot start with _sys_. Default Value N/A
values	Array of strings	Definition Value list Constraints N/A Range N/A Default Value N/A

Example Requests

None

Example Responses

Status code: 200

The custom tags have been obtained.

```
{
  "tags" : [ {
    "key" : "key1",
    "values" : [ "value1", "value2" ]
  } ]
}
```

Status Codes

Status Code	Description
200	The custom tags have been obtained.

Error Codes

See [Error Codes](#).

4.10.5 Obtaining Node Labels

Function

This API is used to obtain the labels of all nodes in a cluster.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/labels

Table 4-1120 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1121 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1122 Response body parameters

Parameter	Type	Description
kind	String	Definition API type Constraints N/A Range N/A Default Value Labels
apiVersion	String	Definition API version Constraints N/A Range N/A Default Value v3
spec	Map<String,Map<String,Array<String>>>	Definition Node label, which is classified by node pool. • key indicates the node pool ID. The value for the default node pool is DefaultPool. • value indicates the label in the format of key-value pairs. key indicates the label key, and value indicates the value list corresponding to the label key. Constraints N/A

Example Requests

None

Example Responses

Status code: 200

The node labels have been obtained.

```
{
  "kind" : "Labels",
```

```
"apiVersion" : "v3",
"spec" : {
  "nodePoolId" : {
    "key1" : [ "value1", "value2" ]
  },
  "DefaultPool" : {
    "key2" : [ "value1", "value2" ]
  }
}
```

Status Codes

Status Code	Description
200	The node labels have been obtained.

Error Codes

See [Error Codes](#).

4.11 Configuration Management

4.11.1 Obtaining the List of Parameters That Can Be Configured for a Node Pool

Function

This API is used to obtain the list of parameters that can be configured for a CCE node pool.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/configuration/detail

Table 4-1123 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1124 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1125 Response body parameters

Parameter	Type	Description
{User defined key}	Map<String,Array < PackageOptions >>	Response body for obtaining the parameter configuration list of a specified node pool

Table 4-1126 PackageOptions

Parameter	Type	Description
name	String	Definition Parameter name Range N/A
default	Object	Definition Default parameter value. If this parameter is not specified, the default value will take effect. The returned parameter can be of integer, string, or Boolean type. Range N/A
validAt	String	Definition How parameters will take effect Range • static: The parameters take effect during the cluster or node pool creation and cannot be changed later. • immediately: The parameters can be modified during the cluster or node pool running, and the modification takes effect immediately.
empty	Boolean	Definition Whether a configuration item can be empty Range • true: If the configuration item is empty, the default value is not used. The value remains empty. • false: If the configuration item is empty, the default value is used.

Parameter	Type	Description
schema	String	Definition Parameter category Range N/A
type	String	Definition Parameter type Range N/A

Example Requests

None

Example Responses

Status code: 200

The list of parameters that can be configured for the node pool is obtained.

```
{
  "kubelet" : [ {
    "name" : "kube-api-qps",
    "default" : 300,
    "validAt" : "immediately",
    "empty" : false,
    "schema" : "",
    "type" : "float"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowNodePoolConfigurationDetailsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
    }
}
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ShowNodePoolConfigurationDetailsRequest request = new
ShowNodePoolConfigurationDetailsRequest();
request.withClusterId("{cluster_id}");
request.withNodepoolId("{nodepool_id}");
try {
    ShowNodePoolConfigurationDetailsResponse response =
client.showNodePoolConfigurationDetails(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowNodePoolConfigurationDetailsRequest()
        request.cluster_id = "{cluster_id}"
        request.nodepool_id = "{nodepool_id}"
        response = client.show_node_pool_configuration_details(request)
```

```
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowNodePoolConfigurationDetailsRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    response, err := client.ShowNodePoolConfigurationDetails(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The list of parameters that can be configured for the node pool is obtained.

Error Codes

See [Error Codes](#).

4.11.2 Obtaining the List of Parameters That Can Be Configured for a Cluster

Function

This API is used to obtain the list of parameters that can be configured for a CCE cluster.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/configuration/detail

Table 4-1127 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1128 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1129 Response body parameters

Parameter	Type	Description
{User defined key}	Map<String,Array < PackageOptions >>	Definition Configuration items supported by a cluster

Table 4-1130 PackageOptions

Parameter	Type	Description
name	String	Definition Parameter name Range N/A
default	Object	Definition Default parameter value. If this parameter is not specified, the default value will take effect. The returned parameter can be of integer, string, or Boolean type. Range N/A
validAt	String	Definition How parameters will take effect Range • static: The parameters take effect during the cluster or node pool creation and cannot be changed later. • immediately: The parameters can be modified during the cluster or node pool running, and the modification takes effect immediately.
empty	Boolean	Definition Whether a configuration item can be empty Range • true: If the configuration item is empty, the default value is not used. The value remains empty. • false: If the configuration item is empty, the default value is used.

Parameter	Type	Description
schema	String	Definition Parameter category Range N/A
type	String	Definition Parameter type Range N/A

Example Requests

None

Example Responses

Status code: 200

The list of parameters that can be configured for a cluster is obtained.

```
{
  "kube-apiserver" : [ {
    "name" : "default-not-ready-toleration-seconds",
    "default" : 300,
    "validAt" : "immediately",
    "empty" : true,
    "schema" : "kubernetes",
    "type" : "int"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowClusterConfigurationDetailsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
    }
}
```



```
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ShowClusterConfigurationDetailsRequest request = new ShowClusterConfigurationDetailsRequest();
request.withClusterId("{cluster_id}");
try {
    ShowClusterConfigurationDetailsResponse response =
client.showClusterConfigurationDetails(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowClusterConfigurationDetailsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_cluster_configuration_details(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
```

```
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowClusterConfigurationDetailsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowClusterConfigurationDetails(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The list of parameters that can be configured for a cluster is obtained.

Error Codes

See [Error Codes](#).

4.11.3 Obtaining the List of Configurable Parameters Supported by a Cluster

Function

This API is used to obtain the detailed configuration items supported by a cluster based on factors like the cluster version or type. It is specified during cluster creation.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/clusters/configuration/detail

Table 4-1131 Query Parameters

Parameter	Mandatory	Type	Description
clusterType	No	String	Definition Used to filter the cluster architecture. Constraints N/A Range ARM64: Only the configuration items supported by the Kunpeng cluster are obtained. Default Value N/A
clusterVersion	No	String	Definition Used to obtain the configuration items supported by a specified cluster version. Constraints N/A Range N/A Default Value N/A

Parameter	Mandatory	Type	Description
clusterID	No	String	Definition Used to obtain the configuration items supported by a specified cluster. Constraints N/A Range N/A Default Value N/A
networkMode	No	String	Definition Used to filter out the configuration items related to the cluster network model. Constraints N/A Range • eni: The configurations related to the Cloud Native 2.0 network model are filtered out. Default Value N/A

Request Parameters

Table 4-1132 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1133 Response body parameters

Parameter	Type	Description
{User defined key}	Map<String,Array < PackageOptions >>	Definition Configuration items supported by a cluster

Table 4-1134 PackageOptions

Parameter	Type	Description
name	String	Definition Parameter name Range N/A
default	Object	Definition Default parameter value. If this parameter is not specified, the default value will take effect. The returned parameter can be of integer, string, or Boolean type. Range N/A
validAt	String	Definition How parameters will take effect Range • static: The parameters take effect during the cluster or node pool creation and cannot be changed later. • immediately: The parameters can be modified during the cluster or node pool running, and the modification takes effect immediately.
empty	Boolean	Definition Whether a configuration item can be empty Range • true: If the configuration item is empty, the default value is not used. The value remains empty. • false: If the configuration item is empty, the default value is used.

Parameter	Type	Description
schema	String	Definition Parameter category Range N/A
type	String	Definition Parameter type Range N/A

Example Requests

None

Example Responses

Status code: 200

The list of configurable parameters supported by the cluster has been obtained.

```
{
  "eni" : [ {
    "name" : "nic-minimum-target",
    "default" : 10,
    "validAt" : "immediately",
    "empty" : false,
    "schema" : "network",
    "type" : "str"
  }, {
    "name" : "nic-warm-target",
    "default" : 2,
    "validAt" : "immediately",
    "empty" : false,
    "schema" : "network",
    "type" : "str"
  } ],
  "kube-apiserver" : [ {
    "name" : "default-not-ready-toleration-seconds",
    "default" : 300,
    "validAt" : "immediately",
    "empty" : true,
    "schema" : "kubernetes",
    "type" : "int"
  } ]
}
```

Status Codes

Status Code	Description
200	The list of configurable parameters supported by the cluster has been obtained.

Error Codes

See [Error Codes](#).

4.11.4 Obtaining the Parameters That Can Be Configured for a Node Pool

Function

This API is used to obtain the parameters that can be configured for a node pool.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/configuration

Table 4-1135 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Parameter	Mandatory	Type	Description
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1136 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1137 Response body parameters

Parameter	Type	Description
apiVersion	String	API version
kind	String	API type. The value is fixed at Configuration .
metadata	ConfigurationMetadata object	Configuration metadata
spec	ClusterConfigurationsSpec object	Configuration specifications
status	Object	Configuration status

Table 4-1138 ConfigurationMetadata

Parameter	Type	Description
name	String	Configuration name

Parameter	Type	Description
labels	Map<String,String>	A configuration label in a key-value pair. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Example: "foo": "bar"

Table 4-1139 ClusterConfigurationsSpec

Parameter	Type	Description
packages	Array of packages objects	Component configuration item details

Table 4-1140 packages

Parameter	Type	Description
name	String	Component name
configurations	Array of ConfigurationItem objects	Component configuration items

Table 4-1141 ConfigurationItem

Parameter	Type	Description
name	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A
value	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Example Requests

None

Example Responses

Status code: 200

The parameter configurations of the node pool are obtained successfully.

```
{
  "kind": "Configuration",
  "apiVersion": "v3",
  "metadata": {
    "name": "configuration",
    "labels": {
      "nodepool_id": "61de338d-a1f9-11ed-8891-0255ac100036"
    }
  },
  "spec": {
    "packages": [ {
      "name": "kube-apiserver",
      "configurations": [ {
        "name": "event-rate-limit-qps",
        "value": 200
      }, {
        "name": "support-overload",
        "value": false
      } ]
    }, {
      "name": "kube-scheduler",
      "configurations": [ {
        "name": "kube-api-qps",
        "value": 100
      }, {
        "name": "default-scheduler",
        "value": "kube-scheduler"
      } ]
    } ]
  },
  "status": { }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowNodePoolConfigurationsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```

```
String projectId = "{project_id}";

ICredential auth = new BasicCredentials()
    .withProjectId(projectId)
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();

ShowNodePoolConfigurationsRequest request = new ShowNodePoolConfigurationsRequest();
request.withClusterId("{cluster_id}");
request.withNodepoolId("{nodepool_id}");
try {
    ShowNodePoolConfigurationsResponse response = client.showNodePoolConfigurations(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowNodePoolConfigurationsRequest()
        request.cluster_id = "{cluster_id}"
        request.nodepool_id = "{nodepool_id}"
        response = client.show_node_pool_configurations(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowNodePoolConfigurationsRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    response, err := client.ShowNodePoolConfigurations(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The parameter configurations of the node pool are obtained successfully.

Error Codes

See [Error Codes](#).

4.11.5 Changing the Values of Configuration Parameters of a Node Pool

Function

This API is used to change the values of parameter configurations of a node pool in the CCE cluster.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/configuration

Table 4-1142 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Parameter	Mandatory	Type	Description
nodepool_id	Yes	String	Details: Node pool ID. For details about how to obtain the ID, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1143 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1144 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	API version. The value is fixed at v3 .
kind	Yes	String	API type. The value is fixed at Configuration .
metadata	Yes	ConfigurationMetadata object	Configuration metadata
spec	Yes	ClusterConfigurationsSpec object	Configuration specifications

Table 4-1145 ConfigurationMetadata

Parameter	Mandatory	Type	Description
name	Yes	String	Configuration name

Parameter	Mandatory	Type	Description
labels	No	Map<String,String>	A configuration label in a key-value pair. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Example: "foo": "bar"

Table 4-1146 ClusterConfigurationsSpec

Parameter	Mandatory	Type	Description
packages	Yes	Array of packages objects	Component configuration item details

Table 4-1147 packages

Parameter	Mandatory	Type	Description
name	No	String	Component name
configurations	No	Array of ConfigurationItem objects	Component configuration items

Table 4-1148 ConfigurationItem

Parameter	Mandatory	Type	Description
name	No	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations. For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations. Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Parameter	Mandatory	Type	Description
value	No	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations . For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations . Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1149 Response body parameters

Parameter	Type	Description
apiVersion	String	API version
kind	String	API type. The value is fixed at Configuration .
metadata	ConfigurationMetadata object	Configuration metadata
spec	ClusterConfigurationsSpec object	Configuration specifications
status	Object	Configuration status

Table 4-1150 ConfigurationMetadata

Parameter	Type	Description
name	String	Configuration name
labels	Map<String,String>	A configuration label in a key-value pair. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain containing a maximum of 253 characters (such as example.com/my-key) can be prefixed to a key. • Value: The value can be left blank or contain 1 to 63 characters that start and end with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. Example: "foo": "bar"

Table 4-1151 ClusterConfigurationsSpec

Parameter	Type	Description
packages	Array of packages objects	Component configuration item details

Table 4-1152 packages

Parameter	Type	Description
name	String	Component name
configurations	Array of ConfigurationItem objects	Component configuration items

Table 4-1153 ConfigurationItem

Parameter	Type	Description
name	String	Definition: Component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations. For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations. Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A
value	AnyType	Definition: Values of component configuration parameters. For details about the components and parameters that can be configured for the current cluster, see Modifying CCE Cluster Configurations. For details about the components and parameters that can be configured for the current node pool, see Modifying Node Pool Configurations. Constraints: If you specify a component or parameter that is not supported, the configuration item will be ignored. Options: N/A Default value: N/A

Example Requests

The following example describes how to update the **system-reserved-mem** and **kube-reserved-mem** parameters in kubelet.

```
/api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/configuration
{
  "kind" : "Configuration",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "configuration"
  },
  "spec" : {
    "packages" : [ {
      "name" : "kubelet",
      "configurations" : [ {
        "name" : "system-reserved-mem",
        "value" : 600
      }, {
        "name" : "kube-reserved-mem",
        "value" : 800
      } ]
    } ]
  }
}
```

Example Responses

Status code: 200

The parameter configurations of the node pool are updated successfully.

```
{
  "kind" : "Configuration",
  "apiVersion" : "v3",
  "metadata" : {
    "name" : "configuration",
    "labels" : {
      "nodepool_id" : "61de338d-a1f9-11ed-8891-0255ac100036"
    }
  },
  "spec" : {
    "packages" : [ {
      "name" : "kube-apiserver",
      "configurations" : [ {
        "name" : "event-rate-limit-qps",
        "value" : 200
      }, {
        "name" : "support-overload",
        "value" : false
      } ]
    }, {
      "name" : "kube-scheduler",
      "configurations" : [ {
        "name" : "kube-api-qps",
        "value" : 100
      }, {
        "name" : "default-scheduler",
        "value" : "kube-scheduler"
      } ]
    } ]
  },
  "status" : { }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

The following example describes how to update the **system-reserved-mem** and **kube-reserved-mem** parameters in kubelet.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateNodePoolConfigurationSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        UpdateNodePoolConfigurationRequest request = new UpdateNodePoolConfigurationRequest();
        request.withClusterId("{cluster_id}");
        request.withNodepoolId("{nodepool_id}");
        UpdateClusterConfigurationsBody body = new UpdateClusterConfigurationsBody();
        List<ConfigurationItem> listPackagesConfigurations = new ArrayList<>();
        listPackagesConfigurations.add(
            new ConfigurationItem()
                .withName("system-reserved-mem")
                .withValue("600")
        );
        listPackagesConfigurations.add(
            new ConfigurationItem()
                .withName("kube-reserved-mem")
                .withValue("800")
        );
        List<ClusterConfigurationsSpecPackages> listSpecPackages = new ArrayList<>();
        listSpecPackages.add(
            new ClusterConfigurationsSpecPackages()
                .withName("kubelet")
                .withConfigurations(listPackagesConfigurations)
        );
        ClusterConfigurationsSpec specbody = new ClusterConfigurationsSpec();
        specbody.withPackages(listSpecPackages);
        ConfigurationMetadata metadatabody = new ConfigurationMetadata();
        metadatabody.withName("configuration");
        body.withSpec(specbody);
        body.withMetadata(metadatabody);
        body.withKind("Configuration");
        body.withApiVersion("v3");
    }
}
```

```
request.withBody(body);
try {
    UpdateNodePoolConfigurationResponse response = client.updateNodePoolConfiguration(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

The following example describes how to update the **system-reserved-mem** and **kube-reserved-mem** parameters in kubelet.

coding: utf-8

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateNodePoolConfigurationRequest()
        request.cluster_id = "{cluster_id}"
        request.nodepool_id = "{nodepool_id}"
        listConfigurationsPackages = [
            ConfigurationItem(
                name="system-reserved-mem",
                value="600"
            ),
            ConfigurationItem(
                name="kube-reserved-mem",
                value="800"
            )
        ]
        listPackagesSpec = [
            ClusterConfigurationsSpecPackages(
                name="kubelet",
                configurations=listConfigurationsPackages
            )
        ]
        specbody = ClusterConfigurationsSpec(
```

```
        packages=listPackagesSpec
    )
    metadatabody = ConfigurationMetadata(
        name="configuration"
    )
    request.body = UpdateClusterConfigurationsBody(
        spec=specbody,
        metadata=metadatabody,
        kind="Configuration",
        api_version="v3"
    )
    response = client.update_node_pool_configuration(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

The following example describes how to update the **system-reserved-mem** and **kube-reserved-mem** parameters in kubelet.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateNodePoolConfigurationRequest{}
    request.ClusterId = "{cluster_id}"
    request.NodepoolId = "{nodepool_id}"
    nameConfigurations := "system-reserved-mem"
    valueConfigurations := "600"
    var valueConfigurationsInterface interface{} = valueConfigurations
    nameConfigurations1 := "kube-reserved-mem"
    valueConfigurations1 := "800"
    var valueConfigurationsInterface1 interface{} = valueConfigurations1
    var listConfigurationsPackages = []model.ConfigurationItem{
        {
            Name: &nameConfigurations,
            Value: &valueConfigurationsInterface,
```

```
    },  
    {  
        Name: &nameConfigurations1,  
        Value: &valueConfigurationsInterface1,  
    },  
}  
namePackages:= "kubelet"  
var listPackagesSpec = []model.ClusterConfigurationsSpecPackages{  
    {  
        Name: &namePackages,  
        Configurations: &listConfigurationsPackages,  
    },  
}  
specbody := &model.ClusterConfigurationsSpec{  
    Packages: listPackagesSpec,  
}  
metadatabody := &model.ConfigurationMetadata{  
    Name: "configuration",  
}  
request.Body = &model.UpdateClusterConfigurationsBody{  
    Spec: specbody,  
    Metadata: metadatabody,  
    Kind: "Configuration",  
    ApiVersion: "v3",  
}  
response, err := client.UpdateNodePoolConfiguration(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The parameter configurations of the node pool are updated successfully.

Error Codes

See [Error Codes](#).

4.12 Chart Management

4.12.1 Uploading a Chart

Function

This API is used to upload a chart.

Calling Method

For details, see [Calling APIs](#).

URI

POST /v2/charts

Request Parameters

Table 4-1154 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1155 FormData parameters

Parameter	Mandatory	Type	Description
parameters	No	String	Parameter configurations for uploading a chart. The following is an example: {"override":true,"skip_lint":true,"source":"package"}" skip_lint: whether lint uploaded chart - override: whether override existed chart - visible: update chart visible
content	Yes	File	Chart package file

Response Parameters

Status code: 201

Table 4-1156 Response body parameters

Parameter	Type	Description
id	String	Chart ID
name	String	Chart name
values	String	Chart value
translate	String	Chart translation resources
instruction	String	Chart description
version	String	Chart version
description	String	Chart description
source	String	Chart source
icon_url	String	URL to chart icons
public	Boolean	Whether the chart is public
chart_url	String	URL to the chart
create_at	String	Creation time.
update_at	String	Update time.

Example Requests

If the **FormData** parameter is present, **multipart/form-data** should be used in **Content-Type** and the file path should be specified.

```
POST /v2/charts -H "X-Auth-Token:$token" -H "Content-Type:multipart/form-data" -F
parameters="{\"skip_lint\":true,\"override\":true,\"source\":\"package\"}" -F content=@/root/neo4j-3.0.1.tgz
```

Example Responses

Status code: 201

Created

```
{
  "id" : "e99a7e86-afdd-11eb-aca3-0255ac100b0e",
  "name" : "neo4j",
  "values" : "{\"acceptLicenseAgreement\":\"no\",\"affinity\":{\"authEnabled\":true,\"clusterDomain\":\"cluster.local\",\"core\":{\"initContainers\":[],\"numberOfServers\":3,\"persistentVolume\":{\"enabled\":true,\"mountPath\":\"/data\",\"size\":\"10Gi\"},\"sidecarContainers\":[],\"defaultDatabase\":\"neo4j\",\"image\":\"neo4j\",\"imagePullPolicy\":\"IfNotPresent\",\"imageTag\":\"4.0.3-enterprise\",\"name\":\"neo4j\",\"nodeSelector\":{\"podDisruptionBudget\":{\"readReplica\":{\"autoscaling\":{\"enabled\":false,\"maxReplicas\":3,\"minReplicas\":1,\"targetAverageUtilization\":70},\"initContainers\":[],\"numberOfServers\":0,\"resources\":{\"sidecarContainers\":[],\"resources\":{\"testImage\":\"markhneedham/k8s-kubectl\",\"testImageTag\":\"master\",\"tolerations\":[],\"useAPOC\":\"true\"}},\"translate\":\"\", \"instruction\" : \"README.md\", \"version\" : \"3.0.1\", \"description\" : \"DEPRECATED Neo4j is the world's leading graph database\", \"source\" : \"\", \"icon_url\" : \"https://info.neo4j.com/rs/773-GON-065/images/neo4j_logo.png\", \"public\" : false, \"chart_url\" : \"neo4j-3.0.1.tgz\", \"create_at\" : \"2021-05-08T08:53:13Z\", \"update_at\" : \"2021-05-08T08:53:13Z\"}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UploadChartSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```

```
ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
UploadChartRequest request = new UploadChartRequest();
try {
    UploadChartResponse response = client.uploadChart(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UploadChartRequest()
        response = client.upload_chart(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```



```
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UploadChartRequest{}
    response, err := client.UploadChart(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	Created

Error Codes

See [Error Codes](#).

4.12.2 Obtaining a Chart List

Function

This API is used to obtain the chart list.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v2/charts

Request Parameters

Table 4-1157 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1158 Response body parameters

Parameter	Type	Description
[items]	Array of ChartResp objects	Chart list

Table 4-1159 ChartResp

Parameter	Type	Description
id	String	Chart ID
name	String	Chart name
values	String	Chart value
translate	String	Chart translation resources
instruction	String	Chart description
version	String	Chart version
description	String	Chart description
source	String	Chart source
icon_url	String	URL to chart icons
public	Boolean	Whether the chart is public
chart_url	String	URL to the chart
create_at	String	Creation time.
update_at	String	Update time.

Example Requests

None

Example Responses

Status code: 200

OK

```
[{"id": "1abd3bd6-0258-11ec-b8b0-0255ac100b05",
  "name": "magento-mysql",
  "values": "{\"basic\":{\"admin_password\":\"*****\",\"admin_username\":\"username\",\"app_name\":\"magento\",\"mysql_database\":\"magento\",\"mysql_name\":\"mysql\",\"mysql_password\":\"*****\",\"mysql_port\":\"3306\",\"mysql_root_password\":\"*****\",\"mysql_user\":\"magento\",\"storage_class\":\"csin-as\",\"storage_model\":\"ReadWriteMany\",\"storage_size\":\"10G\"},\"global\":{\"magento_EIP\":\"100.100.100.100\",\"magento_EPORT\":\"32080\",\"namespace\":\"default\",\"image\":{\"magento_image\":\"example.com/everest/magento:latest\",\"mysql_image\":\"example.com/everest/mysql:5.7.14\"}},\"translate\": \"\"}
```

```
"instruction" : "",
"version" : "1.0.0",
"description" : "chart description",
"source" : "",
"icon_url" : "https://example.com/magento-stack-110x117.png",
"public" : false,
"chart_url" : "magento-mysql-1.0.0.tgz",
"create_at" : "2021-08-20T08:00:29Z",
"update_at" : "2021-08-20T08:00:29Z"
}]
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListChartsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ListChartsRequest request = new ListChartsRequest();
        try {
            ListChartsResponse response = client.listCharts(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListChartsRequest()
        response = client.list_charts(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListChartsRequest{}
```

```
response, err := client.ListCharts(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.3 Obtaining a Release List

Function

This API is used to obtain a release list.

Calling Method

For details, see [Calling APIs](#).

URI

GET /cce/cam/v3/clusters/{cluster_id}/releases

Table 4-1160 Path Parameters

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-1161 Query Parameters

Parameter	Mandatory	Type	Description
chart_id	No	String	Chart ID
namespace	No	String	Namespace of the chart

Request Parameters

Table 4-1162 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1163 Response body parameters

Parameter	Type	Description
[items]	Array of ReleaseResp objects	

Table 4-1164 ReleaseResp

Parameter	Type	Description
chart_name	String	Chart name
chart_public	Boolean	Whether the chart is public
chart_version	String	Chart version
cluster_id	String	Cluster ID
cluster_name	String	Current name
create_at	String	Creation time.
description	String	Release description
name	String	Release name
namespace	String	Namespace to which a chart release belongs
parameters	String	Release parameters
resources	String	Resources required by the release

Parameter	Type	Description
status	String	Release status. • DEPLOYED: The release is normal. • DELETED: The release has been deleted. • FAILED: The release fails to be deployed. • DELETING: The release is being deleted. • PENDING_INSTALL: The release is waiting to be installed. • PENDING_UPGRADE: The release is waiting to be upgraded. • PENDING_ROLLBACK: The release is waiting for rollback. • UNKNOWN: The release status is unknown, indicating that the release is abnormal. You can manually delete the release and reinstall it.
status_description	String	Release status description
update_at	String	Update time.
values	String	Release value
version	Integer	Release version

Example Requests

None

Example Responses

Status code: 200

OK

```
[ {
  "chart_name": "magento-mysql",
  "chart_public": false,
  "chart_version": "1.0.0",
  "cluster_id": "a870253f-5dc7-11ee-bf71-0255ac100b03",
  "cluster_name": "sfs-turbo-test",
  "create_at": "2023-11-14T20:30:57+08:00",
  "description": "Initial install underway",
  "name": "testwww",
  "namespace": "monitoring",
  "parameters": "",
  "resources": "",
  "status": "PENDING_INSTALL",
  "status_description": "Initial install underway",
```

```
"update_at" : "2023-11-14T20:30:57+08:00",
"values" : -{"basic":{"admin_password":{"*****"},"admin_username":{"username"},"app_name":{"magento"},"mysql_database":{"magento"},"mysql_name":{"mysql"},"mysql_password":{"*****"},"mysql_port":{"3306"},"mysql_root_password":{"*****"},"mysql_user":{"magento"},"storage_class":{"csinas"},"storage_model":{"ReadWriteMany"},"storage_size":{"10G"},"global":{"magento_EIP":{"100.100.100.100"},"magento_EPORT":{"32080"},"namespace":{"default"},"image":{"magento_image":{"example.com/everest/magento:latest"},"mysql_image":{"example.com/everest/mysql:5.7.14"}}},
"version" : 1
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ListReleasesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR_REGION>"))
            .build();

        ListReleasesRequest request = new ListReleasesRequest();
        request.withClusterId("{cluster_id}");
        try {
            ListReleasesResponse response = client.listReleases(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListReleasesRequest()
        request.cluster_id = "{cluster_id}"
        response = client.list_releases(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.ListReleasesRequest{}
request.ClusterId = "{cluster_id}"
response, err := client.ListReleases(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.4 Updating a Chart

Function

This API is used to update a chart.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /v2/charts/{chart_id}

Table 4-1165 Path Parameters

Parameter	Mandatory	Type	Description
chart_id	Yes	String	Chart ID

Request Parameters

Table 4-1166 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1167 FormData parameters

Parameter	Mandatory	Type	Description
parameters	No	String	Parameter configurations for uploading a chart. The following is an example: {"override":true,"skip_lint":true,"source":"package"}"- skip_lint: whether lint uploaded chart - override: whether override existed chart - visible: update chart visible
content	Yes	File	Chart package file

Response Parameters

Status code: 200

Table 4-1168 Response body parameters

Parameter	Type	Description
id	String	Chart ID
name	String	Chart name
values	String	Chart value
translate	String	Chart translation resources
instruction	String	Chart description
version	String	Chart version
description	String	Chart description
source	String	Chart source
icon_url	String	URL to chart icons
public	Boolean	Whether the chart is public
chart_url	String	URL to the chart
create_at	String	Creation time.
update_at	String	Update time.

Example Requests

Use **multipart/form-data** for **Content-Type** and specify the file path when the **FormData** parameter is present.

```
PUT /v2/charts/{chart_id} -H "X-Auth-Token:$token" -H "Content-Type:multipart/form-data" -F
parameters='{ "skip_lint":true,"override":true,"source":"package"}' -F content=@/root/neo4j-3.0.1.tgz
```

Example Responses

Status code: 200

OK

```
{
  "id": "e99a7e86-afdd-11eb-aca3-0255ac100b0e",
  "name": "neo4j",
  "values": "{ \"acceptLicenseAgreement\": \"no\", \"affinity\": {}, \"authEnabled\": true, \"clusterDomain\": \"cluster.local\", \"core\": { \"initContainers\": [], \"numberOfServers\": 3, \"persistentVolume\": { \"enabled\": true, \"mountPath\": \"/data\", \"size\": \"10Gi\" }, \"sidecarContainers\": [], \"defaultDatabase\": \"neo4j\", \"image\": \"neo4j\", \"imagePullPolicy\": \"IfNotPresent\", \"imageTag\": \"4.0.3-enterprise\", \"name\": \"neo4j\", \"nodeSelector\": {}, \"podDisruptionBudget\": {}, \"readReplica\": { \"autoscaling\": { \"enabled\": false, \"maxReplicas\": 3, \"minReplicas\": 1, \"targetAverageUtilization\": 70 }, \"initContainers\": [], \"numberOfServers\": 0, \"resources\": {}, \"sidecarContainers\": [], \"resources\": {} }, \"testImage\": \"markhneedham/k8s-kubectl\", \"testImageTag\": \"master\", \"tolerations\": [], \"useAPOC\": \"true\" }",
  "translate": "",
  "instruction": "README.md",
  "version": "3.0.1",
  "description": "DEPRECATED Neo4j is the world's leading graph database",
  "source": "",
  "icon_url": "https://example.com/images/neo4j_logo.png",
  "public": false,
  "chart_url": "neo4j-3.0.1.tgz",
  "create_at": "2021-05-08T08:53:13Z",
  "update_at": "2021-05-08T08:53:13Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpdateChartSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
```



```
        .withRegion(CceRegion.valueOf("<YOUR REGION>"))
        .build();
UpdateChartRequest request = new UpdateChartRequest();
request.withChartId("{chart_id}");
try {
    UpdateChartResponse response = client.updateChart(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.valueOf("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateChartRequest()
        request.chart_id = "{chart_id}"
        response = client.update_chart(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)
```

```
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := cce.NewCceClient(  
        cce.CceClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.UpdateChartRequest{}  
    request.ChartId = "{chart_id}"  
    response, err := client.UpdateChart(request)  
    if err == nil {  
        fmt.Printf("%v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.5 Creating a Release

Function

This API is used to create a release.

Calling Method

For details, see [Calling APIs](#).

URI

POST /cce/cam/v3/clusters/{cluster_id}/releases

Table 4-1169 Path Parameters

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1170 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1171 Request body parameters

Parameter	Mandatory	Type	Description
chart_id	Yes	String	Chart ID
description	No	String	Release description
name	Yes	String	Release name
namespace	Yes	String	Namespace to which a chart release belongs
version	Yes	String	Release version
parameters	No	ReleaseReqBodyParams object	Release parameters
values	Yes	values object	Release value

Table 4-1172 ReleaseReqBodyParams

Parameter	Mandatory	Type	Description
dry_run	No	Boolean	After this function is enabled, only chart parameters are verified, and installation is not performed.

Parameter	Mandatory	Type	Description
name_template	No	String	Release name template
no_hooks	No	Boolean	Whether to disable hooks during installation
replace	No	Boolean	Whether to replace the release with the same name
recreate	No	Boolean	Whether to rebuild the release
reset_values	No	Boolean	Whether to reset values during an update
release_version	No	Integer	Version of the rollback release
include_hooks	No	Boolean	Enable hooks during an update or deletion.

Table 4-1173 values

Parameter	Mandatory	Type	Description
imagePullPolicy	No	String	Image pull policy
imageTag	No	String	Image tag

Response Parameters

Status code: 201

Table 4-1174 Response body parameters

Parameter	Type	Description
chart_name	String	Chart name
chart_public	Boolean	Whether the chart is public
chart_version	String	Chart version
cluster_id	String	Cluster ID
cluster_name	String	Current name
create_at	String	Creation time.
description	String	Release description

Parameter	Type	Description
name	String	Release name
namespace	String	Namespace to which a chart release belongs
parameters	String	Release parameters
resources	String	Resources required by the release
status	String	Release status. • DEPLOYED: The release is normal. • DELETED: The release has been deleted. • FAILED: The release fails to be deployed. • DELETING: The release is being deleted. • PENDING_INSTALL: The release is waiting to be installed. • PENDING_UPGRADE: The release is waiting to be upgraded. • PENDING_ROLLBACK: The release is waiting for rollback. • UNKNOWN: The release status is unknown, indicating that the release is abnormal. You can manually delete the release and reinstall it.
status_description	String	Release status description
update_at	String	Update time.
values	String	Release value
version	Integer	Release version

Example Requests

```
POST /cce/cam/v3/clusters/{cluster_id}/releases
{
  "name": "nino21",
  "namespace": "project01",
  "version": "1.0.0",
  "chart_id": "3c138b72-7ce4-6d76-7c55-604cdb2ce423",
  "values": {
    "imageTag": "v2",
    "imagePullPolicy": "IfNotPresent"
  },
  "parameters": {
    "dry_run": false,
    "no_hooks": false,
  }
}
```

```
"replace" : false,  
"name_template" : ""  
}  
}
```

Example Responses

Status code: 201

Created

```
{  
  "chart_name" : "magento-mysql",  
  "chart_public" : false,  
  "chart_version" : "1.0.0",  
  "cluster_id" : "a870253f-5dc7-11ee-bf71-0255ac100b03",  
  "cluster_name" : "sfs-turbo-test",  
  "create_at" : "2023-11-14T20:30:57+08:00",  
  "description" : "Initial install underway",  
  "name" : "testwww",  
  "namespace" : "monitoring",  
  "parameters" : "",  
  "resources" : "",  
  "status" : "PENDING_INSTALL",  
  "status_description" : "Initial install underway",  
  "update_at" : "2023-11-14T20:30:57+08:00",  
  "values" : "{\n    \"basic\":{\n      \"admin_password\": \"*****\",\n      \"admin_username\": \"username\",\n      \"app_name\": \"magento\",\n      \"mysql_database\": \"magento\",\n      \"mysql_name\": \"mysql\",\n      \"mysql_password\": \"*****\",\n      \"mysql_port\": 3306,\n      \"mysql_root_password\": \"*****\",\n      \"mysql_user\": \"magento\",\n      \"storage_class\": \"csi-nas\",\n      \"storage_mode\": \"ReadWriteMany\",\n      \"storage_size\": \"10G\",\n      \"global\":{\n        \"magento_EIP\": \"100.100.100.100\",\n        \"magento_EPORT\": 32080,\n        \"namespace\": \"default\",\n        \"image\": \"magento_image\",\n        \"example.com/everest/magento:latest\",\n        \"mysql_image\": \"example.com/everest/mysql:5.7.14\"\n      }\n    }\n  }",  
  "version" : 1  
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.cce.v3.region.CceRegion;  
import com.huaweicloud.sdk.cce.v3.*;  
import com.huaweicloud.sdk.cce.v3.model.*;  
  
public class CreateReleaseSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);
```

```
CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
CreateReleaseRequest request = new CreateReleaseRequest();
request.withClusterId("{cluster_id}");
CreateReleaseReqBody body = new CreateReleaseReqBody();
CreateReleaseReqBodyValues valuesbody = new CreateReleaseReqBodyValues();
valuesbody.withImagePullPolicy("IfNotPresent")
    .withImageTag("v2");
ReleaseReqBodyParams parametersbody = new ReleaseReqBodyParams();
parametersbody.withDryRun(false)
    .withNameTemplate("")
    .withNoHooks(false)
    .withReplace(false);
body.withValues(valuesbody);
body.withParameters(parametersbody);
body.withVersion("1.0.0");
body.withNamespace("project01");
body.withName("nino21");
body.withChartId("3c138b72-7ce4-6d76-7c55-604cdb2ce423");
request.withBody(body);
try {
    CreateReleaseResponse response = client.createRelease(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateReleaseRequest()
        request.cluster_id = "{cluster_id}"
```



```
valuesbody = CreateReleaseReqBodyValues(
    image_pull_policy="IfNotPresent",
    image_tag="v2"
)
parametersbody = ReleaseReqBodyParams(
    dry_run=False,
    name_template="",
    no_hooks=False,
    replace=False
)
request.body = CreateReleaseReqBody(
    values=valuesbody,
    parameters=parametersbody,
    version="1.0.0",
    namespace="project01",
    name="nino21",
    chart_id="3c138b72-7ce4-6d76-7c55-604cdb2ce423"
)
response = client.create_release(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateReleaseRequest{}
    request.ClusterId = "{cluster_id}"
    imagePullPolicyValues := "IfNotPresent"
    imageTagValues := "v2"
    valuesbody := &model.CreateReleaseReqBodyValues{
        ImagePullPolicy: &imagePullPolicyValues,
        ImageTag: &imageTagValues,
    }
    dryRunParameters := false
    nameTemplateParameters := ""
    noHooksParameters := false
```

```
replaceParameters:= false
parametersbody := &model.ReleaseReqBodyParams{
    DryRun: &dryRunParameters,
    NameTemplate: &nameTemplateParameters,
    NoHooks: &noHooksParameters,
    Replace: &replaceParameters,
}
request.Body = &model.CreateReleaseReqBody{
    Values: valuesbody,
    Parameters: parametersbody,
    Version: "1.0.0",
    Namespace: "project01",
    Name: "nino21",
    ChartId: "3c138b72-7ce4-6d76-7c55-604cdb2ce423",
}
response, err := client.CreateRelease(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	Created

Error Codes

See [Error Codes](#).

4.12.6 Deleting a Chart

Function

This API is used to delete a chart.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /v2/charts/{chart_id}

Table 4-1175 Path Parameters

Parameter	Mandatory	Type	Description
chart_id	Yes	String	Chart ID

Request Parameters

Table 4-1176 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1177 Response body parameters

Parameter	Type	Description
id	String	Chart ID
name	String	Chart name
values	String	Chart value
translate	String	Chart translation resources
instruction	String	Chart description
version	String	Chart version
description	String	Chart description
source	String	Chart source
icon_url	String	URL to chart icons
public	Boolean	Whether the chart is public
chart_url	String	URL to the chart
create_at	String	Creation time.
update_at	String	Update time.

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "id" : "e99a7e86-afdd-11eb-aca3-0255ac100b0e",
  "name" : "neo4j",
  "values" : "{ \"acceptLicenseAgreement\" : \"no\", \"affinity\" : {}, \"authEnabled\" : true, \"clusterDomain\" : \"cluster.local\", \"core\" : { \"initContainers\" : [], \"numberOfServers\" : 3, \"persistentVolume\" : { \"enabled\" : true, \"mountPath\" : \"/data\", \"size\" : \"10Gi\" }, \"sidecarContainers\" : [], \"defaultDatabase\" : \"neo4j\", \"image\" : \"neo4j\", \"imagePullPolicy\" : \"IfNotPresent\", \"imageTag\" : \"4.0.3-enterprise\", \"name\" : \"neo4j\", \"nodeSelector\" : {}, \"podDisruptionBudget\" : {}, \"readReplica\" : { \"autoscaling\" : { \"enabled\" : false, \"maxReplicas\" : 3, \"minReplicas\" : 1, \"targetAverageUtilization\" : 70 }, \"initContainers\" : [], \"numberOfServers\" : 0, \"resources\" : {}, \"sidecarContainers\" : [], \"resources\" : {}, \"testImage\" : \"markhneedham/k8s-kubectl\", \"testImageTag\" : \"master\", \"tolerations\" : [], \"useAPOC\" : \"true\" } }, \"translate\" : \"\", \"instruction\" : \"README.md\", \"version\" : \"3.0.1\", \"description\" : \"DEPRECATED Neo4j is the world's leading graph database\", \"source\" : \"\",
```

```
"icon_url" : "https://example.com/images/neo4j_logo.png",
"public" : false,
"chart_url" : "neo4j-3.0.1.tgz",
"create_at" : "2021-05-08T08:53:12Z",
"update_at" : "2021-05-08T08:53:12Z"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class DeleteChartSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        DeleteChartRequest request = new DeleteChartRequest();
        request.withChartId("{chart_id}");
        try {
            DeleteChartResponse response = client.deleteChart(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8
```

```
import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteChartRequest()
        request.chart_id = "{chart_id}"
        response = client.delete_chart(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteChartRequest{}
    request.ChartId = "{chart_id}"
    response, err := client.DeleteChart(request)
    if err == nil {
```

```
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.7 Updating a Release

Function

This API is used to update a release.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /cce/cam/v3/clusters/{cluster_id}/namespace/{namespace}/releases/{name}

Table 4-1178 Path Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Release name
namespace	Yes	String	Namespace of the chart

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-1179 Query Parameters

Parameter	Mandatory	Type	Description
show_resources	No	String	Definition Whether to display the resource information of a release Constraints N/A Range If this parameter is set to true , the resource information of the release will be displayed. If this parameter is not specified, the resource information is not displayed by default. Default Value N/A

Request Parameters

Table 4-1180 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Table 4-1181 Request body parameters

Parameter	Mandatory	Type	Description
chart_id	Yes	String	Chart ID

Parameter	Mandatory	Type	Description
action	Yes	String	upgrade for an upgrade and rollback for a rollback
parameters	Yes	ReleaseReqBodyParams object	Release parameters
values	Yes	values object	Release value

Table 4-1182 ReleaseReqBodyParams

Parameter	Mandatory	Type	Description
dry_run	No	Boolean	After this function is enabled, only chart parameters are verified, and installation is not performed.
name_template	No	String	Release name template
no_hooks	No	Boolean	Whether to disable hooks during installation
replace	No	Boolean	Whether to replace the release with the same name
recreate	No	Boolean	Whether to rebuild the release
reset_values	No	Boolean	Whether to reset values during an update
release_version	No	Integer	Version of the rollback release
include_hooks	No	Boolean	Enable hooks during an update or deletion.

Table 4-1183 values

Parameter	Mandatory	Type	Description
imagePullPolicy	No	String	Image pull policy
imageTag	No	String	Image tag

Response Parameters

Status code: 200

Table 4-1184 Response body parameters

Parameter	Type	Description
chart_name	String	Chart name
chart_public	Boolean	Whether the chart is public
chart_version	String	Chart version
cluster_id	String	Cluster ID
cluster_name	String	Current name
create_at	String	Creation time.
description	String	Release description
name	String	Release name
namespace	String	Namespace to which a chart release belongs
parameters	String	Release parameters
resources	String	Resources required by the release
status	String	Release status. • DEPLOYED: The release is normal. • DELETED: The release has been deleted. • FAILED: The release fails to be deployed. • DELETING: The release is being deleted. • PENDING_INSTALL: The release is waiting to be installed. • PENDING_UPGRADE: The release is waiting to be upgraded. • PENDING_ROLLBACK: The release is waiting for rollback. • UNKNOWN: The release status is unknown, indicating that the release is abnormal. You can manually delete the release and reinstall it.
status_description	String	Release status description
update_at	String	Update time.

Parameter	Type	Description
values	String	Release value
version	Integer	Release version

Example Requests

```
PUT /cce/cam/v3/clusters/{cluster_id}/namespace/{namespace}/releases/{name}
```

```
{
  "chart_id": "af4b699e-018c-11ec-b8b0-0255ac100b05",
  "action": "upgrade",
  "parameters": {
    "dry_run": false,
    "name_template": "string",
    "no_hooks": false,
    "replace": false,
    "recreate": false,
    "reset_values": false,
    "release_version": 1,
    "include_hooks": false
  },
  "values": {
    "imagePullPolicy": "IfNotPresent",
    "imageTag": "v2"
  }
}
```

Example Responses

Status code: 200

OK

```
{
  "chart_name": "magento-mysql",
  "chart_public": false,
  "chart_version": "1.0.0",
  "cluster_id": "a870253f-5dc7-11ee-bf71-0255ac100b03",
  "cluster_name": "sfs-turbo-test",
  "create_at": "2023-11-14T20:30:57+08:00",
  "description": "Initial install underway",
  "name": "testwww",
  "namespace": "monitoring",
  "parameters": "",
  "resources": "",
  "status": "PENDING_INSTALL",
  "status_description": "Initial install underway",
  "update_at": "2023-11-14T20:30:57+08:00",
  "values": "{ \"basic\": { \"admin_password\": \"*****\", \"admin_username\": \"username\", \"app_name\": \"magento\", \"mysql_database\": \"magento\", \"mysql_name\": \"mysql\", \"mysql_password\": \"*****\", \"mysql_port\": 3306, \"mysql_root_password\": \"*****\", \"mysql_user\": \"magento\", \"storage_class\": \"csi-nas\", \"storage_model\": \"ReadWriteMany\", \"storage_size\": \"10G\", \"global\": { \"magento_EIP\": \"100.100.100.100\", \"magento_EPORT\": 32080, \"namespace\": \"default\", \"image\": { \"magento_image\": \"example.com/everest/magento:latest\", \"mysql_image\": \"example.com/everest/mysql:5.7.14\" } } }, \"version\": 1 } }
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class UpdateReleaseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateReleaseRequest request = new UpdateReleaseRequest();
        request.setName("{name}");
        request.withNamespace("{namespace}");
        request.withClusterId("{cluster_id}");
        UpdateReleaseReqBody body = new UpdateReleaseReqBody();
        UpdateReleaseReqBodyValues valuesbody = new UpdateReleaseReqBodyValues();
        valuesbody.withImagePullPolicy("IfNotPresent")
            .withImageTag("v2");
        ReleaseReqBodyParams parametersbody = new ReleaseReqBodyParams();
        parametersbody.withDryRun(false)
            .withNameTemplate("string")
            .withNoHooks(false)
            .withReplace(false)
            .withRecreate(false)
            .withResetValues(false)
            .withReleaseVersion(1)
            .withIncludeHooks(false);
        body.withValues(valuesbody);
        body.withParameters(parametersbody);
        body.withAction(UpdateReleaseReqBody.ActionEnum.fromValue("upgrade"));
        body.withChartId("af4b699e-018c-11ec-b8b0-0255ac100b05");
        request.withBody(body);
        try {
            UpdateReleaseResponse response = client.updateRelease(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}
```

Python

```
# coding: utf-8  
  
import os  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkcce.v3.region.cce_region import CceRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkcce.v3 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = os.environ["CLOUD_SDK_AK"]  
    sk = os.environ["CLOUD_SDK_SK"]  
  
    credentials = BasicCredentials(ak, sk)  
  
    client = CceClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(CceRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = UpdateReleaseRequest()  
        request.name = "{name}"  
        request.namespace = "{namespace}"  
        request.cluster_id = "{cluster_id}"  
        valuesbody = UpdateReleaseReqBodyValues(  
            image_pull_policy="IfNotPresent",  
            image_tag="v2"  
        )  
        parametersbody = ReleaseReqBodyParams(  
            dry_run=False,  
            name_template="string",  
            no_hooks=False,  
            replace=False,  
            recreate=False,  
            reset_values=False,  
            release_version=1,  
            include_hooks=False  
        )  
        request.body = UpdateReleaseReqBody(  
            values=valuesbody,  
            parameters=parametersbody,  
            action="upgrade",  
            chart_id="af4b699e-018c-11ec-b8b0-0255ac100b05"  
        )  
        response = client.update_release(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```

```
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateReleaseRequest{}
    request.Name = "{name}"
    request.Namespace = "{namespace}"
    request.ClusterId = "{cluster_id}"
    imagePullPolicyValues:= "IfNotPresent"
    imageTagValues:= "v2"
    valuesbody := &model.UpdateReleaseReqBodyValues{
        ImagePullPolicy: &imagePullPolicyValues,
        ImageTag: &imageTagValues,
    }
    dryRunParameters:= false
    nameTemplateParameters:= "string"
    noHooksParameters:= false
    replaceParameters:= false
    recreateParameters:= false
    resetValuesParameters:= false
    releaseVersionParameters:= int32(1)
    includeHooksParameters:= false
    parametersbody := &model.ReleaseReqBodyParams{
        DryRun: &dryRunParameters,
        NameTemplate: &nameTemplateParameters,
        NoHooks: &noHooksParameters,
        Replace: &replaceParameters,
        Recreate: &recreateParameters,
        ResetValues: &resetValuesParameters,
        ReleaseVersion: &releaseVersionParameters,
        IncludeHooks: &includeHooksParameters,
    }
    request.Body = &model.UpdateReleaseReqBody{
        Values: valuesbody,
        Parameters: parametersbody,
        Action: model.GetUpdateReleaseReqBodyActionEnum().UPGRADE,
        ChartId: "af4b699e-018c-11ec-b8b0-0255ac100b05",
    }
    response, err := client.UpdateRelease(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.8 Obtaining a Chart

Function

This API is used to obtain a chart.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v2/charts/{chart_id}

Table 4-1185 Path Parameters

Parameter	Mandatory	Type	Description
chart_id	Yes	String	Chart ID

Request Parameters

Table 4-1186 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1187 Response body parameters

Parameter	Type	Description
id	String	Chart ID
name	String	Chart name
values	String	Chart value
translate	String	Chart translation resources
instruction	String	Chart description
version	String	Chart version
description	String	Chart description
source	String	Chart source
icon_url	String	URL to chart icons
public	Boolean	Whether the chart is public
chart_url	String	URL to the chart
create_at	String	Creation time.
update_at	String	Update time.

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "id": "e99a7e86-afdd-11eb-aca3-0255ac100b0e",
  "name": "neo4j",
  "values": "{ \"acceptLicenseAgreement\": \"no\", \"affinity\": {}, \"authEnabled\": true, \"clusterDomain\": \"cluster.local\", \"core\": { \"initContainers\": [], \"numberOfServers\": 3, \"persistentVolume\": { \"enabled\": true, \"mountPath\": \"/data\", \"size\": \"10Gi\" }, \"sidecarContainers\": [], \"defaultDatabase\": \"neo4j\", \"image\": \"neo4j\", \"imagePullPolicy\": \"IfNotPresent\", \"imageTag\": \"4.0.3-enterprise\", \"name\": \"neo4j\", \"nodeSelector\": {}, \"podDisruptionBudget\": {}, \"readReplica\": { \"autoscaling\": { \"enabled\": false, \"maxReplicas\": 3, \"minReplicas\": 1, \"targetAverageUtilization\": 70 }, \"initContainers\": [], \"numberOfServers\": 0, \"resources\": {}, \"sidecarContainers\": [], \"resources\": {}, \"testImage\": \"markhneedham/k8s-kubectl\", \"testImageTag\": \"master\", \"tolerations\": [], \"useAPOC\": true } }\",
  \"translate\": \"\",
  \"instruction\": \"README.md\",
  \"version\": \"3.0.1\",
  \"description\": \"DEPRECATED Neo4j is the world's leading graph database\",
  \"source\": \"\",
  \"icon_url\": \"https://info.neo4j.com/rs/773-GON-065/images/neo4j_logo.png\",
  \"public\": false,
  \"chart_url\": \"neo4j-3.0.1.tgz\",
  \"create_at\": \"2021-05-08T08:53:13Z\",
  \"update_at\": \"2021-05-08T08:53:13Z\"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowChartSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowChartRequest request = new ShowChartRequest();
        request.withChartId("{chart_id}");
        try {
            ShowChartResponse response = client.showChart(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```

risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]

credentials = BasicCredentials(ak, sk)

client = CceClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CceRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ShowChartRequest()
    request.chart_id = "{chart_id}"
    response = client.show_chart(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main
```

```
import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowChartRequest{}
    request.ChartId = "{chart_id}"
    response, err := client.ShowChart(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.9 Deleting a Release

Function

This API is used to delete a release.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /cce/cam/v3/clusters/{cluster_id}/namespace/{namespace}/releases/{name}

Table 4-1188 Path Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Release name
namespace	Yes	String	Namespace of the chart

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-1189 Query Parameters

Parameter	Mandatory	Type	Description
show_resources	No	String	Definition Whether to display the resource information of a release Constraints N/A Range If this parameter is set to true , the resource information of the release will be displayed. If this parameter is not specified, the resource information is not displayed by default. Default Value N/A

Request Parameters

Table 4-1190 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1191 Response body parameters

Parameter	Type	Description
chart_name	String	Chart name
chart_public	Boolean	Whether the chart is public
chart_version	String	Chart version
cluster_id	String	Cluster ID
cluster_name	String	Current name
create_at	String	Creation time.
description	String	Release description
name	String	Release name
namespace	String	Namespace to which a chart release belongs
parameters	String	Release parameters
resources	String	Resources required by the release
status	String	Release status. • DEPLOYED: The release is normal. • DELETED: The release has been deleted. • FAILED: The release fails to be deployed. • DELETING: The release is being deleted. • PENDING_INSTALL: The release is waiting to be installed. • PENDING_UPGRADE: The release is waiting to be upgraded. • PENDING_ROLLBACK: The release is waiting for rollback. • UNKNOWN: The release status is unknown, indicating that the release is abnormal. You can manually delete the release and reinstall it.
status_description	String	Release status description
update_at	String	Update time.
values	String	Release value
version	Integer	Release version

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "chart_name": "magento-mysql",
  "chart_public": false,
  "chart_version": "1.0.0",
  "cluster_id": "a870253f-5dc7-11ee-bf71-0255ac100b03",
  "cluster_name": "sfs-turbo-test",
  "create_at": "2023-11-14T20:30:57+08:00",
  "name": "testwww",
  "namespace": "monitoring",
  "parameters": "",
  "resources": "",
  "status": "DELETING",
  "update_at": "2023-11-14T20:30:57+08:00",
  "values": "{\n  \"basic\": {\n    \"admin_password\": \"*****\",\n    \"admin_username\": \"username\",\n    \"app_name\": \"magento\",\n    \"mysql_database\": \"magento\",\n    \"mysql_name\": \"mysql\",\n    \"mysql_password\": \"*****\",\n    \"mysql_port\": 3306,\n    \"mysql_root_password\": \"*****\",\n    \"mysql_user\": \"magento\",\n    \"storage_class\": \"csi-nas\",\n    \"storage_model\": \"ReadWriteMany\",\n    \"storage_size\": \"10G\",\n    \"global\": {\n      \"magento_EIP\": \"100.100.100.100\",\n      \"magento_EPORT\": 32080,\n      \"namespace\": \"default\",\n      \"image\": {\n        \"magento_image\": \"example.com/everest/magento:latest\",\n        \"mysql_image\": \"example.com/everest/mysql:5.7.14\"\n      }\n    }\n  }\n}"
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class DeleteReleaseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
```

```
.withCredential(auth)
.withRegion(CceRegion.valueOf("<YOUR REGION>"))
.build();
DeleteReleaseRequest request = new DeleteReleaseRequest();
request.withName("{name}");
request.withNamespace("{namespace}");
request.withClusterId("{cluster_id}");
try {
    DeleteReleaseResponse response = client.deleteRelease(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteReleaseRequest()
        request.name = "{name}"
        request.namespace = "{namespace}"
        request.cluster_id = "{cluster_id}"
        response = client.delete_release(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteReleaseRequest{}
    request.Name = "{name}"
    request.Namespace = "{namespace}"
    request.ClusterId = "{cluster_id}"
    response, err := client.DeleteRelease(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.10 Downloading a Chart

Function

This API is used to download a chart.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v2/charts/{chart_id}/archive

Table 4-1192 Path Parameters

Parameter	Mandatory	Type	Description
chart_id	Yes	String	Chart ID

Request Parameters

Table 4-1193 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1194 Response body parameters

Parameter	Type	Description
-	File	File name of the downloaded chart

Example Requests

None

Example Responses

Status code: 200

OK

```
"chart-file.tgz"
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;
```

```
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class DownloadChartSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        DownloadChartRequest request = new DownloadChartRequest();
        request.withChartId("{chart_id}");
        try {
            DownloadChartResponse response = client.downloadChart(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.getenv("CLOUD_SDK_AK")
    sk = os.getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
```

```
.with_credentials(credentials) \
.with_region(CceRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = DownloadChartRequest()
    request.chart_id = "{chart_id}"
    response = client.download_chart(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DownloadChartRequest{}
    request.ChartId = "{chart_id}"
    response, err := client.DownloadChart(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.11 Obtaining a Release

Function

This API is used to obtain a release.

Calling Method

For details, see [Calling APIs](#).

URI

GET /cce/cam/v3/clusters/{cluster_id}/namespace/{namespace}/releases/{name}

Table 4-1195 Path Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Release name
namespace	Yes	String	Namespace of the chart
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Table 4-1196 Query Parameters

Parameter	Mandatory	Type	Description
show_resources	No	String	Definition Whether to display the resource information of a release Constraints N/A Range If this parameter is set to true , the resource information of the release will be displayed. If this parameter is not specified, the resource information is not displayed by default. Default Value N/A

Request Parameters

Table 4-1197 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1198 Response body parameters

Parameter	Type	Description
chart_name	String	Chart name
chart_public	Boolean	Whether the chart is public
chart_version	String	Chart version
cluster_id	String	Cluster ID
cluster_name	String	Current name
create_at	String	Creation time.
description	String	Release description
name	String	Release name
namespace	String	Namespace to which a chart release belongs
parameters	String	Release parameters
resources	String	Resources required by the release

Parameter	Type	Description
status	String	Release status. • DEPLOYED: The release is normal. • DELETED: The release has been deleted. • FAILED: The release fails to be deployed. • DELETING: The release is being deleted. • PENDING_INSTALL: The release is waiting to be installed. • PENDING_UPGRADE: The release is waiting to be upgraded. • PENDING_ROLLBACK: The release is waiting for rollback. • UNKNOWN: The release status is unknown, indicating that the release is abnormal. You can manually delete the release and reinstall it.
status_description	String	Release status description
update_at	String	Update time.
values	String	Release value
version	Integer	Release version

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "chart_name": "magento-mysql",
  "chart_public": false,
  "chart_version": "1.0.0",
  "cluster_id": "a870253f-5dc7-11ee-bf71-0255ac100b03",
  "cluster_name": "sfs-turbo-test",
  "create_at": "2023-11-14T20:30:57+08:00",
  "description": "Initial install underway",
  "name": "testwww",
  "namespace": "monitoring",
  "parameters": "",
  "resources": "",
  "status": "PENDING_INSTALL",
  "status_description": "Initial install underway",
}
```

```
"update_at" : "2023-11-14T20:30:57+08:00",
"values" : "{ \"basic\": { \"admin_password\": \"*****\", \"admin_username\": \"username\", \"app_name\": \"magento\", \"mysql_database\": \"magento\", \"mysql_name\": \"mysql\", \"mysql_password\": \"*****\", \"mysql_port\": 3306, \"mysql_root_password\": \"*****\", \"mysql_user\": \"magento\", \"storage_class\": \"csi-nas\", \"storage_mode\": \"ReadWriteMany\", \"storage_size\": \"10G\", \"global\": { \"magento_EIP\": \"100.100.100.100\", \"magento_EPORT\": 32080, \"namespace\": \"default\", \"image\": { \"magento_image\": \"example.com/everest/magento:latest\", \"mysql_image\": \"example.com/everest/mysql:5.7.14\" } } }, \"version\" : 1 } }
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowReleaseSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowReleaseRequest request = new ShowReleaseRequest();
        request.setName("{name}");
        request.withNamespace("{namespace}");
        request.withClusterId("{cluster_id}");
        try {
            ShowReleaseResponse response = client.showRelease(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowReleaseRequest()
        request.name = "{name}"
        request.namespace = "{namespace}"
        request.cluster_id = "{cluster_id}"
        response = client.show_release(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```

```
Build()  
  
request := &model.ShowReleaseRequest{  
    request.Name = "{name}"  
    request.Namespace = "{namespace}"  
    request.ClusterId = "{cluster_id}"  
    response, err := client.ShowRelease(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.12 Obtaining Chart Values

Function

This API is used to obtain chart values.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v2/charts/{chart_id}/values

Table 4-1199 Path Parameters

Parameter	Mandatory	Type	Description
chart_id	Yes	String	Chart ID

Request Parameters

Table 4-1200 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1201 Response body parameters

Parameter	Type	Description
values	Map<String,Object>	Data in values.yaml . The data structure depends on the chart.

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "values": {
    "basic": {
      "admin_password": "*****",
      "admin_username": "username"
    },
    "global": {
      "magento_EIP": "127.0.0.1",
      "magento_EPORT": 32080,
      "namespace": "demo"
    },
    "image": {
      "magento_image": "example.com/demo/magento:latest",
      "mysql_image": "example.com/demo/mysql:5.7.14"
    }
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowChartValuesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```



```
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

CceClient client = CceClient.newBuilder()
    .withCredential(auth)
    .withRegion(CceRegion.valueOf("<YOUR REGION>"))
    .build();
ShowChartValuesRequest request = new ShowChartValuesRequest();
request.withChartId("{chart_id}");
try {
    ShowChartValuesResponse response = client.showChartValues(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowChartValuesRequest()
        request.chart_id = "{chart_id}"
        response = client.show_chart_values(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowChartValuesRequest{}
    request.ChartId = "{chart_id}"
    response, err := client.ShowChartValues(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.13 Obtaining Historical Records of a Release

Function

This API is used to obtain historical records of a release.

Calling Method

For details, see [Calling APIs](#).

URI

GET /cce/cam/v3/clusters/{cluster_id}/namespace/{namespace}/releases/{name}/history

Table 4-1202 Path Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Release name
namespace	Yes	String	Namespace of the chart
cluster_id	Yes	String	Details: Cluster ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Cluster IDs Default value: N/A

Request Parameters

Table 4-1203 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1204 Response body parameters

Parameter	Type	Description
[items]	Array of ReleaseResp objects	

Table 4-1205 ReleaseResp

Parameter	Type	Description
chart_name	String	Chart name
chart_public	Boolean	Whether the chart is public
chart_version	String	Chart version
cluster_id	String	Cluster ID
cluster_name	String	Current name
create_at	String	Creation time.
description	String	Release description
name	String	Release name
namespace	String	Namespace to which a chart release belongs
parameters	String	Release parameters
resources	String	Resources required by the release

Parameter	Type	Description
status	String	Release status. • DEPLOYED: The release is normal. • DELETED: The release has been deleted. • FAILED: The release fails to be deployed. • DELETING: The release is being deleted. • PENDING_INSTALL: The release is waiting to be installed. • PENDING_UPGRADE: The release is waiting to be upgraded. • PENDING_ROLLBACK: The release is waiting for rollback. • UNKNOWN: The release status is unknown, indicating that the release is abnormal. You can manually delete the release and reinstall it.
status_description	String	Release status description
update_at	String	Update time.
values	String	Release value
version	Integer	Release version

Example Requests

None

Example Responses

Status code: 200

OK

```
[ {  
  "chart_name": "magento-mysql",  
  "chart_public": false,  
  "chart_version": "1.0.0",  
  "cluster_id": "a870253f-5dc7-11ee-bf71-0255ac100b03",  
  "cluster_name": "sfs-turbo-test",  
  "create_at": "2023-11-14T20:30:57+08:00",  
  "description": "Initial install underway",  
  "name": "testwww",  
  "namespace": "monitoring",  
  "parameters": "",  
  "resources": "",  
  "status": "PENDING_INSTALL",  
  "status_description": "Initial install underway",  
}
```

```
"update_at" : "2023-11-14T20:30:57+08:00",
"values" : "{ \"basic\": { \"admin_password\": \"*****\", \"admin_username\": \"username\", \"app_name\": \"magento\", \"mysql_database\": \"magento\", \"mysql_name\": \"mysql\", \"mysql_password\": \"*****\", \"mysql_port\": 3306, \"mysql_root_password\": \"*****\", \"mysql_user\": \"magento\", \"storage_class\": \"csi-nas\", \"storage_mode\": \"ReadWriteMany\", \"storage_size\": \"10G\" }, \"global\": { \"magento_EIP\": \"100.100.100.100\", \"magento_EPORT\": 32080, \"namespace\": \"default\", \"image\": \"magento_image\", \"example.com/everest/magento:latest\", \"mysql_image\": \"example.com/everest/mysql:5.7.14\" } }\",
\"version\" : 1
} ]
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowReleaseHistorySolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();

        ShowReleaseHistoryRequest request = new ShowReleaseHistoryRequest();
        request.setName("{name}");
        request.withNamespace("{namespace}");
        request.withClusterId("{cluster_id}");
        try {
            ShowReleaseHistoryResponse response = client.showReleaseHistory(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowReleaseHistoryRequest()
        request.name = "{name}"
        request.namespace = "{namespace}"
        request.cluster_id = "{cluster_id}"
        response = client.show_release_history(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := cce.NewCceClient(
        cce.CceClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```



```
Build()  
  
request := &model.ShowReleaseHistoryRequest{}  
request.Name = "{name}"  
request.Namespace = "{namespace}"  
request.ClusterId = "{cluster_id}"  
response, err := client.ShowReleaseHistory(request)  
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.12.14 Obtaining the Quota of a User Chart

Function

This API is used to obtain the quota of a user chart.

Calling Method

For details, see [Calling APIs](#).

URI

GET /v2/charts/{project_id}/quotas

Table 4-1206 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Details: Project ID. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: None Options: Project IDs of the account Default value: N/A

Request Parameters

Table 4-1207 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Details: The request body type or format Constraints: The GET method is not verified. Options: • application/json • application/json;charset=utf-8 • application/x-pem-file • multipart/form-data (used when the FormData parameter is present) Default value: N/A

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Details: Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details, see Obtaining a User Token . Constraints: None Options: N/A Default value: N/A

Response Parameters

Status code: 200

Table 4-1208 Response body parameters

Parameter	Type	Description
quotas	quotas object	Chart quota

Table 4-1209 quotas

Parameter	Type	Description
resources	Array of resources objects	Resources

Table 4-1210 resources

Parameter	Type	Description
type	String	Resource type
quota	Integer	Resource quota
used	Integer	Used resources

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "quotas" : {
    "resources" : [ {
      "type" : "Charts",
      "quota" : 200,
      "used" : 2
    } ]
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cce.v3.region.CceRegion;
import com.huaweicloud.sdk.cce.v3.*;
import com.huaweicloud.sdk.cce.v3.model.*;

public class ShowUserChartsQuotasSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
            .withProjectId(projectId)
            .withAk(ak)
            .withSk(sk);

        CceClient client = CceClient.newBuilder()
            .withCredential(auth)
            .withRegion(CceRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowUserChartsQuotasRequest request = new ShowUserChartsQuotasRequest();
        try {
            ShowUserChartsQuotasResponse response = client.showUserChartsQuotas(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        }
    }
}
```

```
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcce.v3.region.cce_region import CceRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcce.v3 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CceClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CceRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowUserChartsQuotasRequest()
        response = client.show_user_charts_quotas(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cce "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cce/v3/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    WithProjectId(projectId).
    Build()

client := cce.NewCceClient(
    cce.CceClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowUserChartsQuotasRequest{}
response, err := client.ShowUserChartsQuotas(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in [API Explorer](#). SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.13 Permissions Management

4.13.1 Creating an Access Policy

Function

This API is used to create an access policy.

Calling Method

For details, see [Calling APIs](#).

URI

POST /api/v3/access-policies

Request Parameters

Table 4-1211 Request body parameters

Parameter	Mandatory	Type	Description
kind	No	String	Definition: API type Constraints: The value cannot be changed. Range: N/A Default Value: AccessPolicy
apiVersion	No	String	Definition: API version Constraints: The value cannot be changed. Range: N/A Default Value: v3
name	No	String	Definition: Access policy name Constraints: Enter 1 to 56 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range: N/A Default Value: N/A

Parameter	Mandatory	Type	Description
clusters	Yes	Array of strings	Definition: List of cluster IDs. Wildcards (*) are allowed, which indicate all clusters. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: A maximum of 200 clusters can be authorized at a time. Range: [*] or cluster ID list Default Value: N/A
accessScope	Yes	AccessScope object	Definition: Access scope, which is used to specify the cluster and namespace to be authorized Constraints: N/A
policyType	Yes	String	Definition: Permission type Constraints: N/A Range: • CCEAdminPolicy: administrator permissions • CCEClusterAdminPolicy: O&M permissions • CCEEditPolicy: development permissions • CCEViewPolicy: read-only permissions Default Value: N/A
principal	Yes	Principal object	Definition: Authorization object Constraints: N/A

Table 4-1212 AccessScope

Parameter	Mandatory	Type	Description
namespaces	Yes	Array of strings	Definition List of cluster namespaces. Wildcards (*) are allowed to indicate all namespaces. If different clusters are selected, the namespace list can be a collection of multiple clusters. When RBAC authorization is used, CCE automatically checks whether the namespaces exist in the clusters. Constraints A maximum of 500 namespaces can be authorized at a time. Range [*] or cluster namespace list Default Value N/A

Table 4-1213 Principal

Parameter	Mandatory	Type	Description
type	Yes	String	Definition: Type of the authorization object Constraints: N/A Range: • user: user • group: user group • agency: agency account Default Value: N/A

Parameter	Mandatory	Type	Description
ids	Yes	Array of strings	Definition: List of IDs of authorized objects. Enter the IDs based on the object type, user, user group, and agency account. Constraints: A maximum of 500 users or user groups can be authorized at a time. Range: N/A Default Value: N/A

Response Parameters

Status code: 201

Table 4-1214 Response body parameters

Parameter	Type	Description
kind	String	Definition: API type Constraints: The value cannot be changed. Range: N/A Default Value: AccessPolicy
apiVersion	String	Definition: API version Constraints: The value cannot be changed. Range: N/A Default Value: v3

Parameter	Type	Description
name	String	Definition: Access policy name Constraints: Enter 1 to 56 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range: N/A Default Value: N/A
policyId	String	Definition: Permission ID Constraints: The value is automatically generated and cannot be changed. Range: N/A Default Value: N/A
clusters	Array of strings	Definition: List of cluster IDs. Wildcards (*) are allowed, which indicate all clusters. Constraints: A maximum of 200 clusters can be authorized at a time. Range: [*] or cluster ID list Default Value: N/A
accessScope	AccessScope object	Definition: Access scope, which is used to specify the cluster and namespace to be authorized Constraints: N/A

Parameter	Type	Description
policyType	String	Definition: Permission type Constraints: N/A Range: • CCEAdminPolicy: administrator permissions • CCEClusterAdminPolicy: O&M permissions • CCEEditPolicy: development permissions • CCEViewPolicy: read-only permissions Default Value: N/A
principal	Principal object	Definition: Authorization object Constraints: N/A
createTime	String	Definition: Creation time Constraints: N/A Range: N/A Default Value: N/A
updateTime	String	Definition: Update time Constraints: N/A Range: N/A Default Value: N/A

Table 4-1215 AccessScope

Parameter	Type	Description
namespaces	Array of strings	Definition List of cluster namespaces. Wildcards (*) are allowed to indicate all namespaces. If different clusters are selected, the namespace list can be a collection of multiple clusters. When RBAC authorization is used, CCE automatically checks whether the namespaces exist in the clusters. Constraints A maximum of 500 namespaces can be authorized at a time. Range [*] or cluster namespace list Default Value N/A

Table 4-1216 Principal

Parameter	Type	Description
type	String	Definition: Type of the authorization object Constraints: N/A Range: • user: user • group: user group • agency: agency account Default Value: N/A

Parameter	Type	Description
ids	Array of strings	Definition: List of IDs of authorized objects. Enter the IDs based on the object type, user, user group, and agency account. Constraints: A maximum of 500 users or user groups can be authorized at a time. Range: N/A Default Value: N/A

Example Requests

Create an access policy.

POST /api/v3/access-policies

```
{
  "kind": "AccessPolicy",
  "apiVersion": "v3",
  "name": "test-access-policy",
  "accessScope": {
    "clusters": [ "*" ],
    "namespaces": [ "*" ]
  },
  "policy": "CCEAdminPolicy",
  "principal": {
    "type": "user",
    "ids": [ "069fcc2116c347b89869eae3cdxxxxx1", "069fcc2116c347b89869eae3cdxxxxx2" ]
  }
}
```

Example Responses

Status code: 201

```
{
  "kind": "AccessPolicy",
  "apiVersion": "v3",
  "name": "test-access-policy",
  "policyId": "402358e8-2e3a-4531-bae7-fe9cba14d0d1",
  "clusters": [ "*" ],
  "accessScope": {
    "namespaces": [ "*" ]
  },
  "policy": "CCEAdminPolicy",
  "principal": {
    "type": "user",
    "ids": [ "069fcc2116c347b89869eae3cdxxxxx1", "069fcc2116c347b89869eae3cdxxxxx2" ]
  },
  "createTime": "",
  "updateTime": ""
}
```

Status Codes

Status Code	Description
201	

Error Codes

See [Error Codes](#).

4.13.2 Obtaining the Access Policy List

Function

This API is used to obtain the access policy list.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/access-policies

Table 4-1217 Query Parameters

Parameter	Mandatory	Type	Description
cluster_id	No	String	Definition: Cluster ID. Only actions related to the cluster are returned. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: N/A Range: N/A Default Value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-1218 Response body parameters

Parameter	Type	Description
kind	String	Definition: API type Constraints: The value cannot be changed. Range: N/A Default Value: List
apiVersion	String	Definition: API version Constraints: The value cannot be changed. Range: N/A Default Value: v3
accessPolicyList	Array of AccessPolicyResp objects	

Table 4-1219 AccessPolicyResp

Parameter	Type	Description
kind	String	Definition: API type Constraints: The value cannot be changed. Range: N/A Default Value: AccessPolicy

Parameter	Type	Description
apiVersion	String	Definition: API version Constraints: The value cannot be changed. Range: N/A Default Value: v3
name	String	Definition: Access policy name Constraints: Enter 1 to 56 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range: N/A Default Value: N/A
policyId	String	Definition: Permission ID Constraints: The value is automatically generated and cannot be changed. Range: N/A Default Value: N/A
clusters	Array of strings	Definition: List of cluster IDs. Wildcards (*) are allowed, which indicate all clusters. Constraints: A maximum of 200 clusters can be authorized at a time. Range: [*] or cluster ID list Default Value: N/A

Parameter	Type	Description
accessScope	AccessScope object	Definition: Access scope, which is used to specify the cluster and namespace to be authorized Constraints: N/A
policyType	String	Definition: Permission type Constraints: N/A Range: • CCEAdminPolicy: administrator permissions • CCEClusterAdminPolicy: O&M permissions • CCEEditPolicy: development permissions • CCEViewPolicy: read-only permissions Default Value: N/A
principal	Principal object	Definition: Authorization object Constraints: N/A
createTime	String	Definition: Creation time Constraints: N/A Range: N/A Default Value: N/A

Parameter	Type	Description
updateTime	String	Definition: Update time Constraints: N/A Range: N/A Default Value: N/A

Table 4-1220 AccessScope

Parameter	Type	Description
namespaces	Array of strings	Definition List of cluster namespaces. Wildcards (*) are allowed to indicate all namespaces. If different clusters are selected, the namespace list can be a collection of multiple clusters. When RBAC authorization is used, CCE automatically checks whether the namespaces exist in the clusters. Constraints A maximum of 500 namespaces can be authorized at a time. Range [*] or cluster namespace list Default Value N/A

Table 4-1221 Principal

Parameter	Type	Description
type	String	Definition: Type of the authorization object Constraints: N/A Range: • user: user • group: user group • agency: agency account Default Value: N/A
ids	Array of strings	Definition: List of IDs of authorized objects. Enter the IDs based on the object type, user, user group, and agency account. Constraints: A maximum of 500 users or user groups can be authorized at a time. Range: N/A Default Value: N/A

Example Requests

None

Example Responses

Status code: 200

```
kind: List
apiVersion: v3
accessPolicyList:
- kind: AccessPolicy
  apiVersion: v3
  name: test-access-policy
  policyId: 402358e8-2e3a-4531-bae7-fe9cba14d0d1
  clusters:
    - "*"
  accessScope:
    namespaces:
      - "*"
  policy: CCEAdminPolicy
  principal:
    type: user
    ids:
```

```
- '069fcc2116c347b89869eae3cdxxxxx1'  
- '069fcc2116c347b89869eae3cdxxxxx2'  
createTime: "  
updateTime: "
```

Status Codes

Status Code	Description
200	

Error Codes

See [Error Codes](#).

4.13.3 Obtaining Access Policy Details

Function

This API is used to obtain an access policy.

Calling Method

For details, see [Calling APIs](#).

URI

GET /api/v3/access-policies/{policy_id}

Table 4-1222 Path Parameters

Parameter	Mandatory	Type	Description
policy_id	Yes	String	Definition: Access policy ID. For details about how to obtain the access policy ID, see Obtaining the Access Policy List . Constraints: N/A Range: N/A Default Value: N/A

Request Parameters

None

Response Parameters

Status code: 200

Table 4-1223 Response body parameters

Parameter	Type	Description
kind	String	Definition: API type Constraints: The value cannot be changed. Range: N/A Default Value: AccessPolicy
apiVersion	String	Definition: API version Constraints: The value cannot be changed. Range: N/A Default Value: v3
name	String	Definition: Access policy name Constraints: Enter 1 to 56 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range: N/A Default Value: N/A

Parameter	Type	Description
policyId	String	Definition: Permission ID Constraints: The value is automatically generated and cannot be changed. Range: N/A Default Value: N/A
clusters	Array of strings	Definition: List of cluster IDs. Wildcards (*) are allowed, which indicate all clusters. Constraints: A maximum of 200 clusters can be authorized at a time. Range: [*] or cluster ID list Default Value: N/A
accessScope	AccessScope object	Definition: Access scope, which is used to specify the cluster and namespace to be authorized Constraints: N/A
policyType	String	Definition: Permission type Constraints: N/A Range: • CCEAdminPolicy: administrator permissions • CCEClusterAdminPolicy: O&M permissions • CCEEditPolicy: development permissions • CCEViewPolicy: read-only permissions Default Value: N/A

Parameter	Type	Description
principal	Principal object	Definition: Authorization object Constraints: N/A
createTime	String	Definition: Creation time Constraints: N/A Range: N/A Default Value: N/A
updateTime	String	Definition: Update time Constraints: N/A Range: N/A Default Value: N/A

Table 4-1224 AccessScope

Parameter	Type	Description
namespaces	Array of strings	Definition List of cluster namespaces. Wildcards (*) are allowed to indicate all namespaces. If different clusters are selected, the namespace list can be a collection of multiple clusters. When RBAC authorization is used, CCE automatically checks whether the namespaces exist in the clusters. Constraints A maximum of 500 namespaces can be authorized at a time. Range [*] or cluster namespace list Default Value N/A

Table 4-1225 Principal

Parameter	Type	Description
type	String	Definition: Type of the authorization object Constraints: N/A Range: • user: user • group: user group • agency: agency account Default Value: N/A

Parameter	Type	Description
ids	Array of strings	Definition: List of IDs of authorized objects. Enter the IDs based on the object type, user, user group, and agency account. Constraints: A maximum of 500 users or user groups can be authorized at a time. Range: N/A Default Value: N/A

Example Requests

None

Example Responses

Status code: 200

```
{
  "kind" : "AccessPolicy",
  "apiVersion" : "v3",
  "name" : "test-access-policy",
  "policyId" : "402358e8-2e3a-4531-bae7-fe9cba14d0d1",
  "clusters" : [ "*" ],
  "accessScope" : {
    "namespaces" : [ "*" ]
  },
  "policy" : "CCEAdminPolicy",
  "principal" : {
    "type" : "user",
    "ids" : [ "069fcc2116c347b89869eae3cdxxxxx1", "069fcc2116c347b89869eae3cdxxxxx2" ]
  },
  "createTime" : "",
  "updateTime" : ""
}
```

Status Codes

Status Code	Description
200	

Error Codes

See [Error Codes](#).

4.13.4 Updating an Access Policy

Function

This API is used to update an access policy.

Calling Method

For details, see [Calling APIs](#).

URI

PUT /api/v3/access-policies/{policy_id}

Table 4-1226 Path Parameters

Parameter	Mandatory	Type	Description
policy_id	Yes	String	Definition: Access policy ID. For details about how to obtain the access policy ID, see Obtaining the Access Policy List . Constraints: N/A Range: N/A Default Value: N/A

Request Parameters

Table 4-1227 Request body parameters

Parameter	Mandatory	Type	Description
kind	No	String	Definition: API type Constraints: The value cannot be changed. Range: N/A Default Value: AccessPolicy

Parameter	Mandatory	Type	Description
apiVersion	No	String	Definition: API version Constraints: The value cannot be changed. Range: N/A Default Value: v3
name	No	String	Definition: Access policy name Constraints: Enter 1 to 56 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range: N/A Default Value: N/A
clusters	Yes	Array of strings	Definition: List of cluster IDs. Wildcards (*) are allowed, which indicate all clusters. For details about how to obtain the value, see How to Obtain Parameters in the API URI . Constraints: A maximum of 200 clusters can be authorized at a time. Range: [*] or cluster ID list Default Value: N/A
accessScope	Yes	AccessScope object	Definition: Access scope, which is used to specify the cluster and namespace to be authorized Constraints: N/A

Parameter	Mandatory	Type	Description
policyType	Yes	String	Definition: Permission type Constraints: N/A Range: • CCEAdminPolicy: administrator permissions • CCEClusterAdminPolicy: O&M permissions • CCEEditPolicy: development permissions • CCEViewPolicy: read-only permissions Default Value: N/A
principal	Yes	Principal object	Definition: Authorization object Constraints: N/A

Table 4-1228 AccessScope

Parameter	Mandatory	Type	Description
namespaces	Yes	Array of strings	Definition List of cluster namespaces. Wildcards (*) are allowed to indicate all namespaces. If different clusters are selected, the namespace list can be a collection of multiple clusters. When RBAC authorization is used, CCE automatically checks whether the namespaces exist in the clusters. Constraints A maximum of 500 namespaces can be authorized at a time. Range [*] or cluster namespace list Default Value N/A

Table 4-1229 Principal

Parameter	Mandatory	Type	Description
type	Yes	String	Definition: Type of the authorization object Constraints: N/A Range: • user: user • group: user group • agency: agency account Default Value: N/A

Parameter	Mandatory	Type	Description
ids	Yes	Array of strings	Definition: List of IDs of authorized objects. Enter the IDs based on the object type, user, user group, and agency account. Constraints: A maximum of 500 users or user groups can be authorized at a time. Range: N/A Default Value: N/A

Response Parameters

Status code: 200

Table 4-1230 Response body parameters

Parameter	Type	Description
kind	String	Definition: API type Constraints: The value cannot be changed. Range: N/A Default Value: AccessPolicy
apiVersion	String	Definition: API version Constraints: The value cannot be changed. Range: N/A Default Value: v3

Parameter	Type	Description
name	String	Definition: Access policy name Constraints: Enter 1 to 56 characters starting with a lowercase letter and not ending with a hyphen (-). Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. Range: N/A Default Value: N/A
policyId	String	Definition: Permission ID Constraints: The value is automatically generated and cannot be changed. Range: N/A Default Value: N/A
clusters	Array of strings	Definition: List of cluster IDs. Wildcards (*) are allowed, which indicate all clusters. Constraints: A maximum of 200 clusters can be authorized at a time. Range: [*] or cluster ID list Default Value: N/A
accessScope	AccessScope object	Definition: Access scope, which is used to specify the cluster and namespace to be authorized Constraints: N/A

Parameter	Type	Description
policyType	String	Definition: Permission type Constraints: N/A Range: • CCEAdminPolicy: administrator permissions • CCEClusterAdminPolicy: O&M permissions • CCEEditPolicy: development permissions • CCEViewPolicy: read-only permissions Default Value: N/A
principal	Principal object	Definition: Authorization object Constraints: N/A
createTime	String	Definition: Creation time Constraints: N/A Range: N/A Default Value: N/A
updateTime	String	Definition: Update time Constraints: N/A Range: N/A Default Value: N/A

Table 4-1231 AccessScope

Parameter	Type	Description
namespaces	Array of strings	Definition List of cluster namespaces. Wildcards (*) are allowed to indicate all namespaces. If different clusters are selected, the namespace list can be a collection of multiple clusters. When RBAC authorization is used, CCE automatically checks whether the namespaces exist in the clusters. Constraints A maximum of 500 namespaces can be authorized at a time. Range [*] or cluster namespace list Default Value N/A

Table 4-1232 Principal

Parameter	Type	Description
type	String	Definition: Type of the authorization object Constraints: N/A Range: • user: user • group: user group • agency: agency account Default Value: N/A

Parameter	Type	Description
ids	Array of strings	Definition: List of IDs of authorized objects. Enter the IDs based on the object type, user, user group, and agency account. Constraints: A maximum of 500 users or user groups can be authorized at a time. Range: N/A Default Value: N/A

Example Requests

Update an access policy.

POST /api/v3/access-policies

```
{
  "kind" : "AccessPolicy",
  "apiVersion" : "v2",
  "name" : "test-access-policy",
  "accessScope" : {
    "clusters" : [ "*" ],
    "namespaces" : [ "*" ]
  },
  "policy" : "CCEAdminPolicy",
  "principal" : {
    "type" : "user",
    "ids" : [ "069fcc2116c347b89869eae3cdxxxxx1", "069fcc2116c347b89869eae3cdxxxxx2" ]
  }
}
```

Example Responses

Status code: 200

```
{
  "kind" : "AccessPolicy",
  "apiVersion" : "v3",
  "name" : "test-access-policy",
  "policyId" : "402358e8-2e3a-4531-bae7-fe9cba14d0d1",
  "clusters" : [ "*" ],
  "accessScope" : {
    "namespaces" : [ "*" ]
  },
  "policy" : "CCEAdminPolicy",
  "principal" : {
    "type" : "user",
    "ids" : [ "069fcc2116c347b89869eae3cdxxxxx1", "069fcc2116c347b89869eae3cdxxxxx2" ]
  },
  "createTime" : "",
  "updateTime" : ""
}
```

Status Codes

Status Code	Description
200	

Error Codes

See [Error Codes](#).

4.13.5 Deleting an Access Policy

Function

This API is used to delete an access policy.

Calling Method

For details, see [Calling APIs](#).

URI

DELETE /api/v3/access-policies/{policy_id}

Table 4-1233 Path Parameters

Parameter	Mandatory	Type	Description
policy_id	Yes	String	Definition: Access policy ID. For details about how to obtain the access policy ID, see Obtaining the Access Policy List . Constraints: N/A Range: N/A Default Value: N/A

Request Parameters

None

Response Parameters

Status code: 204

None

Example Requests

None

Example Responses

None

Status Codes

Status Code	Description
204	

Error Codes

See [Error Codes](#).

4.14 Add-on Instance Parameters

4.14.1 CoreDNS

Add-on Overview

CoreDNS is a DNS server that uses chain plug-ins to provide domain name resolution services for Kubernetes clusters. It is the recommended DNS server solution by the Kubernetes community.

Add-on Parameters

Table 4-1234 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1235 object	Flavor parameters

Parameter	Mandatory	Type	Description
custom	Yes	Table 4-1236 object	Custom parameters

Table 4-1235 Configuration of flavor

Parameter	Mandatory	Type	Description
replicas	Yes	int	Number of pods. The default value is 2.
resources	Yes	Array of resources object	Container resource (CPU and memory) quotas

Table 4-1236 Configuration of custom

Parameter	Mandatory	Type	Description
servers	No	object	Configuration of servers
stub_domains	No	Map<String>[]string	Stub domain configuration
multiAZenable	No	bool	Whether to enable the forcible mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on will be forcibly scheduled to nodes in different AZs. If there are fewer AZs than pods, the extra pods will fail to run. If both multiAZenable and multiAZBalance are set to true, the settings of multiAZBalance take effect, which means, the equivalent mode of multi-AZ deployment is used.

Parameter	Mandatory	Type	Description
multiAZBalance	No	bool	Whether to enable the equivalent mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on are evenly scheduled to the nodes in the cluster in each AZ. If a new AZ is added, it is recommended that you increase add-on pods for cross-AZ HA deployment. With the equivalent multi-AZ deployment, the difference between the number of add-on pods in different AZs will be less than or equal to 1. If resources in one of the AZs are insufficient, pods cannot be scheduled to that AZ.
tolerations	No	Array of Table 4-1238	Toleration configuration

Table 4-1237 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at coredns .
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Table 4-1238 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Table 4-1239 Configuration of server

Parameter	Mandatory	Type	Description
port	No	Int	Domain port number. The default value is 5353 .
zones	No	Array of Table 4-1241	Domain configuration
plugins	No	Array of Table 4-1240	Configuration of plugin

Table 4-1240 Configuration of plugin

Parameter	Mandatory	Type	Description
name	Yes	String	Plugin name
configBlock	No	String	Configuration of plugin
parameters	No	String/Int	Extended parameters of the plugin

Table 4-1241 Configuration of zone

Parameter	Mandatory	Type	Description
zone	Yes	String	Domain to be listened on. The default value is a period (.).

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "2292498e-*****-0255ac1001ba",
    "version": "1.29.2",
    "addonTemplateName": "coredns",
    "values": {
      "basic": {
        "basickey": "val"
      },
      "flavor": {
        "replicas": 2,
        "resources": [{
          "limitsCpu": "2000m",
```



```
        "limitsMem": "2000Mi",
        "name": "coredns",
        "requestsCpu": "2000m",
        "requestsMem": "2000Mi"
    }]
},
"custom": {
    "multiAZBalance": false,
    "multiAZEnabled": false,
    "node_match_expressions": [],
    "servers": [{
        "plugins": [{
            "name": "bind",
            "parameters": "${POD_IP}"
        }],
        {
            "configBlock": "servfail 5s",
            "name": "cache",
            "parameters": 30
        },
        {
            "name": "errors"
        },
        {
            "name": "health",
            "parameters": "${POD_IP}:8080"
        },
        {
            "name": "ready",
            "parameters": "${POD_IP}:8081"
        },
        {
            "configBlock": "pods insecure\nfallthrough in-addr.arpa ip6.arpa",
            "name": "kubernetes",
            "parameters": "cluster.local in-addr.arpa ip6.arpa"
        },
        {
            "name": "loadbalance",
            "parameters": "round_robin"
        },
        {
            "name": "prometheus",
            "parameters": "${POD_IP}:9153"
        },
        {
            "configBlock": "policy random",
            "name": "forward",
            "parameters": ". /etc/resolv.conf"
        },
        {
            "name": "reload"
        }
    ]],
    "port": 5353,
    "zones": [{
        "zone": "."
    }]
}],
"stub_domains": {
},
"tolerations": [{
    "key": "node.kubernetes.io/not-ready",
    "operator": "Exists",
    "effect": "NoExecute",
    "tolerationSeconds": 60
},
{
    "key": "node.kubernetes.io/unreachable",
    "operator": "Exists",
```

```
    "effect": "NoExecute",  
    "tolerationSeconds": 60  
  }  
}  
}
```

4.14.2 CCE Container Storage (Everest)

Add-on Overview

The CCE Container Storage (Everest) add-on uses the Container Storage Interface (CSI) to link Kubernetes clusters with cloud storage services.

Add-on Parameters

Table 4-1242 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1243 object	Flavor parameters
custom	Yes	Table 4-1244 object	Custom parameters

Table 4-1243 Configuration of flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description
name	Yes	String	Add-on specification name
replicas	Yes	String	Number of pods. The default value is 1 .
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1244 Configuration of custom

Parameter	Mandatory	Type	Description
default_vpc_id	Yes	String	VPC ID

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID
cluster_name	Yes	String	Cluster name
project_id	Yes	String	Project ID
disable_auto_mount_secret	No	bool	Whether to allow the default AK/SK to be used during OBS volume mounting Default value: false
over_subscription	No	String	Local PV overcommitment ratio Default value: 80
csi_attach_detach_worker_threads	No	String	The number of concurrent workers for detaching volumes Default value: 60
volume_attaching_flow_ctrl	No	String	Attaching flow control data Default value: 0 .
number_of_reserved_disks	No	String	Disk attaching capability reserved for non-container scenarios Default value: 6
flow_control	No	Map<String>string	Flow control parameter Default value: {}
enable_node_attacher	No	bool	Whether to start agent attacher Default value: false
multiAZEnabled	No	bool	Whether to enable the forcible mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on will be forcibly scheduled to nodes in different AZs. If there are fewer AZs than pods, the extra pods will fail to run. If both multiAZEnable and multiAZBalance are set to true, the settings of multiAZBalance take effect, which means, the equivalent mode of multi-AZ deployment is used.

Parameter	Mandatory	Type	Description
multiAZBalance	No	bool	Whether to enable the equivalent mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on are evenly scheduled to the nodes in the cluster in each AZ. If a new AZ is added, it is recommended that you increase add-on pods for cross-AZ HA deployment. With the equivalent multi-AZ deployment, the difference between the number of add-on pods in different AZs will be less than or equal to 1. If resources in one of the AZs are insufficient, pods cannot be scheduled to that AZ.
tolerations	No	Array of Table 4-1246	Toleration configuration
node_match_expressions	No	Array of Table 4-1247	Add-on pod affinity configuration

Table 4-1245 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at everest-csi-controller or everest-csi-driver .
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Table 4-1246 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy

Parameter	Mandatory	Type	Description
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Table 4-1247 nodeMatchExpression node affinity

Parameter	Mandatory	Type	Description
key	No	String	Taint key
values	No	List<String >	Node affinity name
operator	No	String	Operator

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": { "annotations": { "addon.install/type": "install" } },
  "spec": {
    "clusterID": "cea63ae5-df9b-11ee-9f27-0255ac1001b2",
    "version": "2.4.4",
    "addonTemplateName": "everest",
    "values": {
      "basic": {
        "bms_url": "bms.***.com",
        "driver_init_image_version": "2.4.4",
        "ecsEndpoint": "https://ecs.***.com",
        "everest_image_version": "2.4.4",
        "evs_url": "evs.***.com",
        "iam_url": "iam.***.com",
        "ims_url": "ims.***.com",
        "obs_url": "obs.***.com",
        "platform": "linux-amd64",
        "sfs30_url": "obs.***.com",
        "sfs_turbo_url": "sfs-turbo.***.com",
        "sfs_url": "sfs.***.com",
        "supportHcs": false,
        "swr_addr": "swr.***.com",
        "swr_user": "hwofficial",
        "rbac_enabled": true,
        "cluster_version": "v1.23"
      },
      "flavor": {
        "description": "High available",
        "is_default": true,
        "name": "HA50",
        "recommend_cluster_flavor_types": ["small"],
        "replicas": 2,
        "resources": [
          {
            "limitsCpu": "250m",
            "limitsMem": "600Mi",
            "name": "everest-csi-controller",
            "requestsCpu": "250m",
            "requestsMem": "600Mi"
          }
        ]
      }
    }
  }
}
```

```
{
  "limitsCpu": "300m",
  "limitsMem": "300Mi",
  "name": "everest-csi-driver",
  "requestsCpu": "300m",
  "requestsMem": "300Mi"
},
"category": ["CCE", "Turbo"]
},
"custom": {
  "annotations": {},
  "cluster_id": "",
  "cluster_name": "",
  "csi_attacher_detach_worker_threads": "60",
  "csi_attacher_worker_threads": "60",
  "default_vpc_id": "",
  "disable_auto_mount_secret": false,
  "enable_node_attacher": false,
  "flow_control": {},
  "multiAZBalance": false,
  "multiAZEnabled": false,
  "node_match_expressions": [],
  "number_of_reserved_disks": "6",
  "over_subscription": "80",
  "project_id": "",
  "tolerations": [
    {
      "key": "node.kubernetes.io/not-ready",
      "operator": "Exists",
      "effect": "NoExecute",
      "tolerationSeconds": 60
    },
    {
      "key": "node.kubernetes.io/unreachable",
      "operator": "Exists",
      "effect": "NoExecute",
      "tolerationSeconds": 60
    }
  ],
  "volume_attaching_flow_ctrl": "0"
}
}
```

4.14.3 CCE Node Problem Detector

Add-on Overview

CCE Node Problem Detector (node-problem-detector, NPD) is an add-on that monitors abnormal events of cluster nodes and connects to a third-party monitoring platform. It is a daemon running on each node. It collects node issues from different daemons and reports them to the API server. It can run as a DaemonSet or a daemon.

Add-on Parameters

Table 4-1248 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1249 object	Flavor parameters
custom	Yes	Table 4-1250 object	Custom parameters

Table 4-1249 Configuration of flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description
name	Yes	String	Add-on specification name. The value is fixed at Single-instance .
replicas	Yes	String	Number of pods. The default value is 1 .
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1250 Configuration of custom

Parameter	Mandatory	Type	Description
feature_gate	No	String	Feature gate, which is used to enable the beta features
multiAZBalance	No	Bool	Multi AZ deployment
multiAZEnabled	No	Bool	Whether to deploy the add-on pods in multiple AZs. The default value is false . If this parameter is set to true , cross-AZ deployment is forcibly performed. If this parameter is set to false , cross-AZ deployment is preferred.

Parameter	Mandatory	Type	Description
npc	Yes	object Table 4-1252	node-problem-controller configuration
tolerations	No	List<Object> Table 4-1254	Tolerations of the add-on
node_match_expressions	No	List<Object> Table 4-1254	Node affinity configuration of the add-on

Table 4-1251 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at custom-resources .
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Table 4-1252 Data structure of the npc field

Parameter	Mandatory	Type	Description
maxTaintedNode	Yes	String or Int	The maximum number of nodes that NPC can add taints to when a single fault occurs on multiple nodes for minimizing impact. The value can be in int or percentage format.

Table 4-1253 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy

Parameter	Mandatory	Type	Description
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Table 4-1254 nodeMatchExpression node affinity

Parameter	Mandatory	Type	Description
key	No	String	Taint key
values	No	List<String>	Node affinity name
operator	No	String	Operator

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "b78fb690-b82c-11ee-83cf-0255ac100b0f",
    "version": "1.18.48",
    "addonTemplateName": "npd",
    "values": {
      "basic": {
        "image_version": "1.18.48",
        "swr_addr": "****",
        "swr_user": "****",
        "rbac_enabled": true,
        "cluster_version": "v1.23"
      }
    },
    "flavor": {
      "description": "custom resources",
      "name": "custom-resources",
      "replicas": 2,
      "resources": [
        {
          "limitsCpu": "100m",
          "limitsMem": "300Mi",
          "name": "node-problem-controller",
          "requestsCpu": "30m",
          "requestsMem": "100Mi"
        },
        {
          "limitsCpu": "100m",
          "limitsMem": "300Mi",
          "name": "node-problem-detector",
          "requestsCpu": "30m",
          "requestsMem": "100Mi"
        }
      ]
    },
    "category": [
```

```
"CCE",
"Turbo"
]
},
"custom": {
  "annotations": {},
  "common": {},
  "feature_gates": "",
  "multiAZBalance": false,
  "multiAZEnabled": false,
  "node_match_expressions": [],
  "npc": {
    "maxTaintedNode": "10%"
  },
  "tolerations": [
    {
      "key": "node.kubernetes.io/not-ready",
      "operator": "Exists",
      "effect": "NoExecute",
      "tolerationSeconds": 60
    },
    {
      "key": "node.kubernetes.io/unreachable",
      "operator": "Exists",
      "effect": "NoExecute",
      "tolerationSeconds": 60
    }
  ]
}
}
```

4.14.4 Kubernetes Dashboard

Add-on Overview

Kubernetes Dashboard is a general purpose, web-based UI designed for managing Kubernetes clusters. With this tool, you can easily manage applications running within the clusters, troubleshoot issues, and even run commands to manage the clusters themselves.

Add-on Parameters

Table 4-1255 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1256 object	Flavor parameters
custom	Yes	Table 4-1257 object	Custom parameters

Table 4-1256 Configuration of flavor

Parameter	Mandatory	Type	Description
replicas	Yes	String	Number of pods. The default value is 1 .
resources	Yes	Array of resources object	Container resource (CPU and memory) quotas

Table 4-1257 Configuration of custom

Parameter	Mandatory	Type	Description
serviceType	Yes	String	External access type. The value can be NodePort or ELB and defaults to NodePort .
port	No	int	Port number of the kubernetes-dashboard Service. The default value is 8443 .
loadBalancerIP	No	String	IP address of the ELB when external access type is ELB.
elbClass	No	String	ELB type when external access type is ELB. The value can be union (shared load balancer) or performance (dedicated load balancer) and defaults to union .
elbID	No	String	ID of the ELB when external access type is ELB.
certUploaded	No	bool	Whether to use a custom certificate. The default value is true .
cert	No	String	Cert of a custom certificate
key	No	String	Key of a custom certificate

Table 4-1258 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at dashboard .

Parameter	Mandatory	Type	Description
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "2292498e-d169-**-9151-***1001ba",
    "version": "2.2.27",
    "addonTemplateName": "dashboard",
    "values": {
      "basic": {
        "basickey": "val"
      },
      "flavor": {
        "replicas": 1,
        "resources": [{
          "limitsCpu": "100m",
          "limitsMem": "512Mi",
          "name": "dashboard",
          "requestsCpu": "50m",
          "requestsMem": "256Mi"
        }]
      },
      "custom": {
        "cert": "****",
        "certUploaded": true,
        "elbClass": "union",
        "elbID": 0,
        "key": "****",
        "loadBalancerIP": "",
        "port": 8443,
        "serviceType": "NodePort",
        "cluster_id": "2292498e-d169-**-9151-***1001ba",
        "tenant_id": "*****"
      }
    }
  }
}
```

4.14.5 CCE Cluster Autoscaler

Add-on Overview

The CCE Cluster Autoscaler (autoscaler) add-on enables node pools to be scaled in or out.

Add-on Parameters

Table 4-1259 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1260 object	Flavor parameters
custom	Yes	Table 4-1261 object	Custom parameters

Table 4-1260 flavor

Parameter	Mandatory	Type	Description
replicas	Yes	String	Number of pods. The default value is 2 .
resources	Yes	Array of resources object	Container resource (CPU and memory) quotas

Table 4-1261 custom

Parameter	Mandatory	Type	Description
cluster_id	Yes	string	Cluster ID.
tenant_id	Yes	string	ID of the project where the current cluster is in.
scaleDownEnabled	No	bool	Whether to enable auto scale-in. The default value is false .
scaleDownDelayAfterAdd	No	int	Cooldown period (in minutes) for starting scale-in evaluation again after auto scale-out was triggered in a cluster. The default value is 10 .
scaleDownDelayAfterDelete	No	int	Cooldown period (in minutes) for starting scale-in evaluation again after auto scale-in was triggered in a cluster. The default value is 10 .

Parameter	Mandatory	Type	Description
scaleDownDelayAfterFailure	No	int	Cooldown period (in minutes) for starting scale-in evaluation again after auto scale-in triggered by a cluster failed. The default value is 10 .
maxEmptyBulkDeleteFlag	No	int	Number of idle nodes that can be concurrently scaled in. The default value is 10 .
unremovableNodeRecheckTimeout	No	int	Interval (in minutes) for starting the checks again after a node is determined not to be scaled in. The default value is 5 .
scaleDownUtilizationThreshold	No	double	CPU and memory usage thresholds for determining whether a node can be scaled in. The default value is 0.5 .
maxNodesTotal	No	int	Maximum number of nodes that can be added to a cluster. The default value is 1000 .
coresTotal	No	int	Maximum number of CPU cores that can be added to a cluster. The default value is 32000 .
memoryTotal	No	int	Memory upper limit (in Gi) for cluster scale-out. The default value is 128000 .
scaleUpUtilizationEnabled	No	bool	Whether to enable custom scaling. The default value is true .
scaleUpUnscheduledPodEnabled	No	bool	Whether to enable automatic scale-out for unscheduled pods. The default value is true .
ignoreDaemonSetsUtilization	No	bool	Whether to ignore DaemonSets' resource usage when thresholds determine if scale-in is performed. The default value is false .
skipNodesWithCustomControllerPods	No	bool	Scale-in is not performed on the node which has containers created by a third-party controller running on it. The default value is true .
logLevel	No	int	Log level. The default value is 4 .

Parameter	Mandatory	Type	Description
multiAZEnable	No	bool	Whether to enable the forcible mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on will be forcibly scheduled to nodes in different AZs. If there are fewer AZs than pods, the extra pods will fail to run. If both multiAZEnable and multiAZBalance are set to true, the settings of multiAZBalance take effect, which means, the equivalent mode of multi-AZ deployment is used.
multiAZBalance	No	bool	Whether to enable the equivalent mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on are evenly scheduled to the nodes in the cluster in each AZ. If a new AZ is added, it is recommended that you increase add-on pods for cross-AZ HA deployment. With the equivalent multi-AZ deployment, the difference between the number of add-on pods in different AZs will be less than or equal to 1. If resources in one of the AZs are insufficient, pods cannot be scheduled to that AZ.
tolerations	No	Array of Table 4-1263	Toleration configuration
node_match_expressions	No	Array of Table 4-1264	Add-on pod affinity configuration

Table 4-1262 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at autoscaler .

Parameter	Mandatory	Type	Description
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Table 4-1263 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Table 4-1264 nodeMatchExpression node affinity

Parameter	Mandatory	Type	Description
key	No	String	Taint key
values	No	List<String>	Node affinity name
operator	No	String	Operator

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "2292498e-*****-0255ac1001ba",
    "version": "1.23.116",
    "addonTemplateName": "autoscaler",
    "values": {
      "basic": {
        "basickey": "val"
      },
      "flavor": {
        "replicas": 2,
        "resources": [{
          "limitsCpu": "1000m",
          "limitsMem": "1000Mi",
          "name": "autoscaler",
          "requestsCpu": "1000m",
```



```
        "requestsMem": "1000Mi"
      }]
    },
    "custom": {
      "cluster_id": "2292498e-*****-0255ac1001ba",
      "coresTotal": 32000,
      "ignoreDaemonSetsUtilization": false,
      "logLevel": 4,
      "maxEmptyBulkDeleteFlag": 10,
      "maxNodeProvisionTime": 15,
      "maxNodesTotal": 1000,
      "memoryTotal": 128000,
      "multiAZBalance": false,
      "multiAZEnabled": false,
      "node_match_expressions": [],
      "scaleDownDelayAfterAdd": 10,
      "scaleDownDelayAfterDelete": 10,
      "scaleDownDelayAfterFailure": 3,
      "scaleDownEnabled": false,
      "scaleDownUnneededTime": 10,
      "scaleDownUtilizationThreshold": 0.5,
      "scaleUpUnscheduledPodEnabled": true,
      "scaleUpUtilizationEnabled": true,
      "skipNodesWithCustomControllerPods": true,
      "tenant_id": "*****",
      "tolerations": [{
        "key": "node.kubernetes.io/not-ready",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      },
      {
        "key": "node.kubernetes.io/unreachable",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      }
    ],
      "unremovableNodeRecheckTimeout": 5
    }
  }
}
```

4.14.6 NGINX Ingress Controller

Add-on Overview

The NGINX Ingress Controller add-on can automatically modify configurations when there are changes to pods in Services. This add-on uses Nginx's excellent stability, performance, and concurrent processing ability to effectively manage containerized applications at the application layer.

Add-on Parameters

Table 4-1265 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified

Parameter	Mandatory	Type	Description
flavor	Yes	Table 4-1266 object	Flavor parameters
custom	Yes	Table 4-1267 object	Custom parameters

Table 4-1266 flavor

Parameter	Mandatory	Type	Description
replicas	Yes	String	Number of pods. The default value is 1 .
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1267 custom

Parameter	Mandatory	Type	Description
ingressClass	Yes	String	Controller name. The default value is nginx .
namespace	Yes	String	Namespace where the add-on is in. The default value is kube-system .
service	Yes	Table 4-1272	Configuration of a Service that provides external access
config	No	Map<String>String	Nginx configuration parameters. For details, see ConfigMaps .
admissionWebhooks	No	Table 4-1271	Configuration of ingress admission verification.
metrics	No	Table 4-1273	Monitoring metric configuration.
defaultBackendService	No	String	Default 404 service, which is in the format of <i><namespace>/<service_name></i> .
extraArgs	No	Table 4-1274	Extended parameter configuration.

Parameter	Mandatory	Type	Description
multiAZEnable	No	bool	Whether to enable the forcible mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on will be forcibly scheduled to nodes in different AZs. If there are fewer AZs than pods, the extra pods will fail to run. If both multiAZEnable and multiAZBalance are set to true, the settings of multiAZBalance take effect, which means, the equivalent mode of multi-AZ deployment is used.
multiAZBalance	No	bool	Whether to enable the equivalent mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on are evenly scheduled to the nodes in the cluster in each AZ. If a new AZ is added, it is recommended that you increase add-on pods for cross-AZ HA deployment. With the equivalent multi-AZ deployment, the difference between the number of add-on pods in different AZs will be less than or equal to 1. If resources in one of the AZs are insufficient, pods cannot be scheduled to that AZ.
tolerations	No	Array of Table 4-1269	Toleration configuration
node_match_expressions	No	Array of Table 4-1270	Add-on pod affinity configuration

Table 4-1268 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at nginx-ingress .
requestsCpu	Yes	String	Requested CPU size (unit: m)

Parameter	Mandatory	Type	Description
requestsMemory	Yes	String	Requested memory size (unit: Mi)

Table 4-1269 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Table 4-1270 nodeMatchExpression node affinity

Parameter	Mandatory	Type	Description
key	No	String	Taint key
values	No	List<String>	Node affinity name
operator	No	String	Operator

Table 4-1271 admissionWebhook

Parameter	Mandatory	Type	Description
enable	No	bool	Whether to enable ingress resource admission verification. The default value is true .

Table 4-1272 service

Parameter	Mandatory	Type	Description
annotations	No	Map<String>String	Annotations of a Service that provides external access. This parameter is only for configurations of ELB-related options, such as kubernetes.io/elb.class , kubernetes.io/elb.id , and kubernetes.io/elb.pass-through .

Parameter	Mandatory	Type	Description
loadBalancerIP	No	String	Load balancer IP address used by the Service that is interconnected with the load balancer.

Table 4-1273 metrics

Parameter	Mandatory	Type	Description
enable	No	bool	Whether to monitor metrics. The default value is true .
excludeSocketMetrics	No	String	Shielded monitoring metrics. The default value is "nginx_ingress_controller_success,nginx_ingress_controller_header_duration_seconds,nginx_ingress_controller_ingress_upstream_latency_seconds" .

Table 4-1274 extraArg extended parameter

Parameter	Mandatory	Type	Description
default-ssl-certificate	No	String	Default certificate configuration. For details, see Default SSL Certificate .

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "2292498e-*****ac1001ba",
    "version": "2.2.52",
    "addonTemplateName": "nginx-ingress",
    "values": {
      "basic": {
        "basickey": "val"
      },
      "flavor": {
        "replicas": 2,
        "resources": [{
          "limitsCpu": "8000m",
          "limitsMem": "4000Mi",
          "name": "nginx-ingress",
          "requestsCpu": "8000m",
          "requestsMem": "4000Mi"
        }]
      }
    }
  }
}
```

```
    },
    "custom": {
      "config": {
        "keep-alive-requests": "100"
      },
      "defaultBackend": {
        "enabled": true
      },
      "defaultBackendService": "",
      "extraArgs": {
        "default-ssl-certificate": ""
      },
      "ingressClass": "nginx",
      "multiAZBalance": false,
      "multiAZEnabled": false,
      "namespace": "kube-system",
      "node_match_expressions": [],
      "service": {
        "annotations": {
          "kubernetes.io/elb.class": "performance",
          "kubernetes.io/elb.id": "8d6bd485-d8ac-4693-815d-9d54d79b0666"
        },
        "loadBalancerIP": ""
      },
      "tolerations": [{
        "key": "node.kubernetes.io/not-ready",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      },
      {
        "key": "node.kubernetes.io/unreachable",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      }
    ]
  }
}
```

4.14.7 Kubernetes Metrics Server

Add-on Overview

From version 1.8 onwards, Kubernetes provides resource usage metrics, such as the container CPU and memory usage, through the Metrics API. These metrics can be directly accessed by users (for example, by using the **kubectl top** command) or used by controllers (for example, Horizontal Pod Autoscaler) in a cluster for decision-making. The specific component is metrics-server, which is used to substitute for Heapster for providing the similar functions. Heapster has been gradually abandoned since v1.11.

Add-on Parameters

Table 4-1275 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified

Parameter	Mandatory	Type	Description
flavor	Yes	Table 4-1276 object	Flavor parameters
custom	Yes	Table 4-1277 object	Custom parameters

Table 4-1276 flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description
name	Yes	String	Add-on specification name. The value is fixed at Single-instance .
replicas	Yes	String	Number of pods. The default value is 1 .
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1277 custom

Parameter	Mandatory	Type	Description
multiAZBalance	No	Bool	Whether the equivalent mode of multi-AZ deployment is enabled. If this function is enabled, the equivalent mode is used.
multiAZEnabled	No	Bool	Whether to deploy the add-on pods in multiple AZs. The default value is false . If this parameter is set to true , cross-AZ deployment is forcibly performed. If this parameter is set to false , cross-AZ deployment is preferred.
tolerations	No	List<Object> Table 4-1280	Tolerations of the add-on
node_match_expressions	No	List<Object> Table 4-1280	Node affinity configuration of the add-on

Table 4-1278 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at metrics-server .
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Table 4-1279 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Table 4-1280 nodeMatchExpression node affinity

Parameter	Mandatory	Type	Description
key	No	String	Taint key
values	No	List<String>	Node affinity name
operator	No	String	Operator

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "b78fb690-b82c-11ee-83cf-0255ac100b0f",
    "version": "1.3.39",
```



```
"addonTemplateName": "metrics-server",
"values": {
  "basic": {
    "image_version": "v0.6.2",
    "swr_addr": "****",
    "swr_user": "****",
    "rbac_enabled": true,
    "cluster_version": "v1.23"
  },
  "flavor": {
    "description": "Has only one instance",
    "name": "Single",
    "replicas": 1,
    "resources": [
      {
        "limitsCpu": "1000m",
        "limitsMem": "1000Mi",
        "name": "metrics-server",
        "requestsCpu": "100m",
        "requestsMem": "300Mi"
      }
    ],
    "category": [
      "CCE",
      "Turbo"
    ]
  },
  "custom": {
    "annotations": {},
    "multiAZBalance": false,
    "multiAZEnabled": false,
    "node_match_expressions": [],
    "tolerations": [
      {
        "key": "node.kubernetes.io/not-ready",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      },
      {
        "key": "node.kubernetes.io/unreachable",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      }
    ]
  }
}
```

4.14.8 CCE Advanced HPA

Add-on Overview

CCE Advanced HPA (cce-hpa-controller) is an in-house add-on, which can be used to flexibly scale in or out Deployments based on metrics such as CPU usage and memory usage.

Add-on Parameters

Table 4-1281 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1282 object	Flavor parameters
custom	Yes	Table 4-1283 object	Custom parameters

Table 4-1282 flavor

Parameter	Mandatory	Type	Description
replicas	Yes	String	Number of pods. The default value is 2.
resources	Yes	resources object	Container resource (CPU and memory) quotas.

Table 4-1283 custom

Parameter	Mandatory	Type	Description
multiAZEnable	No	bool	Whether to enable the forcible mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on will be forcibly scheduled to nodes in different AZs. If there are fewer AZs than pods, the extra pods will fail to run. If both multiAZEnable and multiAZBalance are set to true, the settings of multiAZBalance take effect, which means, the equivalent mode of multi-AZ deployment is used.

Parameter	Mandatory	Type	Description
multiAZBalance	No	bool	Whether to enable the equivalent mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on are evenly scheduled to the nodes in the cluster in each AZ. If a new AZ is added, it is recommended that you increase add-on pods for cross-AZ HA deployment. With the equivalent multi-AZ deployment, the difference between the number of add-on pods in different AZs will be less than or equal to 1. If resources in one of the AZs are insufficient, pods cannot be scheduled to that AZ.
tolerations	No	Array of Table 4-1285	Toleration configuration
node_matching_expressions	No	Array of Table 4-1286	Add-on pod affinity configuration

Table 4-1284 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at customedhpa-controller .
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Table 4-1285 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy
operator	No	String	Operator

Parameter	Mandatory	Type	Description
tolerationSeconds	No	Int	Toleration time window

Table 4-1286 nodeMatchExpression node affinity

Parameter	Mandatory	Type	Description
key	No	String	Taint key
values	No	List<String >	Node affinity name
operator	No	String	Operator

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "2292498e-*****-0255ac1001ba",
    "version": "1.4.2",
    "addonTemplateName": "cce-hpa-controller",
    "values": {
      "basic": {
        "basickey": "val"
      },
      "flavor": {
        "replicas": 1,
        "resources": [{
          "limitsCpu": "100m",
          "limitsMem": "300Mi",
          "name": "customedhpa-controller",
          "requestsCpu": "100m",
          "requestsMem": "300Mi"
        }]
      },
      "custom": {
        "multiAZBalance": false,
        "multiAZEnabled": false,
        "node_match_expressions": [],
        "tolerations": [{
          "key": "node.kubernetes.io/not-ready",
          "operator": "Exists",
          "effect": "NoExecute",
          "tolerationSeconds": 60
        },
        {
          "key": "node.kubernetes.io/unreachable",
          "operator": "Exists",
          "effect": "NoExecute",
          "tolerationSeconds": 60
        }
      ]
    }
  }
}
```

```
}  
}
```

4.14.9 CCE Cloud Bursting Engine for CCI

Add-on Overview

The virtual-kubelet add-on is an implementation of the open-source Virtual Kubelet project. When momentary traffic spikes occur in CCE clusters, virtual-kubelet uses Huawei Cloud CCI to create pods for Deployments, StatefulSets, and jobs, eliminating the overhead of resizing CCE clusters. For details, see [virtual kubelet](#).

Add-on Parameters

Table 4-1287 Parameters

Parameter	Mandatory	Type	Description
basic	Yes	Table 4-1288 object	Basic add-on configuration parameters
flavor	Yes	Table 4-1289 object	Flavor parameters
custom	Yes	Table 4-1290 object	Custom parameters

Table 4-1288 basic

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID
cluster_name	Yes	String	Cluster name
vpc_id	Yes	String	VPC ID
network_id	Yes	String	Subnet ID
project_id	Yes	String	Project ID

Table 4-1289 flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description

Parameter	Mandatory	Type	Description
name	No	String	Add-on specification name
replicas	Yes	String	Number of pods. The default value is 1.
resources	No	resources object	Container resource (CPU and memory) quotas

Table 4-1290 custom

Parameter	Mandatory	Type	Description
isInstallProxy	No	String	Cross-service interworking. true: Pods in a Cloud Container Engine (CCE) cluster can communicate with pods in a Cloud Container Instance (CCI) cluster through Kubernetes Service.
subnet_id	Yes	String	IPv4 subnet ID
subnets	No	subnet object	Extended subnet configuration

Table 4-1291 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	No	String	CPU size limit (unit: m)
limitsMem	No	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name
requestsCpu	No	String	Requested CPU size (unit: m)
requestsMem	No	String	Requested memory size (unit: Mi)

Table 4-1292 Data structure of the subnet field

Parameter	Mandatory	Type	Description
subnetID	Yes	String	ID of the extended IPv4 subnet

Example Request

```
{
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "ccbe7bdf-4**9-3**b-b**4-0*****78",
    "version": "1.5.15",
    "addonTemplateName": "virtual-kubelet",
    "values": {
      "basic": {
        "cluster_id": "ccbe7bdf-4**9-3**b-b**4-0*****78",
        "cluster_name": "test-vk",
        "vpc_id": "2372199f-e6e0-48be-9437-e774aae6bd70",
        "network_id": "863a5e6c-e4f5-45f8-80d9-5090f17a767b",
        "project_id": "085a4*****00a9ccf7fba"
      },
      "flavor": {
        "description": "Has only one instance",
        "name": "Single",
        "replicas": 1,
        "resources": [
          {
            "name": "virtual-kubelet",
            "limitsCpu": "2000m",
            "requestsCpu": "500m",
            "limitsMem": "4096Mi",
            "requestsMem": "512Mi"
          },
          {
            "name": "bursting-virtual-kubelet",
            "limitsCpu": "2000m",
            "requestsCpu": "500m",
            "limitsMem": "4096Mi",
            "requestsMem": "512Mi"
          },
          {
            "name": "profile-controller",
            "limitsCpu": "1000m",
            "requestsCpu": "250m",
            "limitsMem": "1024Mi",
            "requestsMem": "256Mi"
          },
          {
            "name": "proxy",
            "limitsCpu": "2000m",
            "requestsCpu": "250m",
            "limitsMem": "4096Mi",
            "requestsMem": "512Mi"
          },
          {
            "name": "resource-syner",
            "limitsCpu": "2000m",
            "requestsCpu": "250m",
            "limitsMem": "4096Mi",
            "requestsMem": "512Mi"
          },
          {
            "name": "bursting-resource-syner",
            "limitsCpu": "2000m",
            "requestsCpu": "500m",
            "limitsMem": "1024Mi",
            "requestsMem": "256Mi"
          },
          {
            "name": "webhook",
            "limitsCpu": "2000m",

```

```
      "requestsCpu": "500m",
      "limitsMem": "1024Mi",
      "requestsMem": "256Mi"
    },
    {
      "name": "bursting-webhook",
      "limitsCpu": "2000m",
      "requestsCpu": "500m",
      "limitsMem": "1024Mi",
      "requestsMem": "256Mi"
    }
  ]
},
"custom": {
  "isInstallProxy": true,
  "subnet_id": "ad821361-288b-4d1f-9309-65aa2809e4fd",
  "subnets": [
    {
      "subnetID": "ce374ecf-1559-41ef-8967-9de579b1296012"
    }
  ]
}
}
```

4.14.10 CCE AI Suite (NVIDIA GPU)

Add-on Overview

CCE AI Suite (NVIDIA GPU) is a device management add-on that supports GPUs in containers. To use GPU nodes in a cluster, this add-on must be installed.

Add-on Parameters

Table 4-1293 Parameters

Parameter	Mandatory	Type	Description
basic	Yes	object	Basic add-on configuration parameters
custom	Yes	Table 4-1295 object	Custom parameters

Table 4-1294 Configuration of basic

Parameter	Mandatory	Type	Description
cluster_version	No	String	CCE cluster version
device_version	Yes	String	Add-on version

Parameter	Mandatory	Type	Description
driver_version	Yes	String	Image tag of an add-on pod where a driver is installed. Generally, the value is the same as that of device_version .
obs_url	Yes	String	When a GPU driver is downloaded from the default driver address, the value is the GPU driver address.
swr_addr	Yes	String	Image repository address
swr_user	Yes	String	Tenant path of an image repository

Table 4-1295 Configuration of custom

Parameter	Mandatory	Type	Description
compatible_with_legacy_api	No	Bool	API compatibility switch Default value: false true : The add-on supports the GPU native mode and xGPU virtualization.
component_schedulername	Yes	String	Name of the scheduler used by the add-on. Default value: default-scheduler
disable_mount_path_v1	No	Bool	Default value: false true : /opt/cloud/cce/nvidia is not mounted to the /usr/lib/nvidia directory of a GPU container.
disable_nvidia_gsp	No	Bool	Default value: true true : The GPU GSP firmware is disabled.

Parameter	Mandatory	Type	Description
driver_mount_paths	No	String	Driver file directory that needs to be automatically mounted to a GPU container Default value: "bin,lib64"
enable_fault_isolation	No	Bool	Default value: true true : The add-on detects hardware faults or driver issues of a GPU and then sets the GPU to be unavailable.
enable_health_monitoring	No	Bool	Default value: true true : The add-on detects hardware faults or driver issues of a GPU.
enable_metrics_monitoring	No	Bool	Default value: true true : The add-on collects GPU metrics and reports these metrics to Prometheus.
enable_simple_lib64_mount	No	Bool	Default value: true true : Only the libxxx.so.x file is mounted to a container.
enable_xgpu	No	Bool	Default value: false Whether to enable xGPU virtualization.
gpu_driver_config	No	Map	Configurations of the GPU driver for a single node pool Default value: {}
health_check_xids_v2	No	String	GPU error range for the add-on health checks Default value: "74,79"
inject_ld_library_path	No	String	Value of the LD_LIBRARY_PATH environment variable automatically injected by the add-on to a GPU container Default value: ""
lib64_container_paths	No	String	Mount path of NVIDIA lib64 in a GPU container Default value: "/usr/lib64,/usr/lib/x86_64-linux-gnu"

Parameter	Mandatory	Type	Description
metrics_delete_interval	No	int	Timeout threshold for deleting a metric when the metric cannot be obtained. The unit is millisecond. Default value: 30000
metrics_monitor_interval	No	int	Interval for obtaining metrics, in milliseconds. Default value: 15000
nvidia_driver_download_url	Yes	String	Path for downloading the NVIDIA driver Default value: ""

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "name": "gpu-beta",
  },
  "spec": {
    "clusterID": "80c9e306-***-***-***-0255ac100043",
    "version": "2.0.69",
    "addonTemplateName": "gpu-beta",
    "values": {
      "basic": {
        "cluster_version": "v1.27",
        "device_version": "2.0.69",
        "driver_version": "2.0.69",
        "obs_url": "****",
        "region": "****",
        "swr_addr": "****",
        "swr_user": "****"
      },
      "custom": {
        "compatible_with_legacy_api": true,
        "component_schedulername": "kube-scheduler",
        "disable_mount_path_v1": false,
        "disable_nvidia_gsp": true,
        "driver_mount_paths": "bin,lib64",
        "enable_fault_isolation": true,
        "enable_health_monitoring": true,
        "enable_metrics_monitoring": true,
        "enable_simple_lib64_mount": true,
        "enable_xgpu": true,
        "gpu_driver_config": {},
        "health_check_xids_v2": "74,79",
        "inject_ld_library_path": "",
        "lib64_container_paths": "/usr/lib64,/usr/lib/x86_64-linux-gnu",
        "metrics_delete_interval": 30000,
        "metrics_monitor_interval": 15000,
        "nvidia_driver_download_url": ""
      }
    }
  }
}
```

4.14.11 CCE AI Suite (Ascend NPU)

Add-on Overview

CCE AI Suite (Ascend NPU) is a device management add-on that supports NPUs in containers.

After this add-on is installed, you can create AI-accelerated nodes to quickly and efficiently process inference and image recognition.

Add-on Parameters

Table 4-1296 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1298 object	Flavor parameters
custom	Yes	Table 4-1299 object	Custom parameters

Table 4-1297 Configuration of basic

Parameter	Mandatory	Type	Description
cluster_version	Yes	String	CCE cluster version
device_version	Yes	String	Add-on version
driver_version	Yes	String	Image tag of an add-on pod where the driver is installed when automatic driver installation is enabled for the add-on. Generally, the value is the same as that of device_version .
swr_addr	Yes	String	Image repository address

Parameter	Mandatory	Type	Description
swr_user	Yes	String	Tenant path of an image repository

Table 4-1298 Configuration of flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description
name	Yes	String	Add-on specification name. The value is fixed at Single-instance .
replicas	Yes	String	Number of pods. The default value is 1 .
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1299 Configuration of custom

Parameter	Mandatory	Type	Description
auto_install_npu_driver	No	Bool	Default value: false true : The NPU driver is automatically installed on a node. Only some specifications of 310 and 310P cards are supported.
check_frequency_failed_threshold	No	Int	Threshold for the add-on to check how many times an NPU device is considered unhealthy Default value: 100
check_frequency_fall_times	No	Int	Threshold for the add-on to check whether to isolate a chip when the dominant frequency of the chip is reduced Default value: 3
check_frequency_gate	No	Bool	true : Checks on the chip dominant frequency are enabled. Default value: false

Parameter	Mandatory	Type	Description
check_frequency_recover_threshold	No	Int	Threshold for the add-on to check how many times an NPU device is considered healthy Default value: 100
check_frequency_rise_times	No	Int	Threshold for the add-on to check whether the chip dominant frequency is restored Default value: 2
container_path	No	String	Path for mounting the Hiai Library in a container Default value: "/usr/local/HiAI_unused"
host_path	No	String	Path containing the Hiai library on a host Default value: "/usr/local/HiAI_unused"
npu_driver_config	No	Map	If an NPU driver is automatically installed on a node, the key of this parameter specifies the driver model, and the value specifies the address for downloading the NPU driver of that model. Default value: {}

Table 4-1300 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m) Default value: 1000m
limitsMem	Yes	String	Memory size limit (unit: Mi) Default value: 4096Mi
name	Yes	String	Add-on name. The value is fixed at npu-driver-installer .
requestsCpu	Yes	String	Requested CPU size (unit: m) Default value: 50m
requestsMem	Yes	String	Requested memory size (unit: Mi) Default value: 100Mi

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "name": "huawei-npu",
  },
  "spec": {
    "clusterID": "e93c2716-****-****-****-0255ac10004e",
    "version": "2.0.26",
    "addonTemplateName": "huawei-npu",
    "values": {
      "basic": {
        "cluster_version": "v1.23",
        "device_version": "2.0.26",
        "driver_version": "2.0.26",
        "platform": "linux-amd64",
        "rbac_enabled": true,
        "swr_addr": "****",
        "swr_user": "****",
      },
      "custom": {
        "annotations": {},
        "auto_install_npu_driver": true,
        "check_frequency_failed_threshold": 100,
        "check_frequency_fall_times": 3,
        "check_frequency_gate": false,
        "check_frequency_recover_threshold": 100,
        "check_frequency_rise_times": 2,
        "container_path": "/usr/local/HiAI_unused",
        "host_path": "/usr/local/HiAI_unused",
        "npu_driver_config": {}
      },
      "flavor": {
        "category": [
          "CCE",
          "Turbo"
        ],
        "name": "default",
        "resources": [
          {
            "limitsCpu": "1000m",
            "limitsMem": "4096Mi",
            "name": "npu-driver-installer",
            "requestsCpu": "50m",
            "requestsMem": "100Mi"
          }
        ]
      }
    }
  }
}
```

4.14.12 Volcano Scheduler

Add-on Overview

Volcano is a batch scheduling platform based on Kubernetes. It provides a series of features required by machine learning, deep learning, bioinformatics, genomics, and other big data applications, as a powerful supplement to Kubernetes capabilities.

Add-on Parameters

Table 4-1301 Parameters

Parameter	Mandatory	Type	Description
basic	No	Table 4-1302 object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1303 object	Flavor parameters
custom	Yes	Table 4-1304 object	Custom parameters

Table 4-1302 Configuration of basic

Parameter	Mandatory	Type	Description
swr_addr	Yes	String	Add-on download address, which does not need to be specified
swr_user	Yes	String	User who can download the add-on. This parameter does not need to be specified.
platform	Yes	String	Add-on platform, which does not need to be specified
escEndpoint	Yes	String	ECS address, which does not need to be specified
xccsEndpoint	- Add-on versions $\geq$ 1.16.11: Yes - Add-on versions $<$ 1.16.11: No	String	XCCS service address, which does not need to be specified

Table 4-1303 Configuration of flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description

Parameter	Mandatory	Type	Description
name	Yes	String	Add-on specification name - Add-on versions $\geq$ 1.14.7: Node50, Node200, Node1000, and custom-resources - Add-on versions earlier than 1.14.7: HA, Single, and custom-resources
replicas	Yes	String	Number of pods. The default value is 2.
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1304 Configuration of custom

Parameter	Mandatory	Type	Description
multiAZEnabled	No	Bool	Whether to enable multi-AZ deployment for the add-on. The default value is false . true: Volcano pods are deployed in different AZs based on the hard anti-affinity policy. false: Volcano pods are deployed in multiple AZs based on the soft anti-affinity policy.
controller_kube_api_qps	No	int	API server QPS of the controller component. The default value is 200 .
scheduler_kube_api_qps	No	int	API server QPS of the scheduler component. The default value is 200 .
admission_kube_api_qps	No	int	API server QPS of the admission component. The default value is 200 .
update_pod_status_qps	No	int	Used to update the pod status QPS. The default value is 200 .
admissions	No	string	Webhooks supported by Volcano
colocation_enable	No	string	Whether hybrid deployment is supported
oversubscription_ratio	No	int	Dynamic oversubscription ratio. The default value is 60 .

Parameter	Mandatory	Type	Description
oversubscription_method	No	string	Method of calculating oversubscribed resources. The options are nodeResource and podProfile . nodeResource is the default algorithm based on node resource usage, and podProfile is the algorithm based on pod profiling. By default, nodeResource is used.
oversubscription_profile_period	No	int	Interval for pod profiling, in seconds
workload_balancer_third_party_types	No	string	Character string consisting of group, version, and kind of a third-party workload
workload_balancer_score_annotation_key	No	string	Used to specify the score annotation key of a pod
node_matching_expressions	No	Table 4-1307	Expression for matching the Volcano Scheduler pods to nodes
tolerations	No	Table 4-1306	The format is the same as that of Kubernetes tolerations. It is used to add taints to Volcano Scheduler pods.
oversubscription_ratio	No	int	Node resource overcommitment ratio in the Volcano scheduling environment
descheduler_enable	No	Bool	Whether rescheduling is supported
enable_workload_balancer	No	Bool	Whether load balancers are supported
default_scheduler_conf	Yes	yaml	The format is the same as that of the YAML for Volcano. For details, see Volcano Scheduler .
descheduler Policy	No	yaml	The format is the same as that of the YAML for Volcano descheduling configuration. For details, see Descheduling .

Table 4-1305 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m) The default values are differentiated by component. For details about key components, see Volcano Scheduler .
limitsMem	Yes	String	Memory size limit (unit: Mi) The default values are differentiated by component. For details about key components, see Volcano Scheduler .
name	Yes	String	Add-on name
requestsCpu	Yes	String	Requested CPU size (unit: m) The default values are differentiated by component. For details about key components, see Volcano Scheduler .
requestsMem	Yes	String	Requested memory size (unit: Mi) The values are differentiated by component. For details about key components, see Volcano Scheduler .

Table 4-1306 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint effect
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Table 4-1307 nodeMatchExpression node affinity

Parameter	Mandatory	Type	Description
key	No	String	Taint key
values	No	List<String>	Node affinity name
operator	No	String	Operator

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "ad24dc34-*****-0255ac100030",
    "version": "1.16.8",
    "addonTemplateName": "volcano",
    "values": {
      "basic": {
        "ecsEndpoint": "x.x.x.x",
        "platform": "linux-amd64",
        "swr_addr": "swr.cn-north-7.myhuaweicloud.com",
        "swr_user": "hwofficial"
      },
      "flavor": {
        "description": "For 50 nodes, 5000 pods in cluster",
        "name": "Node50",
        "resources": [
          {
            "name": "volcano-scheduler",
            "limitsCpu": "2000m",
            "requestsCpu": "500m",
            "replicas": 2,
            "limitsMem": "2000Mi",
            "requestsMem": "500Mi"
          },
          {
            "name": "volcano-controller",
            "limitsCpu": "2000m",
            "requestsCpu": "500m",
            "replicas": 2,
            "limitsMem": "2000Mi",
            "requestsMem": "500Mi"
          },
          {
            "name": "volcano-admission",
            "limitsCpu": "500m",
            "requestsCpu": "200m",
            "replicas": 2,
            "limitsMem": "500Mi",
            "requestsMem": "500Mi"
          },
          {
            "limitsCpu": "200m",
            "limitsMem": "200Mi",
            "name": "volcano-agent",
            "requestsCpu": "100m",
            "requestsMem": "150Mi"
          },
          {
            "limitsCpu": "100m",
            "limitsMem": "100Mi",
            "name": "resource-exporter",
            "requestsCpu": "50m",
            "requestsMem": "50Mi"
          },
          {
            "limitsCpu": "1000m",
            "limitsMem": "512Mi",
            "name": "volcano-descheduler",
            "replicas": 2,
            "requestsCpu": "500m",
            "requestsMem": "256Mi"
          }
        ]
      }
    }
  }
}
```

```
    },
    {
      "limitsCpu": "500m",
      "limitsMem": "1000Mi",
      "name": "volcano-recommender",
      "replicas": 2,
      "requestsCpu": "300m",
      "requestsMem": "500Mi"
    },
    {
      "limitsCpu": "300m",
      "limitsMem": "300Mi",
      "name": "volcano-recommender-prometheus-adapter",
      "replicas": 2,
      "requestsCpu": "200m",
      "requestsMem": "200Mi"
    }
  ],
  "size": "small",
  "category": [
    "CCE",
    "Turbo"
  ]
},
"custom": {
  "admission_kube_api_qps": 200,
  "admissions": "/jobs/mutate,/jobs/validate,/podgroups/mutate,/pods/validate,/pods/mutate,/queues/mutate,/queues/validate,/eas/pods/mutate,/eas/pods/validate,/npu/jobs/validate,/resource/validate,/resource/mutate,/workloadbalancer/balancer/validate,/workloadbalancer/balancerpolicytemplate/validate",
  "colocation_enable": "false",
  "controller_kube_api_qps": 200,
  "default_scheduler_conf": {
    "actions": "allocate, backfill, preempt",
    "metrics": {
      "interval": "30s",
      "type": ""
    }
  },
  "tiers": [
    {
      "plugins": [
        {
          "name": "priority"
        },
        {
          "enableJobStarving": false,
          "enablePreemptable": false,
          "name": "gang"
        },
        {
          "name": "conformance"
        }
      ]
    }
  ],
  {
    "plugins": [
      {
        "enablePreemptable": false,
        "name": "drf"
      },
      {
        "name": "predicates"
      },
      {
        "name": "nodeorder"
      }
    ]
  },
  {
    "plugins": [
```

```
        {
          "name": "cce-gpu-topology-predicate"
        },
        {
          "name": "cce-gpu-topology-priority"
        },
        {
          "name": "xgpu"
        }
      ]
    },
    {
      "plugins": [
        {
          "name": "nodelocalvolume"
        },
        {
          "name": "nodeemptydirvolume"
        },
        {
          "name": "nodeCSIscheduling"
        },
        {
          "name": "networkresource"
        }
      ]
    }
  ]
},
"deschedulerPolicy": {
  "profiles": [
    {
      "name": "ProfileName",
      "pluginConfig": [
        {
          "args": {
            "nodeFit": true
          },
          "name": "DefaultEvictor"
        },
        {
          "args": {
            "evictableNamespaces": {
              "exclude": [
                "kube-system"
              ]
            },
            "thresholds": {
              "cpu": 20,
              "memory": 20
            }
          },
          "name": "HighNodeUtilization"
        }
      ],
      "args": {
        "evictableNamespaces": {
          "exclude": [
            "kube-system"
          ]
        },
        "metrics": {
          "type": "prometheus_adaptor"
        },
        "nodeFit": true,
        "targetThresholds": {
          "cpu": 80,
          "memory": 85
        }
      }
    }
  ]
}
```

```
        "thresholds": {
            "cpu": 30,
            "memory": 30
        },
        "name": "LoadAware"
    },
    "plugins": {
        "balance": {
            "enabled": null
        }
    }
},
"descheduler_enable": "false",
"deschedulingInterval": "10m",
"enable_workload_balancer": false,
"multiAZEnabled": false,
"node_match_expressions": [],
"oversubscription_method": "nodeResource",
"oversubscription_profile_period": 300,
"oversubscription_ratio": 60,
"scheduler_kube_api_qps": 200,
"tolerations": [
    {
        "effect": "NoExecute",
        "key": "node.kubernetes.io/not-ready",
        "operator": "Exists",
        "tolerationSeconds": 60
    },
    {
        "effect": "NoExecute",
        "key": "node.kubernetes.io/unreachable",
        "operator": "Exists",
        "tolerationSeconds": 60
    },
    {
        "effect": "NoSchedule",
        "key": "node.cilium.io/agent-not-ready",
        "operator": "Exists"
    }
],
"update_pod_status_qps": 50,
"workload_balancer_score_annotation_key": "",
"workload_balancer_third_party_types": "",
"multiAZBalance": false
}
}
```

4.14.13 CCE Secrets Manager for DEW

Add-on Overview

The CCE Secrets Manager for DEW add-on (dew-provider) is used to interconnect with Data Encryption Workshop (DEW). This add-on allows you to mount secrets stored outside a cluster (DEW for storing sensitive information) to pods. In this way, sensitive information can be decoupled from the cluster environment, which prevents information leakage caused by program hardcoding or plaintext configuration.

Add-on Parameters

Table 4-1308 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
custom	Yes	Table 4-1309 object	Custom parameters

Table 4-1309 Configuration of custom

Parameter	Mandatory	Type	Description
rotation_poll_interval	No	String	Rotation interval. The default value is 2m . The rotation interval specifies the interval for sending requests to Cloud Secret Management Service (CSMS) and getting the latest credentials. The proper interval is [1m, 1440m].

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "2292498e-d169-*****-0255ac1001ba",
    "version": "1.1.1",
    "addonTemplateName": "dew-provider",
    "values": {
      "basic": {
        "basickey": "val"
      },
      "flavor": {
        "resources": [{
          "limitsCpu": "100m",
          "limitsMem": "100Mi",
          "name": "dew-provider",
          "requestsCpu": "100m",
          "requestsMem": "100Mi"
        }]
      },
      "custom": {
        "rotation_poll_interval": "2m"
      }
    }
  }
}
```


4.14.14 CCE Network Metrics Exporter

Add-on Overview

CCE Network Metrics Exporter (dolphin) is an add-on for monitoring and managing container network traffic. It collects traffic statistics of containers that do not use the host network in CCE Turbo clusters and performs node-wide container connectivity checks.

Add-on Parameters

Table 4-1310 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1311 object	Flavor parameters
custom	Yes	Table 4-1312 object	Custom parameters

Table 4-1311 Configuration of flavor

Parameter	Mandatory	Type	Description
name	Yes	String	Add-on specification name. The value is fixed at default .
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1312 Configuration of custom

Parameter	Mandatory	Type	Description
annotations	No	Map<String>String	Custom annotation

Table 4-1313 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m) Default value: 500m
limitsMem	Yes	String	Memory size limit (unit: Mi) Default value: 512Mi
name	Yes	String	Add-on name. The value is fixed at dolphin .
requestsCpu	Yes	String	Requested CPU size (unit: m) Default value: 500m
requestsMem	Yes	String	Requested memory size (unit: Mi) Default value: 512Mi

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "name": "dolphin",
    "alias": "CCE Network Metrics Exporter",
    "addon.install/type": "install"
  },
  "spec": {
    "clusterID": "****",
    "version": "1.4.5",
    "addonTemplateName": "dolphin",
    "values": {
      "basic": {
        "cluster_version": "v1.28",
        "image_version": "1.4.5",
        "platform": "linux-amd64",
        "rbac_enabled": true,
        "swr_addr": "****",
        "swr_user": "****"
      },
      "custom": {
        "annotations": {}
      },
      "flavor": {
        "name": "default",
        "resources": [
          {
            "limitsCpu": "500m",
            "limitsMem": "512Mi",
            "name": "dolphin",
            "requestsCpu": "500m",
            "requestsMem": "512Mi"
          }
        ]
      }
    }
  }
}
```

4.14.15 NodeLocal DNSCache

Add-on Overview

NodeLocal DNSCache is an add-on developed based on the community NodeLocal DNSCache. This add-on functions as a DaemonSet to run the DNS cache proxy on cluster nodes to improve cluster DNS performance.

Add-on Parameters

Table 4-1314 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1315 object	Flavor parameters
custom	Yes	Table 4-1316 object	Custom parameters

Table 4-1315 Configuration of flavor

Parameter	Mandatory	Type	Description
replicas	No	String	Number of admission-controller component instances in the add-on. The default value is 2.
resources	No	Array resources object	Container resource (CPU and memory) quotas

Table 4-1316 Configuration of custom

Parameter	Mandatory	Type	Description
enable_dnsconfig_admission	No	bool	Enables DNSConfig automatic injection. The default value is true . After this function is enabled, a DNSConfig admission controller will be created. The controller intercepts pod creation requests in the namespace labeled with node-localdns-injection=enabled based on Admission Webhook, and automatically configures Pod dnsConfig that uses the DNS cache. If this function is disabled or the pod belongs to a non-target namespace, you must manually configure DNSConfig for the pod.
enable_namespace_admission	No	bool	Adds node-local-dns-injection=enabled to a created namespace. The default value is true . After this tag is added to a namespace, the system identifies the namespace creation request and automatically adds the tag. The target of these operations does not include the built-in namespaces (such as kube-system).
multiAZEnable	No	bool	Whether to enable the forcible mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on will be forcibly scheduled to nodes in different AZs. If there are fewer AZs than pods, the extra pods will fail to run. If both multiAZEnable and multiAZBalance are set to true, the settings of multiAZBalance take effect, which means, the equivalent mode of multi-AZ deployment is used.

Parameter	Mandatory	Type	Description
multiAZBalance	No	bool	Whether to enable the equivalent mode of multi-AZ deployment for the deployment component. The default value is false . Deployment pods of the add-on are evenly scheduled to the nodes in the cluster in each AZ. If a new AZ is added, it is recommended that you increase add-on pods for cross-AZ HA deployment. With the equivalent multi-AZ deployment, the difference between the number of add-on pods in different AZs will be less than or equal to 1. If resources in one of the AZs are insufficient, pods cannot be scheduled to that AZ.
tolerations	No	Array of Table 4-1318	Tolerations of the admission-controller component

Table 4-1317 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Add-on name. The value is fixed at node-local-dns-admission-controller or node-local-dns-cache .
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Table 4-1318 Taints and tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key
effect	No	String	Taint policy
operator	No	String	Operator
tolerationSeconds	No	Int	Toleration time window

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "2292498e-d169-*****-0255ac1001ba",
    "version": "1.6.2",
    "addonTemplateName": "node-local-dns",
    "values": {
      "basic": {
        "basickey": "val"
      },
      "flavor": {
        "replicas": 2,
        "resources": [{
          "limitsCpu": "250m",
          "limitsMem": "512Mi",
          "name": "node-local-dns-admission-controller",
          "requestsCpu": "250m",
          "requestsMem": "512Mi"
        },
        {
          "limitsCpu": "500m",
          "limitsMem": "512Mi",
          "name": "node-local-dns-cache",
          "requestsCpu": "25m",
          "requestsMem": "5Mi"
        }
      ]
    },
    "custom": {
      "enable_dnsconfig_admission": true,
      "enable_namespace_admission": true,
      "multiAZBalance": false,
      "multiAZEnabled": false,
      "node_match_expressions": [],
      "tolerations": [{
        "key": "node.kubernetes.io/not-ready",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      },
      {
        "key": "node.kubernetes.io/unreachable",
        "operator": "Exists",
        "effect": "NoExecute",
        "tolerationSeconds": 60
      }
    ]
  }
}
```

4.14.16 Cloud Native Cluster Monitoring

Add-on Overview

The Cluster Native Cluster Monitoring add-on (kube-prometheus-stack) uses Prometheus-operator and Prometheus to provide easy-to-use, end-to-end Kubernetes cluster monitoring.

This add-on enables monitoring data to connect to the monitoring center, so that you can view monitoring data and configure alarms on the console.

Add-on Parameters

Table 4-1319 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1320 object	Flavor parameters
custom	Yes	Table 4-1321 object	Custom parameters

Table 4-1320 Configuration of flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description
name	Yes	String	Add-on specification name
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1321 Configuration of custom

Parameter	Mandatory	Type	Description
aom_enable	No	Boolean	Whether to interconnect with AOM
aom_instance_id	No	String	AOM instance ID. This parameter is mandatory when AOM is interconnected with the cluster.

Parameter	Mandatory	Type	Description
aom_project_id	No	String	ID of the project to which the AOM instance belongs. This parameter is mandatory when AOM is interconnected with the cluster.
aom_auth_type	No	String	Authentication type for interconnecting with AOM. This parameter is mandatory when AOM is interconnected with the cluster. The value is fixed at Bearer .
aom_app_key	No	String	app_key for interconnecting with AOM. This parameter is mandatory when AOM is interconnected with the cluster.
aom_app_secret	No	String	app_secret for interconnecting with AOM. This parameter is mandatory when AOM is interconnected with the cluster.
deploy_model	Yes	String	Prometheus add-on model. The value can be agent or server . agent is recommended.
enablethird	No	Boolean	Whether to report metrics to the third-party monitoring system.
url_third	No	String	URL for the third-party monitoring system to report metrics.
basic_auth_username_third	No	String	Account for interconnecting with a third-party monitoring system (in the BasicAuth authentication mode)
basic_auth_password_third	No	String	Password for interconnecting with a third-party monitoring system (in the BasicAuth authentication mode)
bearer_token	No	String	Token for interconnecting with a third-party monitoring system (in the BearerToken authentication mode)
cluster	Yes	String	Cluster name
clusterId	Yes	String	Cluster ID
enable_custom_metrics	No	Boolean	Whether to enable custom metric collection. The default value is false .
highAvailability	No	Boolean	Whether HA is enabled. The default value is false .

Parameter	Mandatory	Type	Description
scrapeInterval	Yes	String	Default metric collection period. The default value is 15s .
shards	No	Integer	Number of collected fragments. This parameter is valid only in the agent mode. The default value is 1 .

Table 4-1322 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Workload name
requestsCpu	Yes	String	Requested CPU size (unit: m)
requestsMem	Yes	String	Requested memory size (unit: Mi)

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "b889764e-c100-11ee-80cf-0255ac100b0f",
    "version": "3.9.5",
    "addonTemplateName": "cie-collector",
    "values": {
      "basic": {
        "aom_endpoint": "https://***",
        "aom_url": "https://***",
        "region_id": "***",
        "swr_addr": "***",
        "swr_user": "***",
        "rbac_enabled": true,
        "cluster_version": "v1.27"
      },
      "flavor": {
        "deploy_mode": "agent",
        "description": "Default flavor for agent mode.",
        "name": "Agent-Default",
        "resources": [
          {
            "limitsCpu": "500m",
            "limitsMem": "500Mi",
            "name": "prometheusOperator",
            "requestsCpu": "100m",
```

```
        "requestsMem": "100Mi"
      },
      {
        "limitsCpu": "4",
        "limitsMem": "8Gi",
        "name": "prometheus",
        "requestsCpu": "500m",
        "requestsMem": "500Mi"
      },
      {
        "limitsCpu": "500m",
        "limitsMem": "500Mi",
        "name": "kubeStateMetrics",
        "requestsCpu": "200m",
        "requestsMem": "200Mi"
      },
      {
        "limitsCpu": "500m",
        "limitsMem": "1Gi",
        "name": "nodeExporter",
        "requestsCpu": "200m",
        "requestsMem": "100Mi"
      }
    ],
    "category": [
      "CCE",
      "Turbo"
    ],
    "custom": {
      "aom_app_key": "*****",
      "aom_app_secret": "*****",
      "aom_auth_type": "Bearer",
      "aom_enable": true,
      "aom_instance_id": "a2bf5a6f-2c64-4a7f-a369-78dcd9cb6fd",
      "aom_project_id": "b6315dd3d0ff4be5b31a963256794989",
      "basic_auth_password_third": "",
      "basic_auth_username_third": "",
      "bearer_token": "",
      "cluster": "test-cluster",
      "clusterId": "b889764e-c100-11ee-80cf-0255ac100b0f",
      "deploy_mode": "agent",
      "enable_custom_metrics": true,
      "enablethird": false,
      "highAvailability": false,
      "projectId": "b6315dd3d0ff4be5b31a963256794989",
      "region": "****",
      "scrapeInterval": "15s",
      "shards": 1,
      "url_third": ""
    }
  }
}
```

4.14.17 Cloud Native Log Collection

Add-on Overview

The Cloud Native Logging add-on (log-agent) is developed based on Fluent Bit and OpenTelemetry for collecting logs and Kubernetes events. This add-on supports CRD-based log collection policies. It collects and forwards standard output logs, container file logs, node logs, and Kubernetes event logs in a cluster based on configured policies. It also reports Kubernetes events to AOM for configuring event alarms. By default, all abnormal events and some normal events are reported.

Add-on Parameters

Table 4-1323 Parameters

Parameter	Mandatory	Type	Description
basic	No	object	Basic configuration parameters, which do not need to be specified
flavor	Yes	Table 4-1324 object	Flavor parameters
custom	Yes	Table 4-1325 object	Custom parameters

Table 4-1324 Configuration of flavor

Parameter	Mandatory	Type	Description
description	No	String	Add-on description
name	Yes	String	Add-on specification name. The value is fixed at Single-instance .
replicas	Yes	String	Number of otel-collector component instances. The default value is 1 .
resources	Yes	resources object	Container resource (CPU and memory) quotas

Table 4-1325 Configuration of custom

Parameter	Mandatory	Type	Description
caCert	Yes	String	Specifies a client certificate, which is encrypted using Base64. This parameter is optional during installation. It is generated by the CCE add-on center and used for two-way authentication between components and webhook requests. Domain name to be issued: *.monitoring.svc .

Parameter	Mandatory	Type	Description
serverCert	Yes	String	Specifies a server certificate, which is encrypted using Base64. This parameter is optional during installation. It is generated by CCE add-ons and used for two-way authentication between components and webhook requests. Domain name to be issued: *.monitoring.svc .
serverKey	Yes	String	Specifies the private key of a server, which is encrypted using Base64. This parameter is optional during installation. It is generated by CCE add-ons and used for two-way authentication between components and webhook requests. Domain name to be issued: *.monitoring.svc .
accessKey	No	String	Specifies the access key ID of a user, which is used to request AOM and LTS APIs. If this parameter is left blank, a temporary AK/SK is used.
secretKey	No	String	Specifies the access key of a user, which is used to request AOM and LTS APIs. If this parameter is left blank, a temporary AK/SK is used.
createDefaultStdout	No	Bool	Specifies whether to create the default policy for collecting container standard output logs and reporting them to LTS. This parameter is valid only after it is specified during the add-on installation. The default value is false .
createDefaultEvent	No	Bool	Specifies whether to create the default policy for collecting Kubernetes events and reporting them to LTS. This parameter is valid only after it is specified during the add-on installation. The default value is false .
multiAZEnabled	No	Bool	Specifies whether to deploy the add-on pods in multiple AZs. The default value is false . If this parameter is set to true , cross-AZ deployment is forcibly performed. If this parameter is set to false , cross-AZ deployment is preferred.

Parameter	Mandatory	Type	Description
cluster_category	No	String	The value is fixed at CCE .
ltsAccessEndpoint	No	String	Specifies the address for reporting logs to LTS. If this parameter is left blank, the address specified in basic is used.
ltsEndpoint	No	String	Specifies the address of the LTS APIs. If this parameter is left blank, the address specified in basic is used.
aomEndpoint	No	String	Specifies the address for reporting events to AOM. If this parameter is left blank, the address specified in basic is used.
projectID	Yes	String	Specifies the ID of the project to which the current CCE cluster belongs. This parameter is optional during installation.
clusterID	Yes	String	Specifies the ID of the current CCE cluster. This parameter is optional during installation.
clusterName	Yes	String	Specifies the name of the current CCE cluster.

Table 4-1326 Data structure of the resources field

Parameter	Mandatory	Type	Description
limitsCpu	Yes	String	CPU size limit (unit: m)
limitsMem	Yes	String	Memory size limit (unit: Mi)
name	Yes	String	Component name. Options: fluent-bit : component for collecting node logs cop-logs : generates soft links for collected files on each node. log-operator : generates fluent-bit and otel-collector configurations. otel-collector : reports collected logs and events to AOM and LTS.
requestsCpu	Yes	String	Requested CPU size (unit: m)

Parameter	Mandatory	Type	Description
requestsMem	Yes	String	Requested memory size (unit: Mi)

Example Request

```
{
  "kind": "Addon",
  "apiVersion": "v3",
  "metadata": {
    "annotations": {
      "addon.install/type": "install"
    }
  },
  "spec": {
    "clusterID": "ccbe7bdf-4**9-3**b-b**4-0*****78",
    "version": "1.4.4",
    "addonTemplateName": "log-agent",
    "values": {
      "basic": {
        "aomEndpoint": "https://***",
        "iam_url": "****",
        "ltsAccessEndpoint": "https://***:8102",
        "ltsEndpoint": "https://***",
        "region": "****",
        "swr_addr": "****",
        "swr_user": "****",
        "rbac_enabled": true,
        "cluster_version": "v1.25"
      }
    },
    "flavor": {
      "description": "Recommended when the number of logs per second does not exceed 5000.",
      "name": "Low",
      "replicas": 1,
      "resources": [
        {
          "limitsCpu": "500m",
          "limitsMem": "500Mi",
          "name": "fluent-bit",
          "requestsCpu": "100m",
          "requestsMem": "200Mi"
        },
        {
          "limitsCpu": 1,
          "limitsMem": "500Mi",
          "name": "cop-logs",
          "requestsCpu": "100m",
          "requestsMem": "100Mi"
        },
        {
          "limitsCpu": "500m",
          "limitsMem": "500Mi",
          "name": "log-operator",
          "requestsCpu": "100m",
          "requestsMem": "100Mi"
        },
        {
          "limitsCpu": 1,
          "limitsMem": "2Gi",
          "name": "otel-collector",
          "requestsCpu": "200m",
          "requestsMem": "1Gi"
        }
      ],
      "category": [
```

```
    "CCE",
    "Turbo"
  ]
},
"custom": {
  "accessKey": "",
  "aomEndpoint": "https://***",
  "aomPrivateEndpointIP": "",
  "caCert": "",
  "clusterID": "",
  "clusterName": "clusterName",
  "cluster_category": "CCE",
  "createAudit": false,
  "createDefaultEvent": false,
  "createDefaultEventToAOM": true,
  "createDefaultStdout": false,
  "createKubeApiserver": false,
  "createKubeControllerManager": false,
  "createKubeScheduler": false,
  "ltsAccessEndpoint": "https://***:8102",
  "ltsAuditStreamID": "",
  "ltsEndpoint": "https://***",
  "ltsEventStreamID": "",
  "ltsGroupID": "",
  "ltsKubeApiserverStreamID": "",
  "ltsKubeControllerManagerStreamID": "",
  "ltsKubeSchedulerStreamID": "",
  "ltsLogReportDomain": "",
  "ltsPrivateEndpointIP": "",
  "ltsStdoutStreamID": "",
  "multiAZEnabled": false,
  "paasaksEnable": true,
  "projectID": "",
  "secretKey": "",
  "securityToken": "",
  "serverCert": "",
  "serverKey": ""
}
}
}
```

5 Kubernetes APIs

Description

Kubernetes APIs are resource-based (RESTful) programming interfaces provided through HTTP. It supports query, creation, update, and deletion of various cluster resources using standard HTTP request methods (POST, PUT, PATCH, DELETE, and GET).

CCE allows you to use native [Kubernetes APIs](#) in the following ways:

- [Calling Kubernetes APIs Through the Cluster API Server](#). It is suitable for API calls on scale thanks to its direct connection to the API Server. This is a recommended option.
- [Calling Kubernetes APIs Through API Gateway](#). It applies to small-scale API calls. API gateway flow control may be triggered when APIs are called on scale.

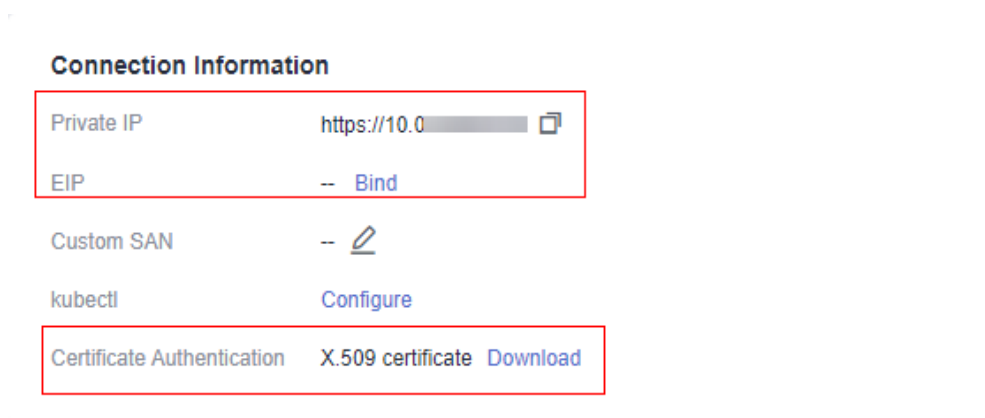
Calling Kubernetes APIs Through the Cluster API Server

You can use the API server of a Kubernetes cluster to call Kubernetes-native APIs.

Step 1 Obtain the cluster certificate and API server.

- Method 1: Obtain the certificate by calling the API for [obtaining the cluster certificate](#), save the returned information to the **kubeconfig.json** file, and extract the certificate, private key, and API server information. The commands are as follows:

```
# Obtain the CA certificate of the cluster and save it as ca.crt.
cat ./kubeconfig.json |grep certificate-authority-data | awk -F "" '{print $4}' | base64 -d > ./ca.crt
# Obtain the client certificate and save it as client.crt.
cat ./kubeconfig.json |grep client-certificate-data | awk -F "" '{print $4}' | base64 -d > ./client.crt
# Obtain the client private key and save it as client.key.
cat ./kubeconfig.json |grep client-key-data | awk -F "" '{print $4}' | base64 -d > ./client.key
# Obtain the API server.
cat ./kubeconfig.json |grep server | awk -F "" '{print $4}'
```
- Method 2: Obtain the API server IP address (private or public network address) on the **Overview** page of the CCE console and download the certificate (**ca.crt**, **client.crt**, and **client.key** files).



Step 2 Call Kubernetes-native APIs using the cluster certificate.

For example, run the **curl** command to call an API to view the pod information. The following is an example:

```
curl --cacert ./ca.crt --cert ./client.crt --key ./client.key https://192.168.0.198:5443/api/v1/namespaces/default/pods/
```

Specifically:

- `./ca.crt`, `./client.crt`, and `./client.key` specify that the certificate files in the current path are used. Replace them with the actual file paths.
- **192.168.0.198:5443** is the IP address of the cluster API server.
- `/api/v1/namespaces/default/pods/` specifies the URI of the cluster API for viewing pod information in the **default** namespace. For more cluster APIs, see [Kubernetes API](#).

----End

Calling Kubernetes APIs Through API Gateway

You can call Kubernetes-native APIs through API Gateway using the URL in the format of **https://{clusterid}.Endpoint/uri**. In the URL, `{clusterid}` indicates a cluster ID, and `uri` indicates the resource path, that is, the path for API access.

Table 5-1 URL parameters

Parameter	Description
{clusterid}	Cluster ID. After a cluster is created, call the API for obtaining a cluster in a specified project to obtain the cluster ID.
Endpoint	Entry (URL) for a web service, which can be obtained from Endpoints .
uri	Access path of an API for performing an operation. Obtain the value from the URI of the API. For details, see Kubernetes API .

- Step 1** Obtain the token of the region where the cluster is located. For details about how to obtain a token, see [Obtaining a Token](#).
- Step 2** Obtain the cluster ID using either of the following methods:
- Method 1: Obtain the cluster UID by calling the API for obtaining cluster information.
 - Method 2: Obtain the cluster ID on the **Overview** page of the CCE console.
- Step 3** Determine the requested URL based on the URL format **https://{clusterid}.Endpoint/uri**.
- **{clusterid}**: Obtain the value by using [Step 2](#).
 - **Endpoint**: Obtain the endpoint from [Regions and Endpoints](#).
For example, the endpoint of CCE in the **AP-Singapore** region is **cce.ap-southeast-3.myhuaweicloud.com**.
 - **uri**: Set this parameter based on the API to be called. For example, if you want to create a Deployment, the request method is POST and the API URI is **/apis/apps/v1/namespaces/{namespace}/deployments**, where **{namespace}** indicates the cluster namespace name. In this example, the value is **default**.
For more APIs, see [Kubernetes APIs](#).

Combine the preceding parameters following the URL format **https://{clusterid}.Endpoint/uri**.

The following is an example of the URL for calling the API to view information about all pods:

```
https://07da5****.cce.ap-southeast-3.myhuaweicloud.com/apis/apps/v1/namespaces/default/deployments
```

- Step 4** Use the request method specified by the API and set the request header parameters. If parameters in the body need to be added, add the structure corresponding to the API by referring to [Kubernetes APIs](#).

Example curl command to call the API for creating a Deployment using POST and adding the corresponding body:

In this example, the **nginx.json** file is used to create a Deployment named **nginx**. The Deployment uses the **nginx:latest** image and contains two pods. Each pod occupies 100m CPU and 200 MiB memory.

```
curl --location --request POST 'https://07da5****.cce.ap-southeast-3.myhuaweicloud.com/apis/apps/v1/namespaces/default/deployments' \
--header 'Content-Type: application/json' \
--header 'X-Auth-Token: MIIWVW*****' \
--data @nginx.json
```

Header parameters contained in the request are as follows:

Table 5-2 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format), for example, application/json.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details about how to obtain a token, see Obtaining a Token .

nginx.json is located in the current directory and contains the following content:

```
{
  "apiVersion": "apps/v1",
  "kind": "Deployment",
  "metadata": {
    "name": "nginx"
  },
  "spec": {
    "replicas": 2,
    "selector": {
      "matchLabels": {
        "app": "nginx"
      }
    },
    "template": {
      "metadata": {
        "labels": {
          "app": "nginx"
        }
      },
      "spec": {
        "containers": [
          {
            "image": "nginx:latest",
            "name": "container-0",
            "resources": {
              "limits": {
                "cpu": "100m",
                "memory": "200Mi"
              },
              "requests": {
                "cpu": "100m",
                "memory": "200Mi"
              }
            }
          }
        ],
        "imagePullSecrets": [
          {
            "name": "default-secret"
          }
        ]
      }
    }
  }
}
```

----End

Related Documents

- [Accessing a Cluster Using Kubernetes APIs](#)
- [Kubernetes official SDKs](#) (including Go, Python, and Java)

Language	Client Library	Sample Program
C	github.com/kubernetes-client/c	Browse
dotnet	github.com/kubernetes-client/csharp	Browse
Go	github.com/kubernetes/client-go/	Browse
Haskell	github.com/kubernetes-client/haskell	Browse
Java	github.com/kubernetes-client/java	Browse
JavaScript	github.com/kubernetes-client/javascript	Browse
Perl	github.com/kubernetes-client/perl/	Browse
Python	github.com/kubernetes-client/python/	Browse
Ruby	github.com/kubernetes-client/ruby/	Browse

6 Out-of-Date APIs

6.1 Obtaining Cluster Certificates (Deprecated)

Function

This API is used to obtain certificates of a specified cluster. This API has been deprecated. Use [this API](#) to obtain the cluster certificate instead.

URI

GET /api/v3/projects/{project_id}/clusters/{cluster_id}/clustercert

Table 6-1 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain a project ID, see How to Obtain Parameters in the API URI .
cluster_id	Yes	String	Cluster ID. For details about how to obtain a cluster ID, see How to Obtain Parameters in the API URI .

Request

Table 6-2 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Possible values: • application/json; charset=utf-8 • application/json
X-Auth-Token	Yes	String	Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details about how to obtain a token, see Authentication . Maximum length: 16,384 characters

Response

Status code: 200**Table 6-3** Response body parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at Config and cannot be changed. Default value: Config
apiVersion	String	API version. The value is fixed to v1 .
preferences	String	This field is not used currently and is left unspecified by default.
clusters	Array of Clusters objects	Cluster list.
users	Array of Users objects	Stores the certificate information and client key information of a specified user.
contexts	Array of Contexts objects	Context list.

Parameter	Type	Description
current-context	String	Current context. If publicip (elastic IP address of the VM) exists, the value is external . If publicip does not exist, the value is internal .

Table 6-4 Clusters

Parameter	Type	Description
name	String	Cluster name. - If the publicip parameter does not exist (that is, no EIP exists), there is only one cluster in the cluster list, and the value of this parameter is internalCluster. - If the publicip parameter exists (that is, the EIP exists), there are at least two clusters in the cluster list, and the value of this parameter is externalCluster.
cluster	ClusterCert object	Cluster information.

Table 6-5 ClusterCert

Parameter	Type	Description
server	String	Server IP address.
certificate-authority-data	String	Certificate authorization data.
insecure-skip-tls-verify	Boolean	Whether to skip the server certificate verification. If the cluster type is externalCluster , the value is true .

Table 6-6 Users

Parameter	Type	Description
name	String	The value is fixed to user .
user	User object	Stores the certificate information and client key information of a specified user.

Table 6-7 User

Parameter	Type	Description
client-certificate-data	String	Client certificate.
client-key-data	String	PEM encoding data from the TLS client key file.

Table 6-8 Contexts

Parameter	Type	Description
name	String	Context name. - If the publicIp parameter does not exist (that is, no EIP exists), there is only one cluster in the cluster list, and the value of this parameter is internal. - If the publicIp parameter exists (that is, the EIP exists), there are at least two clusters in the cluster list, and the value of this parameter is external.
context	Context object	Context information.

Table 6-9 Context

Parameter	Type	Description
cluster	String	Cluster context.
user	String	User context.

Example Request

None

Example Response

Status code: 200

Certificates of the specified cluster are successfully obtained. For details about the certificate file format, see the following structure of **Config** (Kubernetes v1).

```
{
  "kind": "Config",
  "apiVersion": "v1",
  "preferences": { },
}
```



```
"clusters" : [ {  
  "name" : "internalCluster",  
  "cluster" : {  
    "server" : "https://192.168.1.7:5443",  
    "certificate-authority-data" : "Q2VydGlmaWN*****kQgQ0VSVElGSUNBVEUtLS0tLQo="
```

Status Code

Status code	Description
200	Certificates of the specified cluster are successfully obtained. For details about the certificate file format, see the structure of Config (Kubernetes v1).

Error Codes

For details, see [Error Codes](#).

6.2 Creating a PV (Deprecated)

Function

This API is used to create a PersistentVolume associated with cloud storage (for example, EVS, SFS, and OBS). This API has been deprecated. Use the corresponding Kubernetes PV API instead.

NOTE

The format of the storage management URL is **https://{clusterid}.Endpoint/uri**. In the URL, *{clusterid}* indicates a cluster ID, and *uri* indicates the resource path, that is, the path for API access. If **https://Endpoint/uri** is used, the **X-Cluster-ID** parameter must be specified in the request header.

URI

POST /api/v1/cloudpersistentvolumes

Request

Table 6-10 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Possible values: • application/json; charset=utf-8 • application/json
X-Auth-Token	Yes	String	Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details about how to obtain a token, see Authentication . Maximum length: 16,384 characters
X-Cluster-ID	No	String	Cluster ID. This parameter is mandatory when the URL format https://Endpoint/uri is used. For details about how to obtain the value, see How to Obtain Parameters in the API URI .

Table 6-11 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	API version. The value is fixed to v1 . Default value: v1
kind	Yes	String	API type. The value is fixed to PersistentVolume . Default value: PersistentVolume
metadata	Yes	PersistentVolumeMetadata object	PersistentVolume metadata information.

Parameter	Mandatory	Type	Description
spec	Yes	PersistentVolumeSpec object	PersistentVolume specifications.
status	No	PersistentVolumeStatus object	PersistentVolume status information.

Table 6-12 PersistentVolumeMetadata

Parameter	Mandatory	Type	Description
name	Yes	String	PV name. Enter 1 to 253 characters starting and ending with a letter or digit. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. PV names must be unique in a namespace.
labels	No	Map<String,String>	PV labels, in the format of key-value pairs. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key and contain a maximum of 253 characters. Example DNS subdomain: example.com/my-key • Value: The value can be left blank or contain 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed in the character string. Example label: "foo": "bar"

Table 6-13 PersistentVolumeSpec

Parameter	Mandatory	Type	Description
flexVolume	Yes	FlexVolume object	FlexVolume storage plug-in of Kubernetes.
persistentVolumeReclaimPolicy	No	String	PV reclaim policy. Possible values: • Retain: The Retain reclaim policy allows for manual reclamation of the resource. When the PVC is deleted, the PV still exists and the volume is considered "released". • Recycle: The Recycle reclaim policy performs a basic scrub (<code>rm -rf /thevolume / *</code>) on the volume and makes it available again for a new claim. • Delete: For volume plug-ins that support the Delete reclaim policy, deletion removes both the PV object from Kubernetes, as well as the associated storage asset in the external infrastructure. Volumes that were dynamically provisioned inherit the reclaim policy of their StorageClass, which defaults to Delete.

Parameter	Mandatory	Type	Description
accessModes	Yes	Array of strings	Access mode of the PersistentVolume. • ReadWriteOnce: The volume can be mounted as read-write by a single node. NOTE This function is supported only when the cluster version is v1.13.10 and the storage-driver version is 1.0.19. • ReadOnlyMany: The volume can be mounted as read-only by many nodes. • ReadWriteMany: The volume can be mounted as read-write by many nodes.

Table 6-14 FlexVolume

Parameter	Mandatory	Type	Description
driver	Yes	String	Name of the FlexVolume storage plug-in. Set this parameter based on the storage volume type. • huawei.com/fuxivol (EVS) • huawei.com/fuxinfs (SFS) • huawei.com/fuxiobs (OBS) • huawei.com/fuxiefs (SFS Turbo)

Parameter	Mandatory	Type	Description
fsType	Yes	String	File system type. Set this parameter based on the storage volume type. • ext4: EVS volume. For details, see Using EVS Disks as Storage Volumes. • nfs: SFS volume. For details, see Using SFS File Systems as Storage Volumes. • obs: OBS volume. For details, see Using OBS Buckets as Storage Volumes. • efs: SFS Turbo volume. For details, see Using SFS Turbo File Systems as Storage Volumes.
options	Yes	Options object	FlexVolume configuration items.

Table 6-15 Options

Parameter	Mandatory	Type	Description
fsType	Yes	String	File system type. Set this parameter based on the storage volume type. • ext4 (EVS) • nfs (SFS) • obs (OBS) • efs (SFS Turbo)
region	Yes	String	Region where the cloud storage is located.
volumeID	Yes	String	UUID of the cloud storage. If the cloud storage is an OBS bucket, set this parameter to the bucket name.

Parameter	Mandatory	Type	Description
storageType	Yes	String	Cloud storage volume type. • bs (EVS) • nfs (SFS) • obs (OBS) • efs (SFS Turbo)

Table 6-16 PersistentVolumeStatus

Parameter	Mandatory	Type	Description
accessModes	No	Array of strings	Access mode of the PersistentVolume.
phase	No	String	PV status. Possible values: • Available: The volume is not bound to any PVC. • Bound: The PV has been bound to a PVC. • Released: The PVC bound to the PV has been deleted, but the PV has not been made available again for a new claim in the cluster. • Failed: Automatic reclaim of the volume has failed.

Response

Status code: 201

Table 6-17 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. The value is fixed to v1 . Default value: v1
kind	String	API type. The value is fixed to PersistentVolume . Default value: PersistentVolume
metadata	PersistentVolumeMetadata object	PersistentVolume metadata information.

Parameter	Type	Description
spec	PersistentVolumeSpec object	PersistentVolume specifications.
status	PersistentVolumeStatus object	PersistentVolume status information.

Table 6-18 PersistentVolumeMetadata

Parameter	Type	Description
name	String	PV name. Enter 1 to 253 characters starting and ending with a letter or digit. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. PV names must be unique in a namespace.
labels	Map<String,String>	PV labels, in the format of key-value pairs. • Key: Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key and contain a maximum of 253 characters. Example DNS subdomain: example.com/my-key • Value: The value can be left blank or contain 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed in the character string. Example label: "foo": "bar"

Table 6-19 PersistentVolumeSpec

Parameter	Type	Description
flexVolume	FlexVolume object	FlexVolume storage plug-in of Kubernetes.

Parameter	Type	Description
<code>persistentVolumeReclaimPolicy</code>	String	PV reclaim policy. Possible values: ● Retain: The Retain reclaim policy allows for manual reclamation of the resource. When the PVC is deleted, the PV still exists and the volume is considered "released". ● Recycle: The Recycle reclaim policy performs a basic scrub (<code>rm -rf / thevolume / *</code>) on the volume and makes it available again for a new claim. ● Delete: For volume plug-ins that support the Delete reclaim policy, deletion removes both the PV object from Kubernetes, as well as the associated storage asset in the external infrastructure. Volumes that were dynamically provisioned inherit the reclaim policy of their StorageClass, which defaults to Delete.
<code>accessModes</code>	Array of strings	Access mode of the PersistentVolume. ● ReadWriteOnce: The volume can be mounted as read-write by a single node. NOTE This function is supported only when the cluster version is v1.13.10 and the storage-driver version is 1.0.19. ● ReadOnlyMany: The volume can be mounted as read-only by many nodes. ● ReadWriteMany: The volume can be mounted as read-write by many nodes.

Table 6-20 FlexVolume

Parameter	Type	Description
<code>driver</code>	String	Name of the FlexVolume storage plug-in. Set this parameter based on the storage volume type. ● <code>huawei.com/fuxivol</code> (EVS) ● <code>huawei.com/fuxinfs</code> (SFS) ● <code>huawei.com/fuxiobs</code> (OBS) ● <code>huawei.com/fuxiefs</code> (SFS Turbo)

Parameter	Type	Description
fsType	String	File system type. Set this parameter based on the storage volume type. • ext4: EVS volume. For details, see Using EVS Disks as Storage Volumes. • nfs: SFS volume. For details, see Using SFS File Systems as Storage Volumes. • obs: OBS volume. For details, see Using OBS Buckets as Storage Volumes. • efs: SFS Turbo volume. For details, see Using SFS Turbo File Systems as Storage Volumes.
options	Options object	FlexVolume configuration items.

Table 6-21 Options

Parameter	Type	Description
fsType	String	File system type. Set this parameter based on the storage volume type. • ext4 (EVS) • nfs (SFS) • obs (OBS) • efs (SFS Turbo)
region	String	Region where the cloud storage is located.
volumeID	String	UUID of the cloud storage. If the cloud storage is an OBS bucket, set this parameter to the bucket name.
storageType	String	Cloud storage volume type. • bs (EVS) • nfs (SFS) • obs (OBS) • efs (SFS Turbo)

Table 6-22 PersistentVolumeStatus

Parameter	Type	Description
accessModes	Array of strings	Access mode of the PersistentVolume.

Parameter	Type	Description
phase	String	PV status. Possible values: ● Available: The volume is not bound to any PVC. ● Bound: The PV has been bound to a PVC. ● Released: The PVC bound to the PV has been deleted, but the PV has not been made available again for a new claim in the cluster. ● Failed: Automatic reclaim of the volume has failed.

Example Request

- Specifying an EVS volume ID and creating a PersistentVolume:

```
{
  "apiVersion": "v1",
  "kind": "PersistentVolume",
  "metadata": {
    "labels": {
      "name": "pv-test"
    },
    "name": "pv-test"
  },
  "spec": {
    "accessModes": [ "ReadWriteMany" ],
    "flexVolume": {
      "driver": "huawei.com/fuxivol",
      "fsType": "ext4",
      "options": {
        "fsType": "ext4",
        "kubernetes.io/namespace": "default",
        "region": "southchina",
        "volumeID": "76e01b29-08b9-11e8-9ca5-1051722006ec",
        "storageType": "bs"
      }
    },
    "persistentVolumeReclaimPolicy": "Delete"
  }
}
```

- ```
{
 "apiVersion": "v1",
 "kind": "PersistentVolume",
 "metadata": {
 "labels": {
 "name": "pv-test"
 },
 "name": "pv-test"
 },
 "spec": {
 "accessModes": ["ReadWriteMany"],
 "flexVolume": {
 "driver": "huawei.com/fuxivol",
 "fsType": "ext4",
 "options": {
 "fsType": "ext4",
 "region": "southchina",
 "volumeID": "76e01b29-08b9-11e8-9ca5-1051722006ec",
 "storageType": "bs"
 }
 }
 }
}
```

```
 },
 "persistentVolumeReclaimPolicy" : "Delete"
 }
}
```

## Example Response

### Status code: 201

The job for creating a PersistentVolume is successfully delivered.

```
{
 "kind" : "PersistentVolume",
 "apiVersion" : "v1",
 "metadata" : {
 "name" : "pv-test",
 "namespace" : "default",
 "selfLink" : "/api/v1/namespaces/default/persistentvolumes/pv-test",
 "uid" : "e174188f-ff21-11e7-855b-fa163eaf5675",
 "resourceVersion" : "174229",
 "creationTimestamp" : "2018-01-22T03:11:03Z",
 "labels" : {
 "name" : "pv-test"
 },
 "enable" : true
 },
 "spec" : {
 "capacity" : {
 "storage" : "1Gi"
 },
 "accessModes" : ["ReadWriteMany"],
 "flexVolume" : {
 "driver" : "huawei.com/fuxivol",
 "fsType" : "ext4",
 "options" : {
 "fsType" : "ext4",
 "kubernetes.io/namespace" : "default",
 "volumeID" : "0781b22f-4d89-4e9c-b026-80e545cea16c"
 }
 },
 "persistentVolumeReclaimPolicy" : "Delete"
 },
 "status" : {
 "phase" : "Pending"
 }
}
```

## Status Code

| Status Code | Description                                                        |
|-------------|--------------------------------------------------------------------|
| 201         | The job for creating a PersistentVolume is successfully delivered. |

## Error Codes

For details, see [Error Codes](#).

## 6.3 Deleting a PV (Deprecated)

### Function

This API is used to delete a PersistentVolume from a specified namespace. Using this API, you can determine whether to retain the cloud storage associated with the PersistentVolume. This API has been deprecated. Use the corresponding Kubernetes PV API instead.

#### NOTE

The format of the storage management URL is **https://{clusterid}.Endpoint/uri**. In the URL, *{clusterid}* indicates a cluster ID, and *uri* indicates the resource path, that is, the path for API access. If **https://Endpoint/uri** is used, the **X-Cluster-ID** parameter must be specified in the request header.

### URI

DELETE /api/v1/cloudpersistentvolumes/{name}

**Table 6-23** URI parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name      | Yes       | String | <p>Name of the PersistentVolume to be deleted.</p> <p>Name format: <i>Volume name</i>[?deleteVolume=<i>BOOLEAN</i>&amp;storageType=<i>Cloud storage volume type</i>]. The content in square brackets can be omitted. Examples:</p> <ul style="list-style-type: none"><li>• volume-49f1?deleteVolume=true&amp;storageType=bs</li><li>• volume-49f1</li></ul> <p>In the first example:</p> <p><b>deleteVolume:</b> Whether to delete associated backend cloud storage after the PersistentVolume is deleted. The value can be <b>true</b> or <b>false</b> and defaults to <b>false</b>. If it is set to <b>true</b>, it indicates that the associated backend cloud storage is deleted after the PersistentVolume is deleted. If it is set to <b>false</b>, it indicates that the associated backend cloud storage is not deleted after the PersistentVolume is deleted.</p> <p><b>NOTE</b></p> <p>If the storage volume type is <b>efs</b>, this parameter must be set to <b>false</b>.</p> <p><b>storageType:</b> Cloud storage volume type. This parameter is used together with <b>deleteVolume</b>. That is, <b>deleteVolume</b> and <b>storageType</b> must be configured at the same time.</p> <p><b>NOTE</b></p> <ul style="list-style-type: none"><li>• <b>bs:</b> EVS</li><li>• <b>nfs:</b> SFS</li><li>• <b>obs:</b> OBS</li><li>• <b>efs:</b> SFS Turbo</li></ul> |

## Request

**Table 6-24** Request header parameters

| Parameter    | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                |
|--------------|-----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Content-Type | Yes       | String | Message body type (format).<br>Possible values: <ul style="list-style-type: none"><li>• <b>application/json; charset=utf-8</b></li><li>• <b>application/json</b></li></ul>                                                                                                                                 |
| X-Auth-Token | Yes       | String | Requests for calling an API can be authenticated using either a token or AK/SK. If token-based authentication is used, this parameter is mandatory and must be set to a user token. For details about how to obtain a token, see <a href="#">Authentication</a> .<br><br>Maximum length: 16,384 characters |
| X-Cluster-ID | No        | String | Cluster ID. This parameter is mandatory when the URL format <b>https://Endpoint/uri</b> is used. For details about how to obtain the value, see <a href="#">How to Obtain Parameters in the API URI</a> .                                                                                                  |

## Response

**Status code: 200****Table 6-25** Response body parameters

| Parameter  | Type                                            | Description                                                                                  |
|------------|-------------------------------------------------|----------------------------------------------------------------------------------------------|
| apiVersion | String                                          | API version. The value is fixed to <b>v1</b> .<br>Default value: v1                          |
| kind       | String                                          | API type. The value is fixed to <b>PersistentVolume</b> .<br>Default value: PersistentVolume |
| metadata   | <a href="#">PersistentVolumeMetadata</a> object | PersistentVolume metadata information.                                                       |

| Parameter | Type                                          | Description                          |
|-----------|-----------------------------------------------|--------------------------------------|
| spec      | <a href="#">PersistentVolumeSpec</a> object   | PersistentVolume specifications.     |
| status    | <a href="#">PersistentVolumeStatus</a> object | PersistentVolume status information. |

**Table 6-26** PersistentVolumeMetadata

| Parameter | Type               | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-----------|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name      | String             | PV name. Enter 1 to 253 characters starting and ending with a letter or digit. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed. PV names must be unique in a namespace.                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| labels    | Map<String,String> | <p>PV labels, in the format of key-value pairs.</p> <ul style="list-style-type: none"><li>• <b>Key:</b> Enter 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. A DNS subdomain can be prefixed to a key and contain a maximum of 253 characters. Example DNS subdomain: example.com/my-key</li><li>• <b>Value:</b> The value can be left blank or contain 1 to 63 characters starting with a letter or digit. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed in the character string.</li></ul> <p>Example label: "foo": "bar"</p> |

**Table 6-27** PersistentVolumeSpec

| Parameter  | Type                              | Description                               |
|------------|-----------------------------------|-------------------------------------------|
| flexVolume | <a href="#">FlexVolume</a> object | FlexVolume storage plug-in of Kubernetes. |



| Parameter                                  | Type             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------------------------------------------|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>persistentVolumeReclaimPolicy</code> | String           | <p>PV reclaim policy. Possible values:</p> <ul style="list-style-type: none"><li>● <b>Retain</b>: The Retain reclaim policy allows for manual reclamation of the resource. When the PVC is deleted, the PV still exists and the volume is considered "released".</li><li>● <b>Recycle</b>: The Recycle reclaim policy performs a basic scrub (<code>rm -rf / thevolume / *</code>) on the volume and makes it available again for a new claim.</li><li>● <b>Delete</b>: For volume plug-ins that support the Delete reclaim policy, deletion removes both the PV object from Kubernetes, as well as the associated storage asset in the external infrastructure. Volumes that were dynamically provisioned inherit the reclaim policy of their StorageClass, which defaults to <b>Delete</b>.</li></ul> |
| <code>accessModes</code>                   | Array of strings | <p>Access mode of the PersistentVolume.</p> <ul style="list-style-type: none"><li>● <b>ReadWriteOnce</b>: The volume can be mounted as read-write by a single node.</li></ul> <p><b>NOTE</b><br/>This function is supported only when the cluster version is v1.13.10 and the storage-driver version is 1.0.19.</p> <ul style="list-style-type: none"><li>● <b>ReadOnlyMany</b>: The volume can be mounted as read-only by many nodes.</li><li>● <b>ReadWriteMany</b>: The volume can be mounted as read-write by many nodes.</li></ul>                                                                                                                                                                                                                                                                 |

**Table 6-28** FlexVolume

| Parameter           | Type   | Description                                                                                                                                                                                                                                                                                                                                       |
|---------------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>driver</code> | String | <p>Name of the FlexVolume storage plug-in. Set this parameter based on the storage volume type.</p> <ul style="list-style-type: none"><li>● <code>huawei.com/fuxivol</code> (EVS)</li><li>● <code>huawei.com/fuxinfs</code> (SFS)</li><li>● <code>huawei.com/fuxiobs</code> (OBS)</li><li>● <code>huawei.com/fuxiefs</code> (SFS Turbo)</li></ul> |

| Parameter | Type                           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-----------|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| fsType    | String                         | File system type. Set this parameter based on the storage volume type. <ul style="list-style-type: none"><li>• <b>ext4</b>: EVS volume. For details, see <a href="#">Using EVS Disks as Storage Volumes</a>.</li><li>• <b>nfs</b>: SFS volume. For details, see <a href="#">Using SFS File Systems as Storage Volumes</a>.</li><li>• <b>obs</b>: OBS volume. For details, see <a href="#">Using OBS Buckets as Storage Volumes</a>.</li><li>• <b>efs</b>: SFS Turbo volume. For details, see <a href="#">Using SFS Turbo File Systems as Storage Volumes</a>.</li></ul> |
| options   | <a href="#">Options</a> object | FlexVolume configuration items.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

**Table 6-29** Options

| Parameter   | Type   | Description                                                                                                                                                                                           |
|-------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| fsType      | String | File system type. Set this parameter based on the storage volume type. <ul style="list-style-type: none"><li>• ext4 (EVS)</li><li>• nfs (SFS)</li><li>• obs (OBS)</li><li>• efs (SFS Turbo)</li></ul> |
| region      | String | Region where the cloud storage is located.                                                                                                                                                            |
| volumeID    | String | UUID of the cloud storage. If the cloud storage is an OBS bucket, set this parameter to the bucket name.                                                                                              |
| storageType | String | Cloud storage volume type. <ul style="list-style-type: none"><li>• bs (EVS)</li><li>• nfs (SFS)</li><li>• obs (OBS)</li><li>• efs (SFS Turbo)</li></ul>                                               |

**Table 6-30** PersistentVolumeStatus

| Parameter   | Type             | Description                          |
|-------------|------------------|--------------------------------------|
| accessModes | Array of strings | Access mode of the PersistentVolume. |

| Parameter | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                               |
|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| phase     | String | <p>PV status. Possible values:</p> <ul style="list-style-type: none"><li>● <b>Available:</b> The volume is not bound to any PVC.</li><li>● <b>Bound:</b> The PV has been bound to a PVC.</li><li>● <b>Released:</b> The PVC bound to the PV has been deleted, but the PV has not been made available again for a new claim in the cluster.</li><li>● <b>Failed:</b> Automatic reclaim of the volume has failed.</li></ul> |

## Example Request

None

## Example Response

**Status code: 200**

The job for deleting a PersistentVolume is successfully delivered.

```
{
 "kind": "PersistentVolume",
 "apiVersion": "v1",
 "metadata": {
 "name": "pv-test",
 "selfLink": "/api/v1/persistentvolumes/pv-test",
 "uid": "0d93181d-3628-11e7-9093-fa163e8c373b",
 "resourceVersion": "180886",
 "creationTimestamp": "2017-05-11T08:58:51Z",
 "labels": {
 "app": "test"
 }
 },
 "spec": {
 "flexVolume": {
 "driver": "huawei.com/fuxivol",
 "fsType": "ext4",
 "options": {
 "fsType": "ext4",
 "kubernetes.io/namespace": "default",
 "volumeID": "0781b22f-4d89-4e9c-b026-80e545cea16c"
 }
 },
 "capacity": {
 "storage": "1Gi"
 },
 "accessModes": ["ReadWriteMany"],
 "persistentVolumeReclaimPolicy": "Delete"
 },
 "status": {
 "phase": "Available"
 }
}
```

## Status Code

| Status Code | Description                                                        |
|-------------|--------------------------------------------------------------------|
| 200         | The job for deleting a PersistentVolume is successfully delivered. |

## Error Codes

For details, see [Error Codes](#).

# 7 Permissions and Supported Actions

---

You can use Identity and Access Management (IAM) for fine-grained permissions management of your CCE clusters. If your account does not need individual IAM users, you can skip this section.

New IAM users do not have any permissions assigned by default. You need to first add them to one or more groups and attach policies or roles to these groups. Users inherit permissions from the groups to which they are added and can perform specific operations on cloud services based on the assigned permissions. For more information about policy syntax and example policies, see [Permissions Overview](#).

You can grant users permissions by using [roles](#) and [policies](#). Roles are provided by IAM to define service-based permissions that match users' job responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

## NOTE

If you want to allow or deny the access to an API, fine-grained authorization is a good choice.

An account has all of the permissions required to call all APIs, but IAM users must have the required permissions specifically assigned. The required permissions are determined by the actions supported by the API. Only users with the permissions allowing for those actions can call the API successfully. For example, if an IAM user wants to query ECSs using an API, the user must have been granted permissions that allow the **ecs:servers:list** action.

## Supported Actions

CCE provides system-defined policies that can be directly used in IAM. You can also create custom policies to supplement system-defined policies for more refined access control. Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: statements in a policy that allow or deny certain operations.
- APIs: REST APIs that can be called by a user who has been granted specific permissions.

- **Actions:** specific operations that are allowed or denied in a custom policy.
- **Dependencies:** actions which a specific action depends on. When allowing an action for a user, you also need to allow any existing action dependencies for that user.
- **IAM projects/Enterprise projects:** the authorization scope of a custom policy. A custom policy can be applied to IAM projects or enterprise projects or both. Policies that contain actions for both IAM and enterprise projects can be used and applied for both IAM and Enterprise Management. Policies that contain actions only for IAM projects can be used and applied to IAM only. For details about the differences between IAM and enterprise management, see [What Are the Differences Between IAM and Enterprise Management?](#)

 **NOTE**

The check mark (√) and cross symbol (x) respectively indicate that an action takes effect or does not take effect for the corresponding type of projects.

CCE supports the following actions in custom policies.

**Table 7-1** Cluster management actions

| Permission                      | API                                                                        | Action                  | IAM Project | Enterprise Project |
|---------------------------------|----------------------------------------------------------------------------|-------------------------|-------------|--------------------|
| Obtaining clusters in a project | GET /api/v3/projects/{project_id}/clusters                                 | cce:cluster:list        | √           | √                  |
| Obtaining a cluster             | GET /api/v3/projects/{project_id}/clusters/{cluster_id}                    | cce:cluster:get         | √           | √                  |
| Creating a cluster              | POST /api/v3/projects/{project_id}/clusters                                | cce:cluster:create      | √           | √                  |
| Updating a cluster              | PUT /api/v3/projects/{project_id}/clusters/{cluster_id}                    | cce:cluster:update      | √           | √                  |
| Deleting a cluster              | DELETE /api/v3/projects/{project_id}/clusters/{cluster_id}                 | cce:cluster:delete      | √           | √                  |
| Upgrading a cluster             | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/upgrade | cce:cluster:updategrade | √           | √                  |
| Waking up a cluster             | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/awake   | cce:cluster:start       | √           | √                  |

| Permission                               | API                                                                          | Action                 | IAM Project | Enterprise Project |
|------------------------------------------|------------------------------------------------------------------------------|------------------------|-------------|--------------------|
| Hibernating a cluster                    | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/hibernate | cce:cluster:s<br>top   | √           | √                  |
| Changing the specifications of a cluster | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/operation/resize    | cce:cluster:r<br>esize | √           | √                  |
| Obtaining the certificate of a cluster   | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/clustercert         | cce:cluster:g<br>et    | √           | √                  |

**Table 7-2** Node management actions

| Permission                       | API                                                                     | Action              | IAM Project | Enterprise Project                                                                                                                                                                |
|----------------------------------|-------------------------------------------------------------------------|---------------------|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Obtaining all nodes in a cluster | GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes           | cce:node:list       | √           | √                                                                                                                                                                                 |
| Obtaining a node                 | GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/{node_id} | cce:node:ge<br>t    | √           | √                                                                                                                                                                                 |
| Creating a node                  | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes          | cce:node:cre<br>ate | √           | √<br><br><b>NOTE</b><br>If you use enterprise project authorization to create a node, you need to add the global permission of <b>evs:quotas: get</b> and <b>evs:types: get</b> . |
| Updating a node                  | PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/{node_id} | cce:node:up<br>date | √           | √                                                                                                                                                                                 |

| Permission       | API                                                                                                 | Action           | IAM Project | Enterprise Project |
|------------------|-----------------------------------------------------------------------------------------------------|------------------|-------------|--------------------|
| Deleting a node  | DELETE /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/{node_id}                          | cce:node:delete  | √           | √                  |
| Removing a node  | PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/operation/remove                      | cce:node:remove  | √           | √                  |
| Migrating a node | PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodes/operation/migrate/{target_cluster_id} | cce:node:migrate | √           | √                  |

**Table 7-3** Job management actions

| Permission                        | API                                                                                             | Action         | IAM Project | Enterprise Project |
|-----------------------------------|-------------------------------------------------------------------------------------------------|----------------|-------------|--------------------|
| Obtaining information about a job | GET /api/v3/projects/{project_id}/jobs/{job_id}                                                 | cce:job:get    | √           | √                  |
| Listing all jobs                  | GET /api/v2/projects/{project_id}/jobs                                                          | cce:job:list   | √           | √                  |
| Deleting one or all jobs          | DELETE /api/v2/projects/{project_id}/jobs<br>DELETE /api/v2/projects/{project_id}/jobs/{job_id} | cce:job:delete | √           | √                  |

**Table 7-4** Node pool management actions

| Permission                            | API                                                               | Action            | IAM Project | Enterprise Project |
|---------------------------------------|-------------------------------------------------------------------|-------------------|-------------|--------------------|
| Obtaining all node pools in a cluster | GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools | cce:nodepool:list | √           | √                  |



| Permission            | API                                                                                              | Action              | IAM Project | Enterprise Project |
|-----------------------|--------------------------------------------------------------------------------------------------|---------------------|-------------|--------------------|
| Obtaining a node pool | GET /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}                  | cce:nodepool:get    | √           | √                  |
| Creating a node pool  | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools                               | cce:nodepool:create | √           | √                  |
| Updating a node pool  | PUT /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}                  | cce:nodepool:update | √           | √                  |
| Deleting a node pool  | DELETE /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}               | cce:nodepool:delete | √           | √                  |
| Scaling a node pool   | POST /api/v3/projects/{project_id}/clusters/{cluster_id}/nodepools/{nodepool_id}/operation/scale | cce:nodepool:scale  | √           | √                  |

**Table 7-5** Chart management actions

| Permission                          | API                    | Action           | IAM Project | Enterprise Project |
|-------------------------------------|------------------------|------------------|-------------|--------------------|
| Updating a chart                    | PUT /v2/charts/{id}    | cce:chart:update | √           | ×                  |
| Uploading a chart                   | POST /v2/charts        | cce:chart:upload | √           | ×                  |
| Listing all charts                  | GET /v2/charts         | cce:chart:list   | √           | ×                  |
| Obtaining information about a chart | GET /v2/charts/{id}    | cce:chart:get    | √           | ×                  |
| Deleting a chart                    | DELETE /v2/charts/{id} | cce:chart:delete | √           | ×                  |

**Table 7-6** Release management actions

| Permission                            | API                        | Action             | IAM Project | Enterprise Project |
|---------------------------------------|----------------------------|--------------------|-------------|--------------------|
| Updating a release                    | PUT /v2/releases/{name}    | cce:release:update | √           | √                  |
| Listing all releases                  | GET /v2/releases           | cce:release:list   | √           | √                  |
| Creating a release                    | POST /v2/releases          | cce:release:create | √           | √                  |
| Obtaining information about a release | GET /v2/releases/{name}    | cce:release:get    | √           | √                  |
| Deleting a release                    | DELETE /v2/releases/{name} | cce:release:delete | √           | √                  |

**Table 7-7** Storage management actions

| Permission                       | API                                                                      | Action             | IAM Project | Enterprise Project |
|----------------------------------|--------------------------------------------------------------------------|--------------------|-------------|--------------------|
| Creating a PersistentVolumeClaim | POST /api/v1/namespaces/{namespace}/cloudpersistentvolumeclaims          | cce:storage:create | √           | √                  |
| Deleting a PersistentVolumeClaim | DELETE /api/v1/namespaces/{namespace}/cloudpersistentvolumeclaims/{name} | cce:storage:delete | √           | √                  |
| Listing all volumes              | GET /storage/api/v1/namespaces/{namespace}/listvolumes                   | cce:storage:list   | √           | √                  |

**Table 7-8** Add-on management actions

| Permission                  | API                 | Action                   | IAM Project | Enterprise Project |
|-----------------------------|---------------------|--------------------------|-------------|--------------------|
| Creating an add-on instance | POST /api/v3/addons | cce:addonInstance:create | √           | √                  |

| Permission                      | API                                                | Action                     | IAM Project | Enterprise Project |
|---------------------------------|----------------------------------------------------|----------------------------|-------------|--------------------|
| Obtaining an add-on instance    | GET /api/v3/addons/{id}?cluster_id={cluster_id}    | cce:addonInstance:get      | √           | √                  |
| Listing all add-on instances    | GET /api/v3/addons?cluster_id={cluster_id}         | cce:addonInstance:list     | √           | √                  |
| Deleting an add-on instance     | DELETE /api/v3/addons/{id}?cluster_id={cluster_id} | cce:addonInstance:delete   | √           | √                  |
| Updating an add-on instance     | PUT /api/v3/addons/{id}                            | cce:addonInstance:update   | √           | √                  |
| Rolling back an add-on instance | POST /api/v3/addons/{id}/operation/rollback        | cce:addonInstance:rollback | √           | √                  |

**Table 7-9** Quota management actions

| Permission              | API                                      | Action        | IAM Project | Enterprise Project |
|-------------------------|------------------------------------------|---------------|-------------|--------------------|
| Obtaining quota details | GET /api/v3/projects/{project_id}/quotas | cce:quota:get | √           | √                  |

# 8 Appendix

## 8.1 Status Code

Table 8-1 describes the status codes.

Table 8-1 Status code

| Stat<br>us<br>Cod<br>e | Code                          | Description                                                                                                                                                                                                                                          |
|------------------------|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 100                    | Continue                      | The server has received the initial part of the request and the client should continue to send the remaining part.<br><br>It is issued on a provisional basis while request processing continues. It alerts the client to wait for a final response. |
| 101                    | Switching Protocols           | The requester has asked the server to switch protocols and the server has agreed to do so. The target protocol must be more advanced than the source protocol.<br><br>For example, the current HTTP protocol is switched to a later version of HTTP. |
| 201                    | Created                       | The request has been fulfilled, resulting in the creation of a new resource.                                                                                                                                                                         |
| 202                    | Accepted                      | The request has been accepted for processing, but the processing has not been completed.                                                                                                                                                             |
| 203                    | Non-Authoritative Information | The server successfully processed the request, but is returning information that may be from another source.                                                                                                                                         |

| Status Code | Code              | Description                                                                                                                                                                                                 |
|-------------|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 204         | NoContent         | The server has successfully processed the request, but does not return any content.<br>The status code is returned in response to an HTTP OPTIONS request.                                                  |
| 205         | Reset Content     | The server successfully processed the request, but is not returning any content.                                                                                                                            |
| 206         | Partial Content   | The server has successfully processed a part of the GET request.                                                                                                                                            |
| 300         | Multiple Choices  | There are multiple options for the requested resource. For example, this code could be used to present a list of resource characteristics and addresses from which the client such as a browser may choose. |
| 301         | Moved Permanently | This and all future requests should be permanently directed to the given URI indicated in this response.                                                                                                    |
| 302         | Found             | The requested resource was temporarily moved.                                                                                                                                                               |
| 303         | See Other         | The response to the request can be found under a different URI,<br>and should be retrieved using a GET or POST method.                                                                                      |
| 304         | Not Modified      | The requested resource has not been modified. In such case, there is no need to retransmit the resource since the client still has a previously-downloaded copy.                                            |
| 305         | Use Proxy         | The requested resource is available only through a proxy.                                                                                                                                                   |
| 306         | Unused            | This HTTP status code is no longer used.                                                                                                                                                                    |
| 400         | BadRequest        | The request is invalid.<br>The client should modify the request instead of re-initiating it.                                                                                                                |
| 401         | Unauthorized      | The authorization information provided by the client is incorrect or invalid.                                                                                                                               |
| 402         | Payment Required  | This status code is reserved for future use.                                                                                                                                                                |

| Status Code | Code                          | Description                                                                                                                                                                                                                               |
|-------------|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 403         | Forbidden                     | The server has received the request and understood it, but the server is refusing to respond to it.<br>The client should modify the request instead of re-initiating it.                                                                  |
| 404         | NotFound                      | The requested resource cannot be found.<br>The client should modify the request instead of re-initiating it.                                                                                                                              |
| 405         | MethodNotAllowed              | A request method is not supported for the requested resource.<br>The client should modify the request instead of re-initiating it.                                                                                                        |
| 406         | Not Acceptable                | The server cannot fulfill the request based on the content characteristics of the request.                                                                                                                                                |
| 407         | Proxy Authentication Required | This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.                                                                                                                                 |
| 408         | Request Time-out              | The server timed out waiting for the request.<br>The client may re-initiate the request without modifications at any later time.                                                                                                          |
| 409         | Conflict                      | The request cannot be processed due to a conflict.<br>This status code indicates that the resource that the client attempts to create already exists, or the request fails to be processed because of the update of the conflict request. |
| 410         | Gone                          | The requested resource cannot be found.<br>The status code indicates that the requested resource has been deleted permanently.                                                                                                            |
| 411         | Length Required               | The server refused to process the request because the request does not specify the length of its content.                                                                                                                                 |
| 412         | Precondition Failed           | The server does not meet one of the preconditions that the requester puts on the request.                                                                                                                                                 |

| Status Code | Code                            | Description                                                                                                                                                                                                                                                                                                                                                      |
|-------------|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 413         | Request Entity Too Large        | The server refuses to process a request because the request entity is too large. The server may disable the connection to prevent the client from sending requests consecutively. If the server temporarily cannot process the request, the response will contain a Retry-After header field.                                                                    |
| 414         | Request-URI Too Large           | The URI provided was too long for the server to process.                                                                                                                                                                                                                                                                                                         |
| 415         | Unsupported Media Type          | The server does not support the media type in the request.                                                                                                                                                                                                                                                                                                       |
| 416         | Requested range not satisfiable | The requested range is invalid.                                                                                                                                                                                                                                                                                                                                  |
| 417         | Expectation Failed              | The server fails to meet the requirements of the Expect request-header field.                                                                                                                                                                                                                                                                                    |
| 422         | UnprocessableEntity             | The request is well-formed but is unable to be processed due to semantic errors.                                                                                                                                                                                                                                                                                 |
| 429         | TooManyRequests                 | The client sends excessive requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server receives excessive requests within a given time (beyond its processing capability). In this case, the client should repeat requests after the time specified in the Retry-After header of the response expires. |
| 500         | InternalServerError             | The server is able to receive the request but it could not understand the request.                                                                                                                                                                                                                                                                               |
| 501         | Not Implemented                 | The server does not support the requested function.                                                                                                                                                                                                                                                                                                              |
| 502         | Bad Gateway                     | The server acting as a gateway or proxy receives an invalid response from a remote server.                                                                                                                                                                                                                                                                       |
| 503         | ServiceUnavailable              | The requested service is invalid.<br>The client should modify the request instead of re-initiating it.                                                                                                                                                                                                                                                           |
| 504         | ServerTimeout                   | The server could not return a timely response. The response will reach the client only if the request carries a timeout parameter.                                                                                                                                                                                                                               |

| Status Code | Code                       | Description                                                                |
|-------------|----------------------------|----------------------------------------------------------------------------|
| 505         | HTTP Version not supported | The server does not support the HTTP protocol version used in the request. |

## 8.2 Error Codes

If an error occurs in API calling, no result is returned. Identify the cause based on the error code of each API. If an error occurs in API calling, HTTP status code 4xx or 5xx is returned. The response body contains the specific error code and information. If you fail to locate the cause of an error, contact customer service and provide the error code, so that we can help you solve the problem as soon as possible.

### Format of an Error Response Body

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{
 "errorMessage": "The format of message is error",
 "errorCode": "CCE.01400001"
}
```

In the preceding information, **errorCode** is an error code, and **errorMessage** describes the error.

### Error Code Description

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in [Error Codes](#).

| Status Code | Error Code   | Error Message                | Description                    | Troubleshooting                                                                                                    |
|-------------|--------------|------------------------------|--------------------------------|--------------------------------------------------------------------------------------------------------------------|
| 400         | CCE.01400001 | Invalid request.             | Invalid request body.          | Modify the request body based on the returned message and the CCE API documentation, or contact technical support. |
| 400         | CCE.01400002 | Subnet not found in the VPC. | No subnet is found in the VPC. | Check whether the subnet in the request body is in the corresponding VPC.                                          |



| Status Code | Error Code   | Error Message                           | Description                                               | Troubleshooting                                                           |
|-------------|--------------|-----------------------------------------|-----------------------------------------------------------|---------------------------------------------------------------------------|
| 400         | CCE.01400003 | IPv6 not supported for the subnet.      | The subnet does not support IPv6.                         | Use a subnet that supports IPv6.                                          |
| 400         | CCE.01400004 | No available flavors for master nodes.  | There is no available flavor for the master node.         | Change to another available cluster flavor or contact technical support.  |
| 400         | CCE.01400005 | Container network CIDR blocks conflict. | The container CIDR block conflicts with the existing one. | Check the container CIDR block based on the returned message.             |
| 400         | CCE.01400006 | Content type not supported.             | The content type is invalid.                              | Use the supported content type by referring to the CCE API documentation. |
| 400         | CCE.01400007 | Insufficient cluster quota.             | Insufficient cluster quota.                               | Submit a service ticket to increase the cluster quota.                    |
| 400         | CCE.01400008 | Insufficient server quota               | Insufficient ECS quota.                                   | Submit a service ticket to increase the ECS quota.                        |
| 400         | CCE.01400009 | Insufficient CPU quota.                 | Insufficient ECS CPU quota.                               | Submit a service ticket to increase the ECS CPU quota.                    |
| 400         | CCE.01400010 | Insufficient memory quota.              | Insufficient ECS memory quota.                            | Submit a service ticket to increase the ECS memory quota.                 |
| 400         | CCE.01400011 | Insufficient security group quota.      | Insufficient security group quota.                        | Submit a service ticket to increase the security group quota.             |
| 400         | CCE.01400012 | Insufficient EIP quota.                 | Insufficient EIP quota.                                   | Submit a service ticket to increase the EIP quota.                        |

| Status Code | Error Code   | Error Message                                           | Description                                                 | Troubleshooting                                                                   |
|-------------|--------------|---------------------------------------------------------|-------------------------------------------------------------|-----------------------------------------------------------------------------------|
| 400         | CCE.01400013 | Insufficient volume quota.                              | Insufficient disk quota.                                    | Submit a service ticket to increase the disk quota based on the returned message. |
| 400         | CCE.01400014 | Excessive nodes in the cluster.                         | The number of nodes exceeds the cluster scale.              | Submit a service ticket to change the cluster flavor.                             |
| 400         | CCE.01400015 | Version not supported.                                  | The cluster version is not supported.                       | Create a cluster of the supported version based on the returned message.          |
| 400         | CCE.01400016 | Current cluster type does not support this node flavor. | The current cluster type does not support this node flavor. | Use the correct node flavor based on the returned message.                        |
| 400         | CCE.01400017 | No available container CIDR block found.                | No available container CIDR block is found.                 | Use a correct container CIDR block based on the returned message.                 |
| 400         | CCE.01400018 | This type of OS cannot be created in this CCE version.  | The current CCE version does not support this type of OS.   | Use a supported OS based on the returned message.                                 |
| 400         | CCE.01400019 | Insufficient resource tenant quota.                     | The quota of the resource tenant is insufficient.           | Rectify the fault based on the returned message or contact technical support.     |
| 400         | CCE.01400020 | Insufficient VPC quota.                                 | Insufficient VPC quota.                                     | Rectify the fault based on the returned message or contact technical support.     |
| 400         | CCE.01400021 | No available flavors for nodes.                         | There is no available flavor for the node.                  | Change to another available node flavor or contact technical support.             |

| Status Code | Error Code   | Error Message                                             | Description                                                                              | Troubleshooting                                                           |
|-------------|--------------|-----------------------------------------------------------|------------------------------------------------------------------------------------------|---------------------------------------------------------------------------|
| 400         | CCE.01400022 | No available node volumes for nodes.                      | There is no available EVS disk flavor for the node.                                      | Change to another available EVS disk flavor or contact technical support. |
| 400         | CCE.01400023 | operation conflict                                        | Nodes cannot be created during cluster scale-out.                                        | Try again later.                                                          |
| 400         | CCE.01400024 | operation conflict                                        | The cluster cannot be deleted during node creation.                                      | Try again later.                                                          |
| 400         | CCE.01400025 | Unsupported flavor with insufficient sub-ENI quota        | The sub-ENI quota is insufficient. The VM flavor is not supported by CCE Turbo clusters. | Select a VM flavor whose sub-ENI quota is not 0.                          |
| 400         | CCE.01400033 | Snapshot task already exists.                             | The cluster backup task already exists.                                                  | Wait until the cluster backup task is complete and try again.             |
| 400         | CCE.01400034 | Cluster enables deletion protection,for bidden to delete. | The cluster deletion protection is enabled. The cluster cannot be deleted.               | Disable the cluster deletion protection and delete the cluster.           |
| 400         | CCE.01400074 | The key upgrade task is failed and cannot be canceled.    | Key steps in the cluster upgrade fail, and the upgrade task cannot be canceled.          | Retry the upgrade or contact technical support.                           |
| 400         | CCE.01400075 | Task is expired.                                          | The pre-upgrade check has expired.                                                       | Perform the pre-upgrade check again.                                      |

| Status Code | Error Code   | Error Message         | Description                | Troubleshooting                                                                                                    |
|-------------|--------------|-----------------------|----------------------------|--------------------------------------------------------------------------------------------------------------------|
| 400         | CCE.02400001 | Invalid request.      | Invalid request body.      | Modify the request body based on the returned message and the CCE API documentation, or contact technical support. |
| 400         | CCE.03400001 | Invalid request.      | Invalid request body.      | Modify the request body based on the returned message and the CCE API documentation, or contact technical support. |
| 400         | CCE.03400002 | Missing access key.   | The access key is missing. | Ensure that the version of the installed or upgraded storage add-on is correct, or contact technical support.      |
| 401         | CCE.01401001 | Authorization failed. | Authentication failed.     | Rectify the fault based on the returned message or contact technical support.                                      |
| 401         | CCE.02401001 | Authorization failed. | Authentication failed.     | Rectify the fault based on the returned message or contact technical support.                                      |
| 401         | CCE.03401001 | Authorization failed. | Authentication failed.     | Rectify the fault based on the returned message or contact technical support.                                      |
| 403         | CCE.01403001 | Forbidden.            | Access denied.             | Rectify the fault based on the returned message or contact technical support.                                      |

| Status Code | Error Code   | Error Message                                                         | Description                                                                              | Troubleshooting                                                               |
|-------------|--------------|-----------------------------------------------------------------------|------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| 403         | CCE.01403002 | Current cluster status does not allow node pool to be deleted.        | A node pool in the cluster that is in the current state cannot be deleted.               | Wait until the cluster runs properly and try again.                           |
| 403         | CCE.01403003 | Current node pool status does not allow node pool to be deleted.      | The node pool in the current state cannot be deleted.                                    | Wait until the node pool runs properly and try again.                         |
| 403         | CCE.01403004 | Node pool cannot be deleted when it is locked by an order.            | The node pool locked by an order cannot be deleted.                                      | Pay or cancel the yearly/monthly order and try again.                         |
| 403         | CCE.01403005 | Node pool cannot be deleted when it is scaling.                       | The node pool is being scaled out and cannot be deleted.                                 | Wait until the node pool scale-out completes and try again.                   |
| 403         | CCE.01403006 | Node pool cannot be deleted when exists installing or deleting nodes. | There are nodes that are being installed or deleted and the node pool cannot be deleted. | Wait until the nodes in the node pool are installed or deleted and try again. |
| 403         | CCE.02403001 | Forbidden.                                                            | Access denied.                                                                           | Rectify the fault based on the returned message or contact technical support. |
| 403         | CCE.03403001 | Forbidden.                                                            | Access denied.                                                                           | Rectify the fault based on the returned message or contact technical support. |
| 404         | CCE.01404001 | Resource not found.                                                   | Resources not found.                                                                     | Check whether the resource to be accessed has been deleted.                   |

| Status Code | Error Code   | Error Message                                    | Description                                                   | Troubleshooting                                                                         |
|-------------|--------------|--------------------------------------------------|---------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| 404         | CCE.02404001 | Resource not found.                              | Resources not found.                                          | Check whether the resource to be accessed has been deleted.                             |
| 404         | CCE.03404001 | Resource not found.                              | Resources not found.                                          | Check whether the resource to be accessed has been deleted.                             |
| 409         | CCE.01409001 | The resource already exists.                     | The resource already exists.                                  | Delete the resource and try again.                                                      |
| 409         | CCE.01409002 | Resource updated with out-of-date version.       | An out-of-date version is used to update the target resource. | Ensure that the target resource version is the latest one or contact technical support. |
| 409         | CCE.02409001 | The resource already exists.                     | The resource already exists.                                  | Delete the resource and try again.                                                      |
| 409         | CCE.03409001 | Addon instance has installed.                    | The add-on instance has been installed.                       | Delete the add-on instance and install it again.                                        |
| 429         | CCE.01429002 | Resource locked by other requests.               | The resource is locked by another request.                    | Rectify the fault based on the returned message or contact technical support.           |
| 429         | CCE.01429003 | The concurrency limit of tasks has been reached. | The number of concurrent tasks has reached the upper limit.   | Reduce the frequency of sending requests or contact technical support.                  |
| 429         | CCE.02429001 | The throttling threshold has been reached.       | The maximum number of requests has been reached.              | Reduce the frequency of sending requests or contact technical support.                  |

| Status Code | Error Code   | Error Message   | Description     | Troubleshooting                                                               |
|-------------|--------------|-----------------|-----------------|-------------------------------------------------------------------------------|
| 500         | CCE.01500001 | Internal error. | Internal error. | Rectify the fault based on the returned message or contact technical support. |
| 500         | CCE.02500001 | Internal error. | Internal error. | Rectify the fault based on the returned message or contact technical support. |
| 500         | CCE.03500001 | Internal error. | Internal error. | Rectify the fault based on the returned message or contact technical support. |

## 8.3 Obtaining a Project ID

### Scenarios

A project ID is required for some URLs when APIs are called. Therefore, obtain a project ID in advance. Two methods are available:

- [Call an API](#)
- [Use the console](#)

### Obtaining the Project ID by Calling an API

You can obtain the project ID by calling the API used to [obtain projects based on specified criteria](#).

The API for obtaining a project ID is **GET https://{Endpoint}/v3/projects**. **{Endpoint}** indicates the endpoint of IAM, which can be obtained from [Endpoints](#). For details about API authentication, see [Authentication](#).

The following is an example response. The value of **id** is the project ID.

```
{
 "projects": [
 {
 "domain_id": "65382450e8f64ac0870cd180d14e684b",
 "is_domain": false,
 "parent_id": "65382450e8f64ac0870cd180d14e684b",
 "name": "project_name",
 "description": "",
 "links": {
 "next": null,
 "previous": null,
 "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
 }
 }
]
}
```

```
{
 "id": "a4a5d4098fb4474fa22cd05f897d6b99",
 "enabled": true
},
"links": {
 "next": null,
 "previous": null,
 "self": "https://www.example.com/v3/projects"
}
```

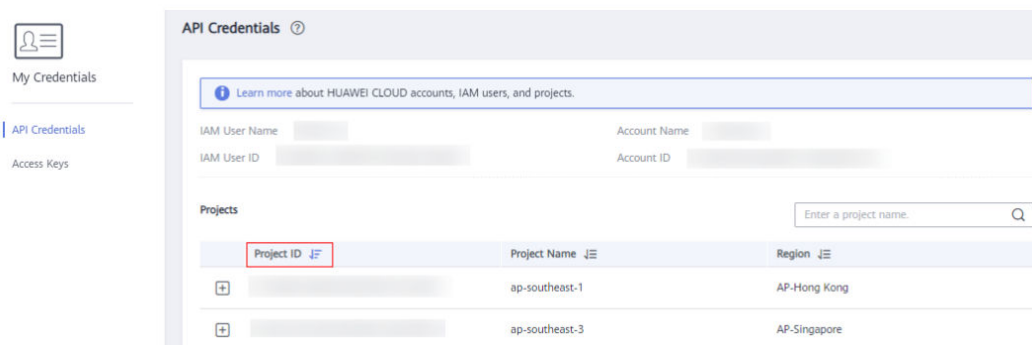
## Obtaining a Project ID from the Console

To obtain a project ID from the console, perform the following operations:

1. Log in to the management console.
2. Hover over the username in the upper right corner and select **My Credentials** from the drop-down list.

On the **API Credentials** page, view the project ID in the project list.

**Figure 8-1** Viewing project IDs



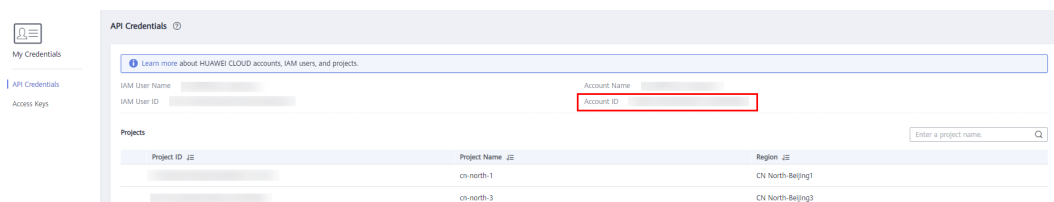
## 8.4 Obtaining an Account ID

An account ID (domain-id) is required for some URLs when an API is called. To obtain an account ID, perform the following operations:

1. Sign up and log in to the management console.
2. Hover over the username and choose **My Credentials** from the drop-down list.

On the **API Credentials** page, view the account ID.

**Figure 8-2** Obtaining an account ID





## 8.5 Specifying Add-ons to Be Installed During Cluster Creation

When creating a cluster, you can add a key-value pair to **annotations** of the **metadata** field in the request body to install an add-on in the cluster. The key is **cluster.install.addons/install**, and the value is a JSON array of **AddonTemplate**.

**Table 8-2** Value data structure

| Parameter | Mandatory | Type                               | Description                                                                                                                                                                                                                                         |
|-----------|-----------|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Value     | Yes       | JSON array of AddonTemplate string | Add-on to be installed in the cluster. If this parameter is not specified, the CoreDNS and CCE Container Storage (Everest) add-ons are installed in the cluster by default. For details about the parameter values, see <a href="#">Table 8-3</a> . |

**Table 8-3** Data structure of the AddonTemplate field

| Parameter         | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-------------------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| addonTemplateName | Yes       | String | Add-on name.<br>Options: <ul style="list-style-type: none"><li>• <b>coredns</b>: Install CoreDNS.</li><li>• <b>everest</b>: Install CCE Container Storage (Everest).</li><li>• <b>node-local-dns</b>: Install NodeLocal DNSCache.</li><li>• <b>volcano</b>: Install Volcano Scheduler.</li><li>• <b>npd</b>: Install CCE Node Problem Detector.</li><li>• <b>cie-collector</b>: Install Cloud Native Cluster Monitoring.</li><li>• <b>log-agent</b>: Install Cloud Native Logging.</li><li>• <b>virtual-kubelet</b>: Install CCE Cloud Bursting Engine for CCI.</li></ul> |

| Parameter | Mandatory | Type     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----------|-----------|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| version   | No        | String   | Add-on version.<br><br>To view the add-on version, log in to the CCE console, choose <b>Add-ons</b> in the navigation pane, click the name of the target add-on, and check the change history. If this parameter is left blank, the latest version is used by default.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| values    | No        | Json Map | <ul style="list-style-type: none"><li>CoreDNS: For details about the parameters required for installing the add-on, see <a href="#">CoreDNS</a>.</li><li>Cloud Native Cluster Monitoring: For details about the parameters required for installing the add-on, see <a href="#">Cloud Native Cluster Monitoring</a>.</li><li>Cloud Native Logging: For details about the parameters required for installing the add-on, see <a href="#">Cloud Native Log Collection</a>.</li><li>CCE Cloud Bursting Engine for CCI: For details about the parameters required for installing the add-on, see <a href="#">CCE Cloud Bursting Engine for CCI</a>.</li></ul> <b>NOTE</b><br>You do not need to specify this parameter when installing the CCE Container Storage (Everest), NodeLocal DNSCache, Volcano Scheduler, or CCE Node Problem Detector add-on. |

## Example Request

Create a cluster using a VPC network and install CoreDNS and CCE Container Storage (Everest) in it.

```
{
 "kind": "Cluster",
 "apiVersion": "v3",
 "metadata": {
 "name": "test",
 "annotations": {
 "cluster.install.addons.external/install": "[{\"addonTemplateName\":\"icagent\",\"extendParam\":{\"logSwitch\":{\"false\":\"false\",\"tDSEnable\":{\"false\"}}}],\"cluster.install.addons/install\": \"[{\"addonTemplateName\":\"coredns\",\"values\":{\"flavor\":{\"is_default\":false,\"name\":\"2500\",\"recommend_cluster_flavor_types\":[\"small\"],\"replicas\":\"2\",\"resources\":{\"limitsCpu\":\"500m\",\"limitsMem\":\"512Mi\",\"name\":\"coredns\",\"replicas\":\"2\",\"requestsCpu\":\"500m\",\"requestsMem\":\"512Mi\"}},\"size\":\"small\",\"category\":[\"CCE\",\"Turbo\"]}],\"addonTemplateName\":\"everest\"]\"
 }
 }
}
```

```
},
"spec": {
 "category": "CCE",
 "flavor": "cce.s1.small",
 "version": "v1.29",
 "type": "VirtualMachine",
 "hostNetwork": {
 "vpc": "*****",
 "subnet": "*****"
 },
 "containerNetwork": {
 "mode": "vpc-router",
 "cidrs": [
 {
 "cidr": "10.0.0.0/16"
 }
]
 },
 "ipv6enable": false,
 "description": "",
 "billingMode": 0,
 "kubeProxyMode": "iptables",
 "extendParam": {
 "alpha.cce/fixPoolMask": "25",
 "enterpriseProjectId": "0"
 },
 "authentication": {
 "mode": "rbac"
 },
 "configurationsOverride": [
 {
 "name": "kube-apiserver",
 "configurations": [
 {
 "name": "support-overload",
 "value": true
 }
]
 }
],
 "deletionProtection": false,
 "serviceNetwork": {
 "IPv4CIDR": "10.247.0.0/16"
 }
}
```

## 8.6 How to Obtain Parameters in the API URI

### Obtaining a Project ID (project\_id)

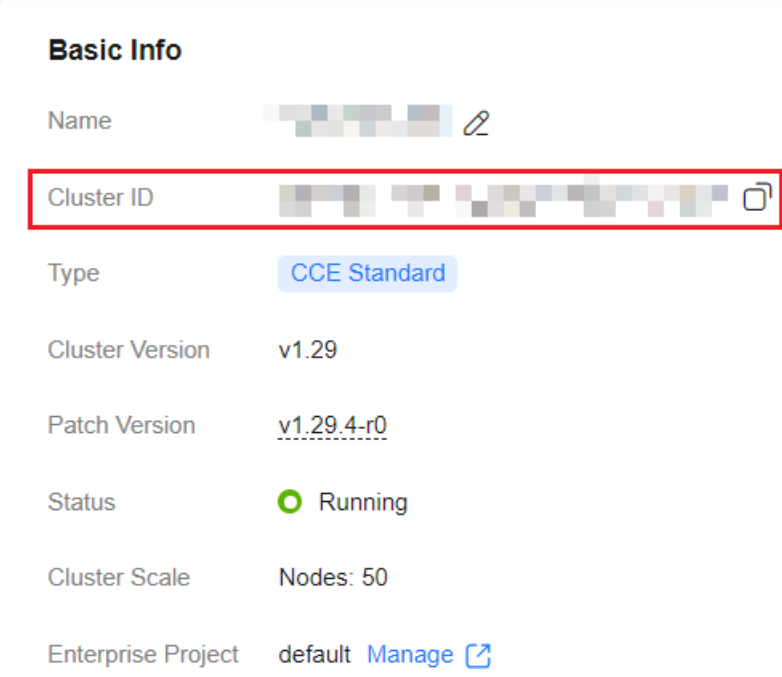
**project\_id** indicates the project ID, which can be obtained from the console or APIs. For details, see [Obtaining a Project ID](#).

### Obtaining a Cluster ID (cluster\_id)

**Step 1** Log in to the CCE console. In the navigation pane, choose **Clusters**.

**Step 2** Click the name of the created cluster. The cluster details page is displayed. Obtain the cluster ID.

Figure 8-3 Obtaining the cluster ID



----End

Obtaining a Node ID (node\_id)

- Step 1 Log in to the CCE console. In the navigation pane, choose **Clusters**.
- Step 2 Click the name of the created cluster to access the cluster console. In the navigation pane, choose **Nodes**. On the **Nodes** tab, move the cursor to the node name and view the node ID.

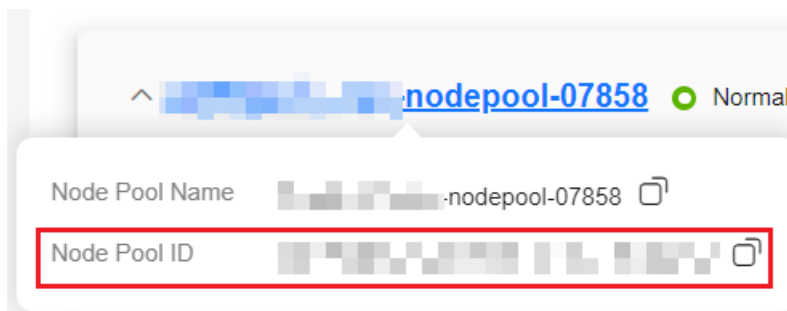
Figure 8-4 Obtaining the node ID



----End

Obtaining a Node Pool ID (nodepool\_id)

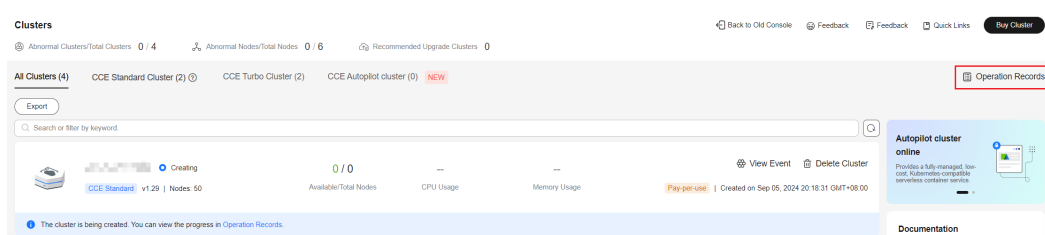
- Step 1 Log in to the CCE console. In the navigation pane, choose **Clusters**.
- Step 2 Click the name of the created cluster to access the cluster console. In the navigation pane, choose **Nodes**. On the **Node Pools** tab, move the cursor to the node pool name and view the node pool ID.

**Figure 8-5** Obtaining the node pool ID

----End

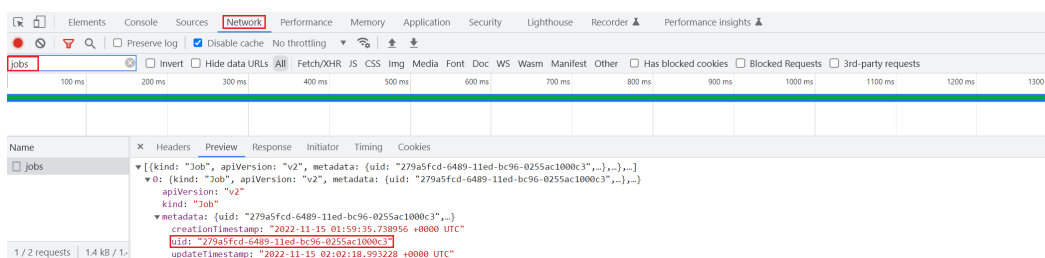
## Obtaining a Job ID (job\_id)

**Step 1** Log in to the CCE console. In the navigation pane, choose **Clusters**. The following uses cluster management as an example to describe how to obtain the **job\_id** of the cluster that is being created.

**Figure 8-6** Creating a cluster

**Step 2** Obtain the job ID.

1. If you are using Google Chrome, press **F12**. On the pane displayed on the right, click the **Network** tab.
2. Click **Operation Records** on the CCE console to view details about cluster operation records.
3. Enter **jobs** in the **Filter** text box to filter out the jobs. Select a job from the list on the left and click **Preview**. The **uid** field indicates the job UID.

**Figure 8-7** Obtaining the job ID

----End

## 8.7 Creating a VPC and Subnet

### Context

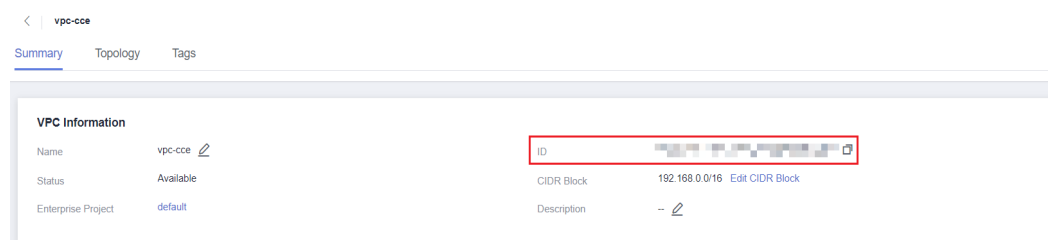
To provide a secure and isolated network environment for CCE, create a VPC before creating a cluster.

If you have already created a VPC, you do not need to create it again.

### Creating a VPC

- Step 1** On the management console, click **Service List**, and choose **Network > Virtual Private Cloud** to launch the VPC console.
- Step 2** On the VPC console, click **Create VPC** to create a VPC.
- Step 3** The created VPC is displayed in the list. Click its name and obtain the VPC ID, which will be required in [cluster creation](#).

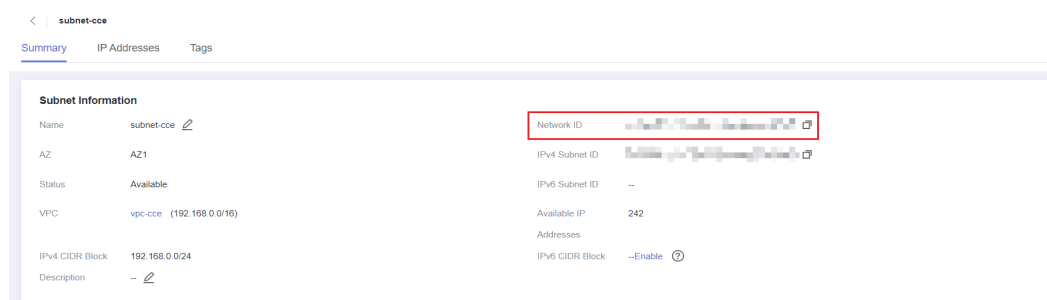
**Figure 8-8** Obtaining the VPC ID



----End

### Creating a Subnet

- Step 1** On the management console, click **Service List**, and choose **Network > Virtual Private Cloud** to launch the VPC console.
- Step 2** On the **VPC console**, choose **Virtual Private Cloud > Subnets** in the navigation pane, and click **Create Subnet** in the upper right corner.
- Step 3** Create a subnet as prompted and click its name to obtain the network ID, which will be required in [cluster creation](#).

**Figure 8-9** Obtaining the network ID of a subnet

----End

## 8.8 Creating a Key Pair

### Context

Create a key pair before you create a container cluster. Key pairs are used for user identity authentication upon login to a worker node.

If you have already created a key pair, you do not need to create it again.

### Procedure

- Step 1** On the management console, click **Service List**, and choose **Compute > Elastic Cloud Server** to go to the ECS console.
- Step 2** In the navigation pane, choose **Key Pair**.
- Step 3** Click **Create Key Pair** and create a key pair as prompted. For details, see [Key Pairs](#).
- Step 4** After the creation is complete, a key file in the **.pem** format is generated and automatically saved to the default directory on your local computer.

----End

## 8.9 Node Flavor Description

### NOTE

Different regions support different node flavors, and node flavors may be changed or sold out. You are advised to log in to the CCE console and check whether the required node flavors are supported on the page for creating nodes.

- **CCE standard cluster**

You are not advised to use a flavor below **2 vCPUs and 4 GiB of memory** for a node in a CCE cluster because system components occupy certain resources. If the flavor is too small, pods with large specifications may fail to be scheduled to that node, and the remaining resources cannot be effectively used. As a result, the overall resource utilization is low. You are advised to obtain node flavors on the console. For details about node flavors, see [x86 ECS Specifications](#).

You need to enter the specific flavor name, for example, **c6.large.2**.

The following tables list the specifications supported by CCE. (The specifications of some ECS models vary by region. The actual specifications displayed on the console are used.)

- ECS (VM)

| Node Type          | Flavor Type                    | Flavor Name                                                       |
|--------------------|--------------------------------|-------------------------------------------------------------------|
| x86 node           | General computing-plus         | c9, ac8, ac7, c7, c7e, c7n, c7t, c6s, c6, c6nl, c6x, c3, and c3ne |
|                    | General computing              | s7, s7n, as7, s6, s6nl, s3, and s2                                |
|                    | General computing-basic        | t6                                                                |
|                    | Memory-optimized               | m7, m7n, m6, m6nl, m3, m2, am7, and m3ne                          |
|                    | Large-memory                   | e7, e6, and e3                                                    |
|                    | Disk-intensive                 | d7, d6, d6nl, d2, and d3                                          |
|                    | Ultra-high I/O                 | i7, i7n, i3, ir7, ir7n, and ir3                                   |
|                    | High-performance computing     | h3 and hc2                                                        |
|                    | GPU-accelerated                | pi2 and pi1<br>p2s, p2v, p2vs, and p1<br>g6, g5, and g6v          |
|                    | AI-accelerated                 | ai1s and ai1                                                      |
|                    | FlexusX                        | x1 and x1e                                                        |
| Kunpeng (Arm) node | Kunpeng general computing-plus | kc1, kc1n, and kc2                                                |
|                    | Kunpeng memory-optimized       | km1, km1s, and km2                                                |
|                    | Kunpeng ultra-high I/O         | ki1                                                               |
|                    | AI-accelerated                 | kai1s and kai1                                                    |

- ECS (physical server): c6 and c7 are supported.



| Node Type              | Flavor Name                                                                                                        |
|------------------------|--------------------------------------------------------------------------------------------------------------------|
| General computing-plus | c6.22xlarge.2.physical<br>c6.22xlarge.2.physical.129nic<br>c6.22xlarge.4.physical<br>c6.22xlarge.4.physical.129nic |

- BMS:

x86 nodes: BMSs of the physical.d2, physical.s4, physical.c6ne, and physical.d6ne types are supported.

Arm nodes: BMS of the physical.a1.2xlarge type is supported.

- **CCE Turbo cluster**

- ECS (VM)

| Node Type          | Flavor Type                    | Flavor Name                                           |
|--------------------|--------------------------------|-------------------------------------------------------|
| x86 node           | General computing-plus         | c9, ac8, ac7, c7, c7n, c7e, c7t, c6ne, c6sne, and c3n |
|                    | General computing              | s7n, s7, as7, and s6ne                                |
|                    | Memory-optimized               | m7, m7n, m3n, m6ne, m6sne, and am7                    |
|                    | Large-memory                   | e7                                                    |
|                    | Disk-intensive                 | d7                                                    |
|                    | Ultra-high I/O                 | i7, i7n, i3ne, ir7, and ir7n                          |
|                    | GPU-accelerated                | pi2ne                                                 |
|                    | FlexusX                        | x1 and x1e                                            |
| Kunpeng (Arm) node | Kunpeng general computing-plus | kc1n                                                  |
|                    | Kunpeng memory-optimized       | km1n                                                  |

- ECS (physical server): c6 and c7 are supported.

| Node Type              | Flavor Name                                                                                                        |
|------------------------|--------------------------------------------------------------------------------------------------------------------|
| General computing-plus | c6.22xlarge.2.physical<br>c6.22xlarge.2.physical.129nic<br>c6.22xlarge.4.physical<br>c6.22xlarge.4.physical.129nic |

 NOTE

The available specifications of IPv6 dual-stack nodes vary with regions. For details, see [Constraints](#).

## 8.10 Adding a Salt in the password Field When Creating a Node

When a node is created through the API, you need to add a salt to the **password** field to safeguard the password. The procedure is as follows:

 NOTE

The salt must be set based on the password complexity requirements:

- A string of 8–26 characters.
- Contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters !@\$%^&\*\_+=+[]{};.,/?
- Cannot contain the username or the username spelled backwards.
- Cannot contain the username, the username spelled backwards, or more than two consecutive characters in the username (for Windows ECSs).

### Python

To salt a password in the Python 3.7.7 environment, perform the following steps:

```
pip install passlib
python -c "import base64; from passlib.hash import sha512_crypt; salted_password = base64.b64encode(sha512_crypt.hash('*****', salt='salt', rounds=5000).encode()).decode(); print(salted_password)"
```

 NOTE

The **python crypt** package has compatibility issues in macOS. If the package cannot be executed, run it in Linux.

### Java

To salt a password in the Java environment, perform the following steps:

1. Obtain a random number as the salt.

```
private static String getCharAndNumr(int length) {
 String val = "";
 Random random = new SecureRandom();
 for (int i = 0; i < length; i++) {
 // Indicates whether to output letters or digits.
 String charOrNum = random.nextInt(2) % 2 == 0 ? "char" : "num";
```

```
// Character string
if ("char".equalsIgnoreCase(charOrNum)) {
 // Indicates whether an upper-case or lower-case letter is obtained.
 int choice = random.nextInt(2) % 2 == 0 ? 65 : 97;
 val += (char) (choice + random.nextInt(26));
} else if ("num".equalsIgnoreCase(charOrNum)) { // Digit
 val += String.valueOf(random.nextInt(10));
}
}
return val;
}
```

2. Generate a salt.

```
private static String generateSalt() {
 String salt;
 try {
 salt = "6" + getCharAndNumr(16);
 } catch (Exception e) {
 salt = defaultSalt;
 }

 return salt;
}
```

3. Generate a ciphertext password based on the salt.

```
public static String getSaltPassword(String password) {
 if (StringUtils.isBlank(password)) {
 throw new BizException("password is empty");
 }

 String salt = generateSalt();

 Crypt crypt = new Crypt();
 return crypt.crypt(password, salt);
}
```

4. Encode the value of the **password** field using Base64.

```
(Base64.getEncoder().encodeToString(AddSaltPasswordUtil.getSaltPassword(cceNodeCreateVo.getPassword()).getBytes()))
```

5. A complete example is as follows:

```
import java.util.Base64;
import java.util.Random;
import java.security.SecureRandom;

import org.apache.commons.codec.digest.Crypt;
import org.apache.commons.lang.StringUtils;

public class PassWord {

 static String defaultSalt = null;

 public static void main(String[] args) throws Exception {
 System.out.println(Base64.getEncoder().encodeToString(PassWord.getSaltPassword("Custom password").getBytes()));
 }

 // Generate a ciphertext password based on the salt.
 public static String getSaltPassword(String password) throws Exception {
 if (StringUtils.isBlank(password)) {
 throw new Exception("password is empty");
 }
 String salt = generateSalt();
 return Crypt.crypt(password, salt);
 }

 //Generate a salt.
 private static String generateSalt() {
 String salt;
 try {
```

```
 salt = "6" + getCharAndNumr(16);
 } catch (Exception e) {
 salt = defaultSalt;
 }
 return salt;
}

// Obtain a random number as the salt.
private static String getCharAndNumr(int length) {
 String val = "";
 Random random = new SecureRandom();
 for (int i = 0; i < length; i++) {
 // Indicates whether to output letters or digits.
 String charOrNum = random.nextInt(2) % 2 == 0 ? "char" : "num";
 // Character string
 if ("char".equalsIgnoreCase(charOrNum)) {
 // Indicates whether an upper-case or lower-case letter is obtained.
 int choice = random.nextInt(2) % 2 == 0 ? 65 : 97;
 val += (char) (choice + random.nextInt(26));
 } else if ("num".equalsIgnoreCase(charOrNum)) { // Digit
 val += String.valueOf(random.nextInt(10));
 }
 }
 return val;
}
```

## Go

You can use either of the following methods to salt passwords for the Go language:

- <https://github.com/amoghe/go-crypt>
- <https://github.com/GehirnInc/crypt>

## 8.11 Maximum Number of Pods That Can Be Created on a Node

### Calculation of the Maximum Number of Pods on a Node

The maximum number of pods that can be created on a node is calculated based on the cluster type:

| Network Model  | Maximum Number of Pods on a Node                 | Recommended Configuration |
|----------------|--------------------------------------------------|---------------------------|
| Tunnel network | <a href="#">Maximum number of pods on a node</a> | None                      |

| Network Model                                     | Maximum Number of Pods on a Node                                                                                                                                     | Recommended Configuration                                                                                                                                                                                                                                  |
|---------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| VPC network                                       | The smaller value between the <a href="#">Maximum Number of Pods on a Node</a> and <a href="#">Number of Allocatable Pod IP Addresses on a Node</a>                  | To ensure that new pods run smoothly on a node, verify that the maximum number of pods on the node does not exceed the number of allocatable pod IP addresses. If the node lacks sufficient pod IP addresses, new pods will not function properly.         |
| Cloud Native Network 2.0 (for CCE Turbo clusters) | The smaller value between the <a href="#">maximum number of pods on a node</a> and the <a href="#">number of network interfaces on a node in a CCE Turbo cluster</a> | To ensure new pods run smoothly on a node, make sure that the number of pods on the node does not exceed the number of network interfaces on it. If there are not enough network interfaces available on the node, the new pods may not function properly. |

## Number of Allocatable Pod IP Addresses on a Node

The number of allocatable pod IP addresses on a node is the count of IP addresses available for allocation to pods on that node. When creating a cluster in the VPC network model, follow the [rules for managing pod IP addresses](#) and specify the number of pod IP addresses that each node can allocate using `alpha.cce/fixPoolMask`.

The maximum number of pods that can be created on a node is determined by the number of pod IP addresses available for allocation. In a [containerized environment](#), each pod requires its own unique IP address. If the node runs out of reserved pod IP addresses, new pods cannot be created. If `hostNetwork: true` is configured in the YAML file, pods will use the [host network](#) instead of the reserved pod IP addresses. For details, see [Pod IP Address Allocation Differences Between the Container Network and Host Network](#).

**Figure 8-10** Specifying the number of allocatable pod IP addresses on a node in the VPC network model

The screenshot shows the 'Container Network' configuration page. Under 'Network Model', the 'VPC network' is selected, described as suitable for high performance and integrated with the underlying network. The 'Tunnel network' is also visible, described as suitable for large-scale networking. Below this, 'DataPlane V2' is enabled with a toggle switch. The 'Container CIDR Block' is set to 'Manually set' with a value of 10.0.0.0/16. A warning message states: 'The value cannot be changed after creation. For cross-VPC passthrough networking, make sure the container CIDR block does not overlap with the VPC CIDR block to be accessed.' At the bottom, 'Reserved Pod IP Per Node' is set to 128, with a note that the setting cannot be modified after cluster creation and that the current configuration supports a maximum of 512 worker nodes.

By default, each node in a cluster is assigned a CIDR block from which pod IP addresses are allocated. **The usable number of IP addresses for pods within this block is typically the total number of addresses in the CIDR block minus three reserved addresses** (including the network address, gateway address, and broadcast address). For example, in the preceding figure, the usable number of IP addresses for pods is 125 ( $128 - 3$ ).

## Maximum Number of Pods on a Node

When creating a node, you can configure the maximum number of pods (maxPods) that can be created on the node. This parameter is a configuration item of kubelet and determines the maximum number of pods that can be created by kubelet.

### NOTICE

For nodes in the default node pool (**DefaultPool**), the maximum number of pods cannot be changed after the nodes are created.

After a node in a custom node pool is created, you can modify the **max-pods** parameter in the node pool configuration to change the maximum number of pods on the node. For details, see [Configuring a Node Pool](#).

By default, the maximum number of pods on a node can be adjusted to 256. To increase the deployment density on a node, submit a [service ticket](#) to increase the maximum number of pods on a node, which can be 512.

**Figure 8-11** Configuring the maximum number of pods during node creation

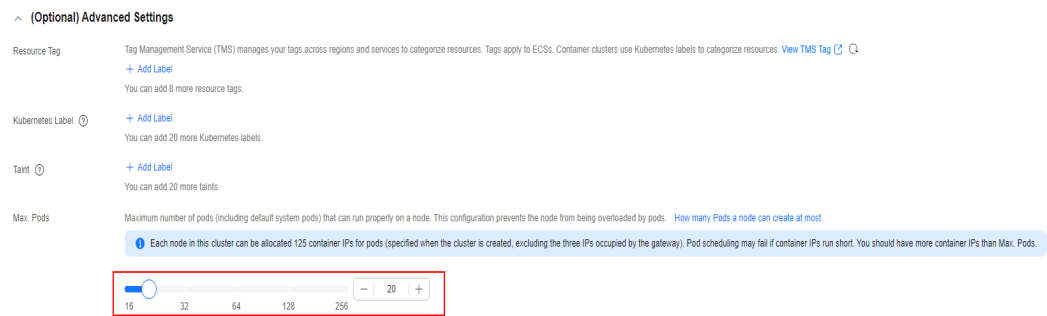


**Table 8-4** lists the default maximum number of pods on a node based on node specifications.

**Table 8-4** Default maximum number of pods on a node

| Memory         | Max. Pods |
|----------------|-----------|
| 4 GB           | 20        |
| 8 GB           | 40        |
| 16 GB          | 60        |
| 32 GB          | 80        |
| 64 GB or above | 110       |

## Number of Node Network Interfaces (Available Only in CCE Turbo Clusters)

In a CCE Turbo cluster, ECS nodes use supplementary network interfaces and BMS nodes use network interfaces. The maximum number of pods that can be created on a node depends on the number of network interfaces that can be used by the node.

**Figure 8-12** Node network interfaces

## Pod IP Address Allocation Differences Between the Container Network and Host Network

When creating a pod, you can select the container network or host network for the pod.

- Container network (default): Each pod is assigned an IP address by the cluster networking add-ons, which occupies the IP addresses of the container network.

- Host network: Pods with **hostNetwork: true** configured directly use the network of the host. For details, see [Configuring hostNetwork for Pods](#). After the configuration, the pods use the ports on the host and their IP address is identical to that of the host, **without relying on the IP addresses of the container network**. When using a host network, avoid conflicts between the pod ports and service ports on the host. Use a host network only if a particular application requires access to a specific port on the host.

## 8.12 Node OS

### Mappings Between Cluster Versions and OS Versions

The following table lists the mappings between released cluster versions and OS versions.

Table 8-5 ECS (VM) OSs

| OS                       | Cluster Version | CCE Standard Cluster |                                   | CCE Turbo Cluster        | Latest Kernel                           |
|--------------------------|-----------------|----------------------|-----------------------------------|--------------------------|-----------------------------------------|
|                          |                 | VPC Network          | Tunnel Network                    | Cloud Native Network 2.0 |                                         |
| Huawei Cloud EulerOS 2.0 | v1.32           | √                    | √                                 | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.x86_64 |
|                          | v1.31           | √                    | √                                 | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.x86_64 |
|                          | v1.30           | √                    | √                                 | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.x86_64 |
|                          | v1.29           | √                    | √                                 | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.x86_64 |
|                          | v1.28           | √                    | √                                 | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.x86_64 |
|                          | v1.27           | √                    | Supported in v1.27.3-r0 or later. | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.x86_64 |



| OS                             | Cluster Version            | CCE Standard Cluster |                                    | CCE Turbo Cluster        | Latest Kernel                                                                                                                                                                                                              |
|--------------------------------|----------------------------|----------------------|------------------------------------|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                |                            | VPC Network          | Tunnel Network                     | Cloud Native Network 2.0 |                                                                                                                                                                                                                            |
|                                | v1.25 (end of maintenance) | √                    | Supported in v1.25.6-r0 or later.  | √                        | <ul style="list-style-type: none"><li>Clusters of v1.25.3-r0 or later<br/>5.10.0-182.0.0.95.r2673_211.hce2.x86_64</li><li>Clusters of versions earlier than v1.25.3-r0<br/>5.10.0-60.18.0.50.r865_35.hce2.x86_64</li></ul> |
|                                | v1.23 (end of maintenance) | √                    | Supported in v1.23.11-r0 or later. | √                        | <ul style="list-style-type: none"><li>Clusters of v1.23.8-r0 or later<br/>5.10.0-182.0.0.95.r2673_211.hce2.x86_64</li><li>Clusters of versions earlier than v1.23.8-r0<br/>5.10.0-60.18.0.50.r865_35.hce2.x86_64</li></ul> |
| Huawei Cloud EulerOS 2.0 (Arm) | v1.32                      | √                    | √                                  | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.aarch64                                                                                                                                                                                   |
|                                | v1.31                      | √                    | √                                  | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.aarch64                                                                                                                                                                                   |
|                                | v1.30                      | √                    | √                                  | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.aarch64                                                                                                                                                                                   |
|                                | v1.29                      | √                    | √                                  | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.aarch64                                                                                                                                                                                   |
|                                | v1.28                      | √                    | √                                  | √                        | 5.10.0-182.0.0.95.r2673_211.hce2.aarch64                                                                                                                                                                                   |

| OS           | Cluster Version            | CCE Standard Cluster              |                                    | CCE Turbo Cluster                 | Latest Kernel                                                                                                                                                                                             |
|--------------|----------------------------|-----------------------------------|------------------------------------|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|              |                            | VPC Network                       | Tunnel Network                     | Cloud Native Network 2.0          |                                                                                                                                                                                                           |
|              | v1.27                      | √                                 | Supported in v1.27.3-r0 or later.  | √                                 | 5.10.0-182.0.0.95.r2673_211.hce2.aarch64                                                                                                                                                                  |
|              | v1.25 (end of maintenance) | Supported in v1.25.3-r0 or later. | Supported in v1.25.6-r0 or later.  | Supported in v1.25.3-r0 or later. | 5.10.0-182.0.0.95.r2673_211.hce2.aarch64                                                                                                                                                                  |
|              | v1.23 (end of maintenance) | Supported in v1.23.6-r0 or later. | Supported in v1.23.11-r0 or later. | Supported in v1.23.6-r0 or later. | <ul style="list-style-type: none"><li>Clusters of v1.23.8-r0 or later<br/>5.10.0-182.0.0.95.r2673_211.hce2.aarch64</li><li>v1.23.6-r0 to v1.23.7-r0:<br/>5.10.0-60.18.0.50.r865_35.hce2.aarch64</li></ul> |
| Ubuntu 22.04 | v1.32                      | √                                 | ×                                  | √                                 | 5.15.0-138-generic                                                                                                                                                                                        |
|              | v1.31                      | √                                 | ×                                  | √                                 | 5.15.0-138-generic                                                                                                                                                                                        |
|              | v1.30                      | √                                 | ×                                  | √                                 | 5.15.0-138-generic                                                                                                                                                                                        |
|              | v1.29                      | √                                 | ×                                  | √                                 | 5.15.0-138-generic                                                                                                                                                                                        |
|              | v1.28                      | √                                 | ×                                  | √                                 | 5.15.0-138-generic                                                                                                                                                                                        |

| OS                       | Cluster Version            | CCE Standard Cluster              |                | CCE Turbo Cluster                 | Latest Kernel                                                                                                                                                               |
|--------------------------|----------------------------|-----------------------------------|----------------|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                          |                            | VPC Network                       | Tunnel Network | Cloud Native Network 2.0          |                                                                                                                                                                             |
|                          | v1.27                      | √                                 | ×              | √                                 | <ul style="list-style-type: none"><li>Clusters of v1.27.3-r0 or later 5.15.0-138-generic</li><li>Clusters of versions earlier than v1.27.3-r0 5.15.0-86-generic</li></ul>   |
|                          | v1.25 (end of maintenance) | Supported in v1.25.3-r0 or later. | ×              | Supported in v1.25.3-r0 or later. | <ul style="list-style-type: none"><li>Clusters of v1.25.6-r0 or later 5.15.0-138-generic</li><li>Clusters of versions earlier than v1.25.6-r0 5.15.0-86-generic</li></ul>   |
|                          | v1.23 (end of maintenance) | Supported in v1.23.8-r0 or later. | ×              | Supported in v1.23.8-r0 or later. | <ul style="list-style-type: none"><li>Clusters of v1.23.11-r0 or later 5.15.0-138-generic</li><li>Clusters of versions earlier than v1.23.11-r0 5.15.0-86-generic</li></ul> |
| Huawei Cloud EulerOS 1.1 | v1.32                      | √                                 | √              | √                                 | 3.10.0-1160.76.2.hce1c.x86_64                                                                                                                                               |
|                          | v1.31                      | √                                 | √              | √                                 | 3.10.0-1160.76.2.hce1c.x86_64                                                                                                                                               |
|                          | v1.30                      | √                                 | √              | √                                 | 3.10.0-1160.76.2.hce1c.x86_64                                                                                                                                               |
|                          | v1.29                      | √                                 | √              | √                                 | 3.10.0-1160.76.2.hce1c.x86_64                                                                                                                                               |

| OS                  | Cluster Version               | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                             |
|---------------------|-------------------------------|----------------------|----------------|--------------------------|-------------------------------------------|
|                     |                               | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                           |
|                     | v1.28                         | √                    | √              | √                        | 3.10.0-1160.76.2.hce1c.x86_64             |
|                     | v1.27                         | √                    | √              | √                        | 3.10.0-1160.76.2.hce1c.x86_64             |
|                     | v1.25<br>(end of maintenance) | √                    | √              | √                        | 3.10.0-1160.76.2.hce1c.x86_64             |
|                     | v1.23<br>(end of maintenance) | √                    | √              | √                        | 3.10.0-1160.76.2.hce1c.x86_64             |
|                     | v1.21<br>(end of maintenance) | √                    | √              | √                        | 3.10.0-1160.76.2.hce1c.x86_64             |
| EulerOS release 2.9 | v1.32                         | √                    | √              | √                        | 4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64 |
|                     | v1.31                         | √                    | √              | √                        | 4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64 |
|                     | v1.30                         | √                    | √              | √                        | 4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64 |
|                     | v1.29                         | √                    | √              | √                        | 4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64 |
|                     | v1.28                         | √                    | √              | √                        | 4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64 |
|                     | v1.27                         | √                    | √              | √                        | 4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64 |

| OS                        | Cluster Version               | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                                                                                                                                                                                                                    |
|---------------------------|-------------------------------|----------------------|----------------|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                           |                               | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                                                                                                                                                                                                                  |
|                           | v1.25<br>(end of maintenance) | √                    | √              | √                        | 4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64                                                                                                                                                                                        |
|                           | v1.23<br>(end of maintenance) | √                    | √              | √                        | <ul style="list-style-type: none"><li>Clusters of v1.23.5-r0 or later<br/>4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64</li><li>Clusters of versions earlier than v1.23.5-r0<br/>4.18.0-147.5.1.6.h1017.eulerosv2r9.x86_64</li></ul> |
|                           | v1.21<br>(end of maintenance) | √                    | √              | √                        | <ul style="list-style-type: none"><li>Clusters of v1.21.7-r0 or later<br/>4.18.0-147.5.1.6.h1511.eulerosv2r9.x86_64</li><li>Clusters of versions earlier than v1.21.7-r0<br/>4.18.0-147.5.1.6.h1017.eulerosv2r9.x86_64</li></ul> |
|                           | v1.19<br>(end of maintenance) | √                    | √              | √                        | 4.18.0-147.5.1.6.h1152.eulerosv2r9.x86_64                                                                                                                                                                                        |
| EulerOS release 2.9 (Arm) | v1.32                         | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64                                                                                                                                                                                  |
|                           | v1.31                         | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64                                                                                                                                                                                  |

| OS | Cluster Version            | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                                                                                                                                                                                                                               |
|----|----------------------------|----------------------|----------------|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                            | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                                                                                                                                                                                                                             |
|    | v1.30                      | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64                                                                                                                                                                                             |
|    | v1.29                      | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64                                                                                                                                                                                             |
|    | v1.28                      | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64                                                                                                                                                                                             |
|    | v1.27                      | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64                                                                                                                                                                                             |
|    | v1.25 (end of maintenance) | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64                                                                                                                                                                                             |
|    | v1.23 (end of maintenance) | √                    | √              | √                        | <ul style="list-style-type: none"><li>Clusters of v1.23.5-r0 or later<br/>4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64</li><li>Clusters of versions earlier than v1.23.5-r0<br/>4.19.90-vhulk2103.1.0.h990.eulerosv2r9.aarch64</li></ul> |

| OS                                            | Cluster Version               | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                                                                                                                                                                                                                               |
|-----------------------------------------------|-------------------------------|----------------------|----------------|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                               |                               | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                                                                                                                                                                                                                             |
|                                               | v1.21 (end of maintenance)    | √                    | √              | √                        | <ul style="list-style-type: none"><li>Clusters of v1.21.7-r0 or later<br/>4.19.90-vhulk2103.1.0.h1433.eulerosv2r9.aarch64</li><li>Clusters of versions earlier than v1.21.7-r0<br/>4.19.90-vhulk2103.1.0.h990.eulerosv2r9.aarch64</li></ul> |
|                                               | v1.19 (end of maintenance)    | √                    | √              | √                        | 4.19.90-vhulk2103.1.0.h1144.eulerosv2r9.aarch64                                                                                                                                                                                             |
| EulerOS release 2.8 (Arm, end of maintenance) | v1.27 or later                | ×                    | ×              | ×                        | None                                                                                                                                                                                                                                        |
|                                               | v1.25 (end of maintenance)    | √                    | √              | √                        | 4.19.36-vhulk1907.1.0.h1350.eulerosv2r8.aarch64                                                                                                                                                                                             |
|                                               | v1.23 (end of maintenance)    | √                    | √              | √                        | 4.19.36-vhulk1907.1.0.h1350.eulerosv2r8.aarch64                                                                                                                                                                                             |
|                                               | v1.21 (end of maintenance)    | √                    | √              | √                        | 4.19.36-vhulk1907.1.0.h1350.eulerosv2r8.aarch64                                                                                                                                                                                             |
|                                               | v1.19.16 (end of maintenance) | √                    | √              | √                        | 4.19.36-vhulk1907.1.0.h1350.eulerosv2r8.aarch64                                                                                                                                                                                             |

| OS                                       | Cluster Version               | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                                  |
|------------------------------------------|-------------------------------|----------------------|----------------|--------------------------|------------------------------------------------|
|                                          |                               | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                                |
|                                          | v1.19.10 (end of maintenance) | √                    | √              | √                        | 4.19.36-vhulk1907.1.0.h962.eulerosv2r8.aarch64 |
|                                          | v1.17.17 (end of maintenance) | √                    | √              | √                        | 4.19.36-vhulk1907.1.0.h962.eulerosv2r8.aarch64 |
|                                          | v1.15.11 (end of maintenance) | √                    | √              | √                        | 4.19.36-vhulk1907.1.0.h702.eulerosv2r8.aarch64 |
| EulerOS release 2.5 (end of maintenance) | v1.27 or later                | ×                    | ×              | ×                        | None                                           |
|                                          | v1.25 (end of maintenance)    | √                    | √              | √                        | 3.10.0-862.14.1.5.h687.eulerosv2r7.x86_64      |
|                                          | v1.23 (end of maintenance)    | √                    | √              | √                        | 3.10.0-862.14.1.5.h687.eulerosv2r7.x86_64      |
|                                          | v1.21 (end of maintenance)    | √                    | √              | √                        | 3.10.0-862.14.1.5.h687.eulerosv2r7.x86_64      |
|                                          | v1.19.16 (end of maintenance) | √                    | √              | √                        | 3.10.0-862.14.1.5.h687.eulerosv2r7.x86_64      |
|                                          | v1.19.10 (end of maintenance) | √                    | √              | √                        | 3.10.0-862.14.1.5.h520.eulerosv2r7.x86_64      |



| OS                                            | Cluster Version                  | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                             |
|-----------------------------------------------|----------------------------------|----------------------|----------------|--------------------------|-------------------------------------------|
|                                               |                                  | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                           |
|                                               | v1.19.8 (end of maintenance)     | √                    | √              | √                        | 3.10.0-862.14.1.5.h520.eulerosv2r7.x86_64 |
|                                               | v1.17.17 (end of maintenance)    | √                    | √              | √                        | 3.10.0-862.14.1.5.h470.eulerosv2r7.x86_64 |
|                                               | v1.17.9 (end of maintenance)     | √                    | √              | √                        | 3.10.0-862.14.1.5.h428.eulerosv2r7.x86_64 |
|                                               | v1.15.11 (end of maintenance)    | √                    | √              | √                        | 3.10.0-862.14.1.5.h428.eulerosv2r7.x86_64 |
|                                               | v1.15.6-r1 (end of maintenance)  | √                    | √              | √                        | 3.10.0-862.14.1.5.h328.eulerosv2r7.x86_64 |
|                                               | v1.13.10-r1 (end of maintenance) | √                    | √              | √                        | 3.10.0-862.14.1.2.h249.eulerosv2r7.x86_64 |
|                                               | v1.13.7-r0 (end of maintenance)  | √                    | √              | √                        | 3.10.0-862.14.1.0.h197.eulerosv2r7.x86_64 |
| CentOS Linux release 7.6 (end of maintenance) | v1.32                            | √                    | √              | √                        | 3.10.0-1160.119.1.el7.x86_64              |
|                                               | v1.31                            | √                    | √              | √                        | 3.10.0-1160.119.1.el7.x86_64              |
|                                               | v1.30                            | √                    | √              | √                        | 3.10.0-1160.119.1.el7.x86_64              |
|                                               | v1.29                            | √                    | √              | √                        | 3.10.0-1160.119.1.el7.x86_64              |

| OS | Cluster Version               | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                                                                                                                                                                                                                                                      |
|----|-------------------------------|----------------------|----------------|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                               | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                                                                                                                                                                                                                                                    |
|    | v1.28                         | √                    | √              | √                        | 3.10.0-1160.119.1.el7.x86_64                                                                                                                                                                                                                                       |
|    | v1.27                         | √                    | √              | √                        | 3.10.0-1160.119.1.el7.x86_64                                                                                                                                                                                                                                       |
|    | v1.25<br>(end of maintenance) | √                    | √              | √                        | 3.10.0-1160.119.1.el7.x86_64                                                                                                                                                                                                                                       |
|    | v1.23<br>(end of maintenance) | √                    | √              | √                        | <ul style="list-style-type: none"><li>Clusters of v1.23.3-r0 or later<br/>3.10.0-1160.119.1.el7.x86_64</li><li>Clusters of versions earlier than v1.23.3-r0<br/>3.10.0-1160.66.1.el7.x86_64</li></ul>                                                              |
|    | v1.21<br>(end of maintenance) | √                    | √              | √                        | <ul style="list-style-type: none"><li>Clusters of v1.21.5-r0 or later<br/>3.10.0-1160.119.1.el7.x86_64</li><li>v1.21.4-r0 clusters:<br/>3.10.0-1160.66.1.el7.x86_64</li><li>Clusters of versions earlier than v1.21.4-r0<br/>3.10.0-1160.25.1.el7.x86_64</li></ul> |
|    | v1.19<br>(end of maintenance) | √                    | √              | √                        | 3.10.0-1160.108.1.el7.x86_64                                                                                                                                                                                                                                       |

| OS                                             | Cluster Version                  | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel               |
|------------------------------------------------|----------------------------------|----------------------|----------------|--------------------------|-----------------------------|
|                                                |                                  | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                             |
|                                                | v1.17.17 (end of maintenance)    | √                    | √              | √                        | 3.10.0-1160.15.2.el7.x86_64 |
|                                                | v1.17.9 (end of maintenance)     | √                    | √              | √                        | 3.10.0-1062.12.1.el7.x86_64 |
|                                                | v1.15.11 (end of maintenance)    | √                    | √              | √                        | 3.10.0-1062.12.1.el7.x86_64 |
|                                                | v1.15.6-r1 (end of maintenance)  | √                    | √              | √                        | 3.10.0-1062.1.1.el7.x86_64  |
|                                                | v1.13.10-r1 (end of maintenance) | √                    | √              | √                        | 3.10.0-957.21.3.el7.x86_64  |
|                                                | v1.13.7-r0 (end of maintenance)  | √                    | √              | √                        | 3.10.0-957.21.3.el7.x86_64  |
| Ubuntu 18.04 server 64bit (end of maintenance) | v1.27 or later                   | ×                    | ×              | ×                        | None                        |
|                                                | v1.25 (end of maintenance)       | √                    | ×              | √                        | 4.15.0-171-generic          |
|                                                | v1.23 (end of maintenance)       | √                    | ×              | √                        | 4.15.0-171-generic          |

| OS | Cluster Version                  | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel      |
|----|----------------------------------|----------------------|----------------|--------------------------|--------------------|
|    |                                  | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                    |
|    | v1.21<br>(end of maintenance)    | √                    | ×              | √                        | 4.15.0-171-generic |
|    | v1.19.16<br>(end of maintenance) | √                    | ×              | √                        | 4.15.0-171-generic |
|    | v1.19.8<br>(end of maintenance)  | √                    | ×              | √                        | 4.15.0-136-generic |
|    | v1.17.17<br>(end of maintenance) | √                    | ×              | √                        | 4.15.0-136-generic |

Table 8-6 ECS (PM) OSs

| OS                   | Cluster Version | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                               |
|----------------------|-----------------|----------------------|----------------|--------------------------|---------------------------------------------|
|                      |                 | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                             |
| EulerOS release 2.10 | v1.32           | √                    | √              | √                        | 4.18.0-147.5.2.15.h1109.eulerosv2r10.x86_64 |
|                      | v1.31           | √                    | √              | √                        | 4.18.0-147.5.2.15.h1109.eulerosv2r10.x86_64 |
|                      | v1.30           | √                    | √              | √                        | 4.18.0-147.5.2.15.h1109.eulerosv2r10.x86_64 |

| OS | Cluster Version                  | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                                       |
|----|----------------------------------|----------------------|----------------|--------------------------|-----------------------------------------------------|
|    |                                  | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                                     |
|    | v1.29                            | √                    | √              | √                        | 4.18.0-147.5.2.15.<br>h1109.eulerosv2r1<br>0.x86_64 |
|    | v1.28                            | √                    | √              | √                        | 4.18.0-147.5.2.15.<br>h1109.eulerosv2r1<br>0.x86_64 |
|    | v1.27                            | √                    | √              | √                        | 4.18.0-147.5.2.15.<br>h1109.eulerosv2r1<br>0.x86_64 |
|    | v1.25                            | √                    | √              | √                        | 4.18.0-147.5.2.15.<br>h1109.eulerosv2r1<br>0.x86_64 |
|    | v1.23                            | √                    | √              | √                        | 4.18.0-147.5.2.15.<br>h1109.eulerosv2r1<br>0.x86_64 |
|    | v1.21<br>(end of maintenance)    | √                    | √              | √                        | 4.18.0-147.5.2.15.<br>h1109.eulerosv2r1<br>0.x86_64 |
|    | v1.19.16<br>(end of maintenance) | √                    | √              | √                        | 4.18.0-147.5.2.15.<br>h1109.eulerosv2r1<br>0.x86_64 |

**Table 8-7** BMS OSs

| OS                                                                                                       | Cluster Version | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                            |
|----------------------------------------------------------------------------------------------------------|-----------------|----------------------|----------------|--------------------------|------------------------------------------|
|                                                                                                          |                 | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                          |
| Huawei Cloud EulerOS 2.0 (supported by certain regions and server models. For details, see the console.) | v1.32           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
|                                                                                                          | v1.31           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
|                                                                                                          | v1.30           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
|                                                                                                          | v1.29           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
|                                                                                                          | v1.28           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
|                                                                                                          | v1.27           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
|                                                                                                          | v1.25           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
|                                                                                                          | v1.23           | √                    | √              | ×                        | 5.10.0-60.18.0.50.r1083_58.hce2.x86_64   |
| EulerOS release 2.9 (restricted use. Submit a service ticket for confirmation.)                          | v1.32           | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                                                                                          | v1.31           | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                                                                                          | v1.30           | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |

| OS                                       | Cluster Version            | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                            |
|------------------------------------------|----------------------------|----------------------|----------------|--------------------------|------------------------------------------|
|                                          |                            | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                          |
|                                          | v1.29                      | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                          | v1.28                      | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                          | v1.27                      | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                          | v1.25                      | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                          | v1.23                      | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                          | v1.21 (end of maintenance) | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
|                                          | v1.19 (end of maintenance) | √                    | √              | ×                        | 4.18.0-147.5.1.6.h841.eulerosv2r9.x86_64 |
| EulerOS release 2.3 (end of maintenance) | v1.27 or later             | ×                    | ×              | ×                        | None                                     |
|                                          | v1.25 (end of maintenance) | √                    | √              | ×                        | 3.10.0-514.41.4.28.h62.x86_64            |
|                                          | v1.23 (end of maintenance) | √                    | √              | ×                        | 3.10.0-514.41.4.28.h62.x86_64            |
|                                          | v1.21 (end of maintenance) | √                    | √              | ×                        | 3.10.0-514.41.4.28.h62.x86_64            |

| OS | Cluster Version                  | CCE Standard Cluster |                | CCE Turbo Cluster        | Latest Kernel                     |
|----|----------------------------------|----------------------|----------------|--------------------------|-----------------------------------|
|    |                                  | VPC Network          | Tunnel Network | Cloud Native Network 2.0 |                                   |
|    | v1.19<br>(end of maintenance)    | √                    | √              | ×                        | 3.10.0-514.41.4.28<br>.h62.x86_64 |
|    | v1.17<br>(end of maintenance)    | √                    | √              | ×                        | 3.10.0-514.41.4.28<br>.h62.x86_64 |
|    | v1.15.11<br>(end of maintenance) | √                    | √              | ×                        | 3.10.0-514.41.4.28<br>.h62.x86_64 |

## 8.13 Space Allocation of a Data Disk

This section describes how to allocate data disk space to nodes so that you can configure the data disk space accordingly.

### Allocating Default Data Disk Space

#### NOTE

- In clusters of a version earlier than v1.23.18-r0, v1.25.13-r0, v1.27.10-r0, v1.29.4-r0, or v1.28.8-r0, a default data disk will be added to the node for the container runtime and kubelet components. You can customize the space allocation of the default data disk.
- In clusters of v1.23.18-r0, v1.25.13-r0, v1.27.10-r0, v1.28.8-r0, v1.29.4-r0, or later, a default data disk will be added to the node for the container runtime and kubelet components only if **System Component Storage** is set to **Data Disk**. You can customize the space allocation of the default data disk.

When creating a node, you can customize Data Disk Space Allocation in the expanded area of **Data Disk**.



**Figure 8-13** Allocating data disk space

The screenshot shows the 'Storage Settings' configuration page. At the top, it says 'Configure storage resources for containers and applications on the node.' There are two main sections: 'System Disk' and 'Data Disk'. The 'System Disk' section has a dropdown menu set to 'Ultra-high I/O', a value of '50' with minus and plus buttons, and the unit 'GiB'. Below it is an 'Expand' dropdown and a status 'System Disk Encryption: Not encrypted'. The 'Data Disk' section has a 'Data Disk' button and a 'System Disk' button. Below these is a 'Data Disk' dropdown set to 'Ultra-high I/O', a value of '100' with minus and plus buttons, the unit 'GiB', a 'Quantity' dropdown set to '1' with minus and plus buttons, and a 'Default Data Disk' button. Below the 'Data Disk' section, there is a warning: 'Used by the container runtime and kubelet. Do not uninstall this disk. Otherwise, the node will become unavailable.' followed by links 'Choosing a Data Disk Size' and 'How do I allocate data disk space?'. Below the warning is an 'Expand' dropdown, a status 'Container Engine Space: Shared disk space', 'Pod Space: All', 'Write Mode: Linear', and 'Data Disk Encryption: Not encrypted'. At the bottom, there is an 'Add' button and a note: 'You can add 15 more EVS data disks.'

- **Space Allocation for Container Engines**

- Specified disk space: CCE divides the data disk space for two parts by default. One part is used to store the Docker/containerd working directories, container image data, and image metadata. The other is reserved for kubelet and emptyDir volumes. The available container engine space affects image pulls and container startup and running.
  - Container engine and container image space (90% by default): stores the container runtime working directories, container image data, and image metadata.
  - kubelet and emptyDir space (10% by default): stores pod configuration files, secrets, and mounted storage such as emptyDir volumes.

 **NOTE**

If the sum of the container engine and container image space and the kubelet and emptyDir space is less than 100%, the remaining space will be allocated for user data. You can mount the storage volume to a service path. Do not leave the path empty or set it to a key OS path such as the root directory.

- Shared disk space: In clusters of v1.21.10-r0, v1.23.8-r0, v1.25.3-r0, or later versions, CCE allows **a container engine (Docker/containerd) and kubelet components to share data disk space**.
- **Space Allocation for Pods:** indicates the basesize of a pod. You can set an upper limit for the disk space occupied by each workload pod (including the space occupied by container images). This setting prevents the pods from taking all the disk space available, which may cause service exceptions. It is recommended that the value is less than or equal to 80% of the container engine space. This parameter is related to the node OS and container storage Rootfs and is not supported in some scenarios. For details, see **Mapping Between OS and Container Storage Rootfs**.
- **Write Mode**
  - **Linear:** A linear logical volume integrates one or more physical volumes. Data is written to the next physical volume when the previous one is used up.
  - **Striped:** available only if there are at least two data disks. A striped logical volume stripes data into blocks of the same size and stores them in multiple physical volumes in sequence. This allows data to be concurrently read and written. A storage pool consisting of striped volumes cannot be scaled-out.

## Space Allocation for Container Engines

For nodes using a data disk shared between a container engine and kubelet components, the container storage Rootfs is of the **OverlayFS** type. For details about data disk space allocation, see [Data Disk Shared Between a Container Engine and kubelet Components](#).

For a node using a non-shared data disk (100 GiB for example), the division of the disk space varies depending on the container storage Rootfs type **Device Mapper** or **OverlayFS**. For details about the container storage Rootfs corresponding to different OSs, see [Mapping Between OS and Container Storage Rootfs](#).

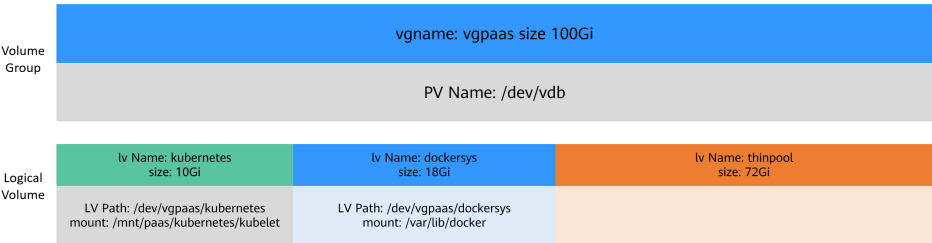
- **Rootfs (Device Mapper)**

By default, the container engine and image space, occupying 90% of the data disk, can be divided into the following two parts:

- The **/var/lib/docker** directory is used as the Docker working directory and occupies 20% of the container engine and container image space by default. (Space size of the **/var/lib/docker** directory = **Data disk space x 90% x 20%**)
- The thin pool is used to store container image data, image metadata, and container data, and occupies 80% of the container engine and container image space by default. (Thin pool space = **Data disk space x 90% x 80%**)

The thin pool is dynamically mounted. You can view it by running the **lsblk** command on a node, but not the **df -h** command.

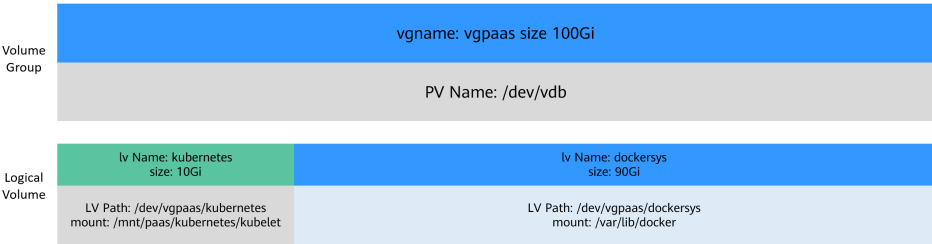
**Figure 8-14** Space allocation for container engines of Device Mapper



- **Rootfs (OverlayFS)**

No separate thin pool. The entire container engine and container image space (90% of the data disk by default) are in the **/var/lib/docker** directory.

**Figure 8-15** Space allocation for container engines of OverlayFS



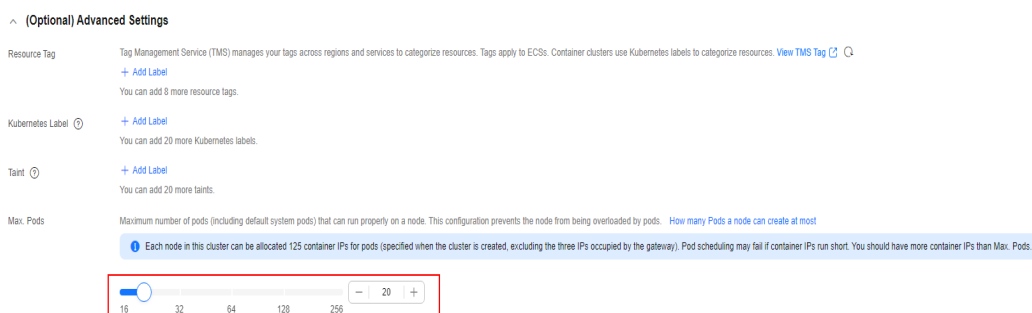
## Space Allocation for Pods

The customized pod container space (**basesize**) is related to the node OS and container storage Rootfs. For details about the container storage Rootfs, see [Mapping Between OS and Container Storage Rootfs](#).

- Device Mapper supports custom pod basesize. The default value is 10 GiB.
- In OverlayFS mode, the pod container space is not limited by default.

When configuring **basesize**, consider the maximum number of pods allowed on one node. The container engine space should be greater than the total disk space used by containers. Formula: **Container engine space and container image space (90% by default) > Number of containers x basesize**. Otherwise, the container engine space allocated to the node may be insufficient and the container cannot be started.

**Figure 8-16** Configuring the maximum number of pods during node creation



For nodes that support **basesize**, when Device Mapper is used, although you can limit the size of the **/home** directory of a single container (to 10 GiB by default), all containers on the node still share the thin pool of the node for storage. They are not completely isolated. When the sum of the thin pool space used by certain containers reaches the upper limit, other containers cannot run properly.

In addition, after a file is deleted in the **/home** directory of the container, the thin pool space occupied by the file is not released immediately. Therefore, even if **basesize** is set to 10 GiB, the thin pool space occupied by files keeps increasing until 10 GiB when files are created in the container. The space released after file deletion will be reused but after a while. If **the number of containers on the node multiplied by basesize** is greater than the thin pool space size of the node, there is a possibility that the thin pool space has been used up.

## Mapping Between OS and Container Storage Rootfs

**Table 8-8** Node OSs and container engines in CCE clusters

| OS          | Container Storage Rootfs                                                                                                                                                                                                                                                          | Custom Basesize                                                                                                                                                                                                                                                                                                                                         |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CentOS 7.x  | Clusters of v1.19.16 and earlier use Device Mapper.<br>Clusters of v1.19.16 and later use OverlayFS.<br><b>NOTE</b><br>When upgrading clusters of earlier versions, Device Mapper nodes will not automatically switch to OverlayFS. You need to manually reset these nodes.       | Supported when Rootfs is set to Device Mapper and the runtime is Docker. The default value is 10 GiB.<br><br>If Rootfs is set to OverlayFS, the basesize cannot be customized.                                                                                                                                                                          |
| EulerOS 2.3 | Device Mapper                                                                                                                                                                                                                                                                     | Supported only when the runtime is Docker. The default value is 10 GiB.                                                                                                                                                                                                                                                                                 |
| EulerOS 2.5 | Device Mapper                                                                                                                                                                                                                                                                     | Supported only when the runtime is Docker. The default value is 10 GiB.                                                                                                                                                                                                                                                                                 |
| EulerOS 2.8 | Clusters of v1.19.16-r2 and earlier use Device Mapper.<br>Clusters of v1.19.16-r2 and later use OverlayFS.<br><b>NOTE</b><br>When upgrading clusters of earlier versions, Device Mapper nodes will not automatically switch to OverlayFS. You need to manually reset these nodes. | Supported when Rootfs is set to Device Mapper and the runtime is Docker. The default value is 10 GiB.<br><br>Supported when Rootfs is set to OverlayFS and the runtime is Docker. There are no limits by default.                                                                                                                                       |
| EulerOS 2.9 | OverlayFS                                                                                                                                                                                                                                                                         | Supported by Docker clusters of v1.19.16-r0, v1.21.3-r0, v1.23.3-r0, or later. There are no limits by default.<br><br>Supported by containerd clusters of v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later. There are no limits by default.<br><br>Not supported by clusters of a version earlier than v1.19.16-r0, v1.21.3-r0, or v1.23.3-r0. |

| OS                       | Container Storage Rootfs | Custom Basesize                                                                                                                                                                                                                                                                                       |
|--------------------------|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| EulerOS 2.10             | OverlayFS                | Supported only by Docker clusters of a version earlier than v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, or v1.28.4-r0. There are no limits by default.<br><br>Supported by both Docker and containerd clusters of v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later. There are no limits by default. |
| Ubuntu 18.04             | OverlayFS                | Not supported                                                                                                                                                                                                                                                                                         |
| Ubuntu 22.04             | OverlayFS                | Not supported                                                                                                                                                                                                                                                                                         |
| Huawei Cloud EulerOS 1.1 | OverlayFS                | Not supported                                                                                                                                                                                                                                                                                         |
| Huawei Cloud EulerOS 2.0 | OverlayFS                | Supported only by Docker clusters of a version earlier than v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, or v1.28.4-r0. There are no limits by default.<br><br>Supported by both Docker and containerd clusters of v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later. There are no limits by default. |

**Table 8-9** Node OSs and container engines in CCE Turbo clusters

| OS           | Container Storage Rootfs | Custom Basesize                                                                                                                                                                                                     |
|--------------|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CentOS 7.x   | OverlayFS                | Not supported                                                                                                                                                                                                       |
| Ubuntu 18.04 | OverlayFS                | Not supported                                                                                                                                                                                                       |
| Ubuntu 22.04 | OverlayFS                | Not supported                                                                                                                                                                                                       |
| EulerOS 2.9  | OverlayFS                | Supported when Rootfs is set to OverlayFS and the runtime is Docker. There are no limits by default. Supported by both Docker and containerd clusters of v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later. |

| OS                       | Container Storage Rootfs    | Custom Basesize                                                                                                                                                                                                                                                                                       |
|--------------------------|-----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| EulerOS 2.10             | Device Mapper for ECSs (PM) | Supported when Rootfs is set to Device Mapper and the runtime is containerd. The default value is 10 GiB.                                                                                                                                                                                             |
| Huawei Cloud EulerOS 1.1 | OverlayFS                   | Not supported                                                                                                                                                                                                                                                                                         |
| Huawei Cloud EulerOS 2.0 | OverlayFS                   | Supported only by Docker clusters of a version earlier than v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, or v1.28.4-r0. There are no limits by default.<br><br>Supported by both Docker and containerd clusters of v1.23.14-r0, v1.25.9-r0, v1.27.6-r0, v1.28.4-r0, or later. There are no limits by default. |

## Garbage Collection Policies for Container Images

When the container engine space is insufficient, image garbage collection is triggered.

The policy for garbage collecting images takes two factors into consideration: **HighThresholdPercent** and **LowThresholdPercent**. Disk usage exceeding the high threshold (default: 80%) will trigger garbage collection. The garbage collection will delete least recently used images until the low threshold (default: 70%) is met.

## Recommended Configuration for the Container Engine Space

- The container engine space should be greater than the total disk space used by containers. Formula: **Container engine space** > **Number of containers** x **basesize**
- You are advised to create and delete files of containerized services in local storage volumes (such as emptyDir and hostPath volumes) or cloud storage directories mounted to the containers. In this way, the thin pool space is not occupied. emptyDir volumes occupy the kubelet space. Therefore, properly plan the size of the kubelet space.
- You can deploy services on nodes that use the OverlayFS (for details, see [Mapping Between OS and Container Storage Rootfs](#)) so that the disk space occupied by files created or deleted in containers can be released immediately.

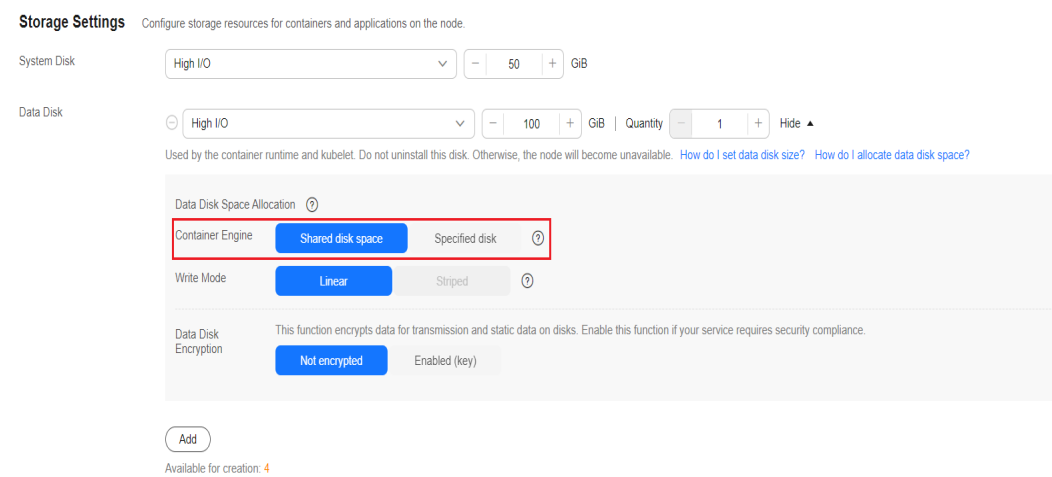
## Data Disk Shared Between a Container Engine and kubelet Components

Docker/containerd and kubelet components share the space of a data disk.

NOTICE

- This function is available only to clusters of v1.21.10-r0, v1.23.8-r0, v1.25.3-r0, or later versions.
- If Rootfs is set to OverlayFS, shared data disks are supported. If Rootfs is set to Device Mapper, shared data disks are not supported.
- If you have installed an NPD add-on in the cluster, upgrade the add-on to v1.18.10 or later. Otherwise, false alarms will be generated.
- If you have installed a log-agent add-on in the cluster, upgrade the add-on to v1.3.0 or later. Otherwise, log collection will be affected.
- If you have installed ICAgent in the cluster, upgrade it to v5.12.140 or later. Otherwise, log collection will be affected. For details about how to view or upgrade an ICAgent version, see [CCE Access](#).

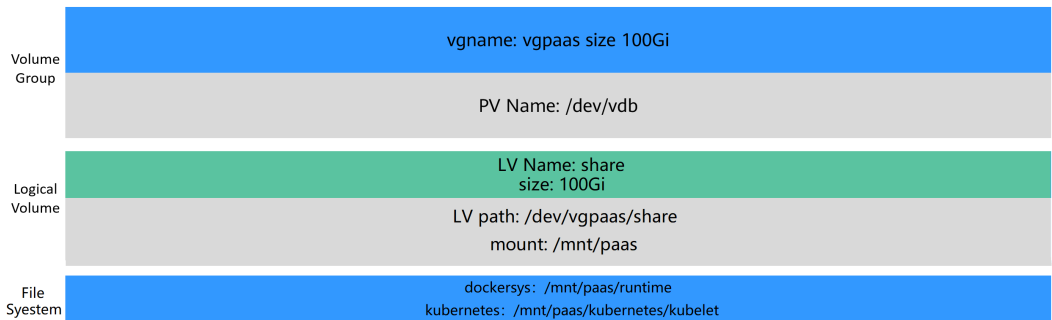
Figure 8-17 Configuration for sharing disk space



For nodes using a shared data disk, the container storage Rootfs is of the **OverlayFS** type. After such a node is created, the data disk space (for example, 100 GiB) will not be divided for the container engines, container images, and kubelet components. The data disk is mounted to **/mnt/paas**, and the storage space is divided using two file systems.

- dockersys: /mnt/paas/runtime
- Kubernetes: /mnt/paas/kubernetes/kubelet

Figure 8-18 Allocating the storage space of a shared data disk



## Helpful Links

[How Do I Expand the Storage Capacity of a Container?](#)

[Expanding the Disk Capacity of a Node in a CCE Cluster](#)

## 8.14 Attaching Disks to a Node

### Challenges

In disk planning and striped logical disk creation, it is difficult for users to flexibly attach and partition disks when creating a node.

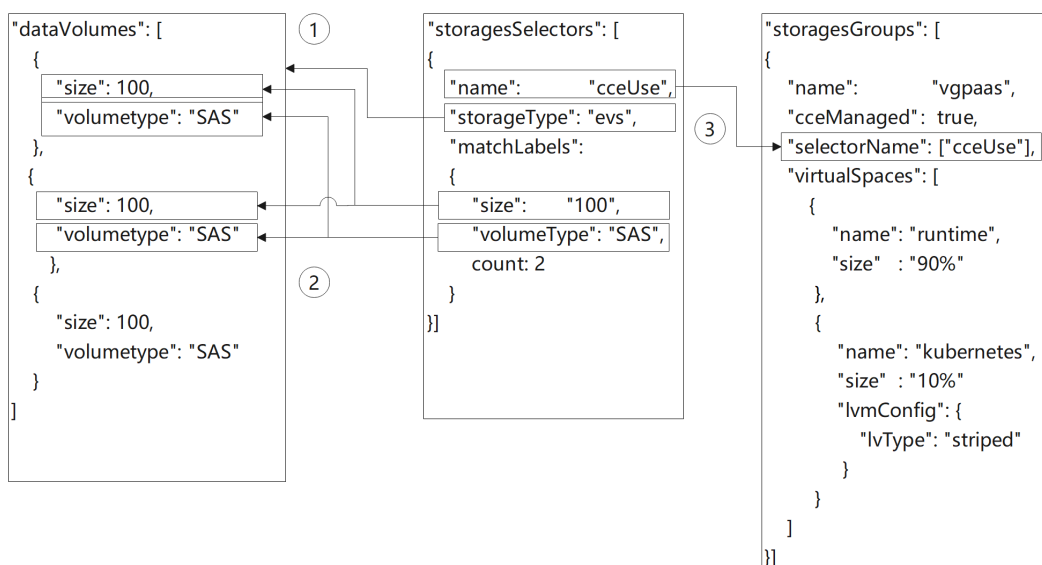
During node creation, the **storage** field selects a data disk based on parameters, such as the disk size and disk type, to prevent failures in node creation, resetting, migration, and management caused by drive letter matching failures. For example, if **DockerLVMConfigOverride.diskType** in the NodeExtendParam field is set to **evs** during node creation, C7 nodes will not be created.

### Solution

This section provides details about the **storage** field used in node creation so that you can implement complex disk selection and partitioning by calling the node creation API.

The **storage** field consists of **storageSelectors** and **storageGroups**. The **storageSelectors** field is responsible for disk selection, and the **storageGroups** field is responsible for disk processing.

The basic logic for field matching is as follows:



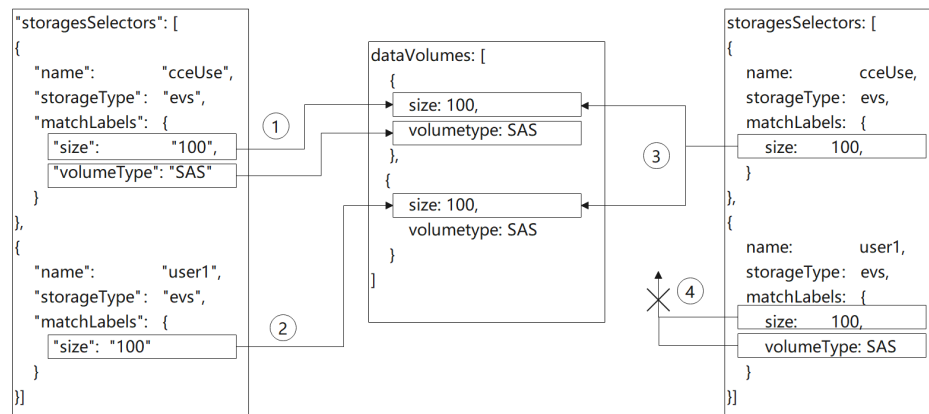
1. **storageSelectors** selects an EVS disk or a local disk based on the value of **storageType**.
  - a. Local disks do not support exact match. All local disks will be selected as data disks.



If some local disks need to be reserved, occupy the disks in the pre-installation script. The script is similar to the following:

```
prepare
vgName=vg-test
storageDevice=/dev/vdb
vgcreate
vgcreate ${vgName} ${storageDevice}
```

- b. EVS disks match the disks created in **dataVolumes** based on the settings of **matchLabels**.
2. Policies have priorities to match **matchLabels**. The **matchLabels** policy nearest to **storageSelectors** has a higher priority and the disk nearest to **dataVolumes** will be preferentially matches. As **matchLabels** uses loose matching, you are advised to place the **matchLabels** policy with a small matching range on the top. For example:



- a. In step 1, the first disk in **dataVolumes** is matched by the EVS disk whose size is 100 GiB and storage class is SAS. In step 2, the second disk in **dataVolumes** is matched by the EVS disk whose size is 100 GiB because the first disk has been selected.
- b. In step 3, two disks in **dataVolumes** can be matched because **volumeType** or **count** is not specified in **matchLabels**. In this case, no disk is available for matching in step 4.
3. **storageGroups** associates with **storageSelectors** based on **selectorName**. Finally, two 100 GiB disks are selected. The CCE backend groups the two PVs into a volume group (VG) and divides the VG into two logical volumes (LVs) in the ratio of 9:1. 10% of Kubernetes LVs are partitioned in striped mode. 90% runtime LVs are partitioned in linear mode by default because runtimeConfig is not configured.

## Creating a Raw Disk

During node creation on the CCE console, click **Add Data Disk**. Then, click **Expand** next to the newly added data disk and select **Default**. The created disk is a raw disk.

**Storage Settings** Configure storage resources for containers and applications on the node.

System Disk General-purpose SSD - 50 + GiB

Data Disk General-purpose SSD - 100 + GiB | Quantity - 1 +

Used by the container runtime and kubelet. Do not uninstall this disk. Otherwise, the node will become unavailable. [How do I set data disk size?](#) [How do I allocate data disk space?](#)

Expand ▼ Container Engine: Shared disk space | Pod: All | Write Mode: Linear | Data Disk Encryption: Not encrypted

General-purpose SSD - 100 + GiB | Quantity - 1 +

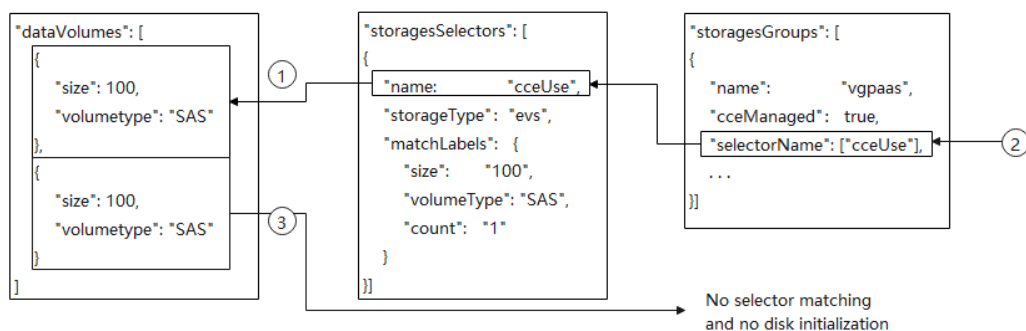
For a common data disk, you can choose not to perform any operation (by default) or attach it in a specified mode.

Hide ^

Mount Settings ☒ Default ☐ Mount Disk ☐ Use as PV ☐ Use as ephemeral volume

Data Disk Encryption This function encrypts data for transmission and static data on disks. Enable this function if your service requires security compliance.

The following figure shows the API calling logic.



1. The **cceUse** selector matches a 100 GiB data disk.
2. The selected disk is managed by CCE and used as a data disk.
3. The other 100 GiB data disk created in **dataVolumes** is not selected by any selector and is managed by storageGroups. Therefore, this EVS disk will be attached to the node as a raw disk and will not be initialized.

After the node is created, log in to the node and check whether a 100 GiB disk has been attached but not initialized.

```
[root@test-83790 ~]# lsblk -n
sda 8:0 0 50G 0 disk
└─sda1 8:1 0 50G 0 part /
sdb 8:16 0 100G 0 disk
├─vgpaas-dockersys 253:0 0 18G 0 lvm /var/lib/docker
├─vgpaas-thinpool_tmeta 253:1 0 3G 0 lvm
├─vgpaas-thinpool 253:3 0 67G 0 lvm
│ └─docker-253:0-786433-7cb37dc21202bfe2fc78dd1d33b70571e7e1982e56a4118f6acdc630cbc8b38 253:5 0 10G 0 dm /var/lib/docker/devicemapper/n
│ └─docker-253:0-786433-e17cd8670b9f423eaff34b92bd82a2e620118227c26da2e41eda7894361c9942 253:6 0 10G 0 dm /var/lib/docker/devicemapper/n
│ └─docker-253:0-786433-0dedb47e75eed3f635ce2d47c584587ae622c70dc0b0eafade9e14693a3146a0 253:7 0 10G 0 dm /var/lib/docker/devicemapper/n
│ └─docker-253:0-786433-93ed7e6e14313d13ecfa1152937b153fe599c48cfdaf9ecd43c1c36cae89a38a 253:8 0 10G 0 dm /var/lib/docker/devicemapper/n
│ └─docker-253:0-786433-e706be08bf5c6249850a09e080cf43d9a7be499eae33aa8feb06c027d26fa1e9 253:9 0 10G 0 dm /var/lib/docker/devicemapper/n
│ └─docker-253:0-786433-5ecc4420da9a58fb66108db599a8267af3e8856da86b9c3d7fb82090a8781ae8 253:10 0 10G 0 dm /var/lib/docker/devicemapper/n
└─vgpaas-thinpool_tdata 253:2 0 67G 0 lvm
 └─vgpaas-thinpool 253:3 0 67G 0 lvm
 └─docker-253:0-786433-7cb37dc21202bfe2fc78dd1d33b70571e7e1982e56a4118f6acdc630cbc8b38 253:5 0 10G 0 dm /var/lib/docker/devicemapper/n
 └─docker-253:0-786433-e17cd8670b9f423eaff34b92bd82a2e620118227c26da2e41eda7894361c9942 253:6 0 10G 0 dm /var/lib/docker/devicemapper/n
 └─docker-253:0-786433-0dedb47e75eed3f635ce2d47c584587ae622c70dc0b0eafade9e14693a3146a0 253:7 0 10G 0 dm /var/lib/docker/devicemapper/n
 └─docker-253:0-786433-93ed7e6e14313d13ecfa1152937b153fe599c48cfdaf9ecd43c1c36cae89a38a 253:8 0 10G 0 dm /var/lib/docker/devicemapper/n
 └─docker-253:0-786433-e706be08bf5c6249850a09e080cf43d9a7be499eae33aa8feb06c027d26fa1e9 253:9 0 10G 0 dm /var/lib/docker/devicemapper/n
 └─docker-253:0-786433-5ecc4420da9a58fb66108db599a8267af3e8856da86b9c3d7fb82090a8781ae8 253:10 0 10G 0 dm /var/lib/docker/devicemapper/n
└─vgpaas-kubernetes 253:4 0 10G 0 lvm /mnt/paas/kubernetes/kubelet
sdc 8:32 0 100G 0 disk
```

The following is an API example:

```
{
 "kind": "Node",
 "apiVersion": "v3",
 "metadata": {
```

```
 "name": "test-83790"
 },
 "spec": {
 "flavor": "c3.large.2",
 "az": "ap-southeast-1a",
 "os": "EulerOS 2.9",
 "dataVolumes": [
 {
 "size": 100,
 "volumetype": "SAS"
 },
 {
 "size": 100,
 "volumetype": "SAS"
 }
],
 "billingMode": 0,
 "extendParam": {
 "maxPods": 110
 },
 "nodeNicSpec": {
 "primaryNic": {
 "subnetId": "ca964acf-8468-4735-8229-97940ef6c881"
 }
 },
 "rootVolume": {
 "size": 50,
 "volumetype": "SAS"
 },
 "runtime": {
 "name": "docker"
 },
 "login": {
 "userPassword": {
 "username": "root",
 "password": "*****"
 }
 },
 "storage": {
 "storageSelectors": [
 {
 "name": "cceUse",
 "storageType": "evs",
 "matchLabels": {
 "size": "100",
 "volumeType": "SAS",
 "count": "1"
 }
 }
]
 },
 "storageGroups": [
 {
 "name": "vgpaas",
 "selectorNames": [
 "cceUse"
],
 "cceManaged": true,
 "virtualSpaces": [
 {
 "name": "runtime",
 "size": "90%"
 },
 {
 "name": "kubernetes",
 "size": "10%"
 }
]
 }
]
 }
}
```

```
 },
 "count": 1
 }
}
```

## Attaching a Disk to a Specified Path

During node creation on the CCE console, click **Add Data Disk**. Then, click **Expand** next to the newly added data disk, select **Mount Disk**, and set the mount path. In this case, CCE initializes and attaches the disk by default.

**Storage Settings** Configure storage resources for containers and applications on the node.

System Disk General-purpose SSD - 50 + GiB

Data Disk General-purpose SSD - 100 + GiB | Quantity - 1 +

Used by the container runtime and kubelet. Do not uninstall this disk. Otherwise, the node will become unavailable. [How do I set data disk size?](#) [How do I allocate data disk space?](#)

**Expand** Container Engine: Shared disk space | Pod: All | Write Mode: Linear | Data Disk Encryption: Not encrypted

General-purpose SSD - 100 + GiB | Quantity - 1 +

For a common data disk, you can choose not to perform any operation (by default) or attach it in a specified mode.

**Hide** ^

Mount Settings Default **Mount Disk** Use as PV Use as ephemeral volume

Mount Path /tmp2 ⓘ

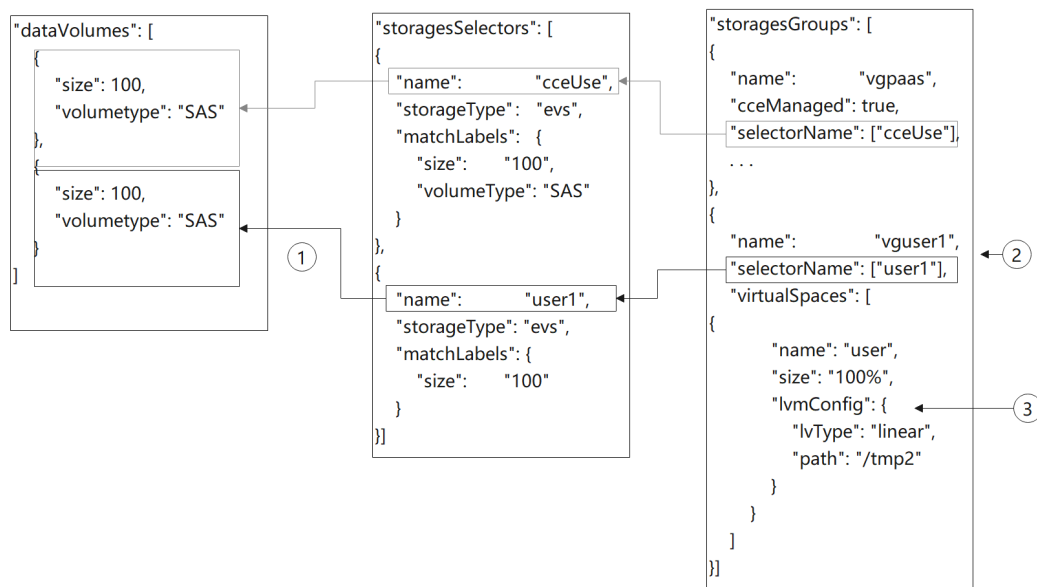
Set the mount path to the service path. Do not leave the mount path empty or set it to a key OS path such as the root directory.

Write Mode **Linear** Striped ⓘ

Data Disk Encryption This function encrypts data for transmission and static data on disks. Enable this function if your service requires security compliance.

**Not encrypted** Enabled (key) Enabled (KMS key ID)

The following figure shows the API calling logic.



1. The **user1** selector selects a 100 GiB data disk.
2. Create a VG named **vguser1** using LVM.
3. Strip all the space of **vguser1** into an LV named **user** and format the disk in ext4 format. Finally, attach the disk to the **/tmp2** directory.

After the node is created, log in to the node and check whether a 100 GiB disk has been attached and managed by LVM.

```
[root@test-37106 ~]# lsblk -n
sda 8:0 0 50G 0 disk
├─sda1 8:1 0 50G 0 part /
├─sdb 8:16 0 100G 0 disk
├─vgpaas-dockersys 253:0 0 16G 0 lvm /var/lib/docker
├─vgpaas-thinpool_tmeta 253:1 0 3G 0 lvm
├─vgpaas-thinpool 253:3 0 59G 0 lvm
├─docker-253:0-917505-3a36be80c1a49db5da9639d222f19ce5983489080a36efdda1f17fa2d0bb7da9 253:6 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-46a876d16929a54d4f5ea97da81c3603c79cd5630be1c1010b476387a5d3c086 253:7 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-93081c85109968299fdca13a077e82252e725a6e37cae7299841db482656b815 253:8 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-513c5bda896de61ac85d917366da4ea4d78ab9f87cd4caae9e465badc0003c62 253:9 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-a6ac0d3ae8bffb57a92e6812079e503db49942619d5bbc69bb9516b31e15e67 253:10 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-f9dfa31cdc3eb514a797c98311372ac8497d9a99581acdfefff0114bdf8e525 253:11 0 10G 0 dm /var/lib/docker/devicemapper/
├─vgpaas-thinpool_tdata 253:2 0 59G 0 lvm
├─vgpaas-thinpool 253:3 0 59G 0 lvm
├─docker-253:0-917505-3a36be80c1a49db5da9639d222f19ce5983489080a36efdda1f17fa2d0bb7da9 253:6 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-46a876d16929a54d4f5ea97da81c3603c79cd5630be1c1010b476387a5d3c086 253:7 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-93081c85109968299fdca13a077e82252e725a6e37cae7299841db482656b815 253:8 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-513c5bda896de61ac85d917366da4ea4d78ab9f87cd4caae9e465badc0003c62 253:9 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-a6ac0d3ae8bffb57a92e6812079e503db49942619d5bbc69bb9516b31e15e67 253:10 0 10G 0 dm /var/lib/docker/devicemapper/
├─docker-253:0-917505-f9dfa31cdc3eb514a797c98311372ac8497d9a99581acdfefff0114bdf8e525 253:11 0 10G 0 dm /var/lib/docker/devicemapper/
├─vgpaas-kubernetes 253:4 0 20G 0 lvm /mnt/paas/kubernetes/kubelet
├─vguser1-user 253:5 0 100G 0 lvm /tmp2
sdc 8:32 0 100G 0 disk
```

The following is an API example. There are two data disks. One is used by CCE, and the other is mounted to the **/tmp2** directory.

```
{
 "kind": "Node",
 "apiVersion": "v3",
 "metadata": {
 "name": "test-37106"
 },
 "spec": {
 "flavor": "c3.large.2",
 "az": "ap-southeast-1a",
 "os": "EulerOS 2.9",
 "dataVolumes": [
 {
 "size": 100,
 "volumetype": "SAS"
 },
 {
 "size": 100,
 "volumetype": "SAS"
 }
],
 "billingMode": 0,
 "extendParam": {
 "maxPods": 110
 },
 "nodeNicSpec": {
 "primaryNic": {
 "subnetId": "ca964acf-8468-4735-8229-97940ef6c881"
 }
 },
 "rootVolume": {
 "size": 50,
 "volumetype": "SAS"
 },
 "runtime": {
 "name": "docker"
 },
 "login": {
 "userPassword": {
 "username": "root",
 "password": "*****"
 }
 },
 "storage": {
 "storageSelectors": [
 {
 "name": "cceUse",

```

```
 "storageType": "evs",
 "matchLabels": {
 "size": "100",
 "volumeType": "SAS",
 "count": "1"
 }
 },
 {
 "name": "user1",
 "storageType": "evs",
 "matchLabels": {
 "size": "100",
 "volumeType": "SAS",
 "count": "1"
 }
 }
],
 "storageGroups": [
 {
 "name": "vgpaas",
 "selectorNames": [
 "cceUse"
],
 "cceManaged": true,
 "virtualSpaces": [
 {
 "name": "runtime",
 "size": "80%"
 },
 {
 "name": "kubernetes",
 "size": "20%"
 }
]
 },
 {
 "name": "vguser1",
 "selectorNames": [
 "user1"
],
 "virtualSpaces": [
 {
 "name": "user",
 "size": "100%",
 "lvmConfig": {
 "lvType": "linear",
 "path": "/tmp2"
 }
 }
]
 }
]
 },
 "count": 1
}
```

## Creating Striped LVs to Improve Disk Performance

Currently, the striped LV function is supported only by calling an API. The following is an example:



1. **storageSelectors** matches all EVS disks in **dataVolumes** because **matchLabels** is not contained in **storageSelectors**.
2. Create a VG named **vgpaas** using LVM.
3. Strip 90% of the **vgpaas** space into runtime LVs.
4. Strip 10% of the **vgpaas** space into Kubernetes LVs.

#### NOTE

- Two or more data disks are required for striping.
- When creating a striped LV, ensure that the types and sizes of the PVs added to the VG are the same. Otherwise, the creation will fail.
- When creating a striped LV, use the striping configuration for both the runtime LV and Kubernetes LV. Otherwise, the creation will fail.

Log in to the node and run the following command to view the striping result:

```
[root@test-83773 ~]# lvdisplay -m | grep -C 10 striped
LV Size 36.00 GiB
Current LE 9216
Segments 1
Allocation inherit
Read ahead sectors auto
- currently set to 512
Block device 253:0

--- Segments ---
Logical extents 0 to 9215:
 Type striped
 Stripes 2
 Stripe size 64.00 KiB
 Stripe 0:
 Physical volume /dev/sdb
 Physical extents 0 to 4607
 Stripe 1:
 Physical volume /dev/sdc
 Physical extents 0 to 4607

--
LV Size 20.00 GiB
Current LE 5120
Segments 1
Allocation inherit
Read ahead sectors auto
- currently set to 8192
Block device 253:4

--- Segments ---
Logical extents 0 to 5119:
 Type striped
 Stripes 2
 Stripe size 64.00 KiB
 Stripe 0:
 Physical volume /dev/sdb
```

The following is an API example:

```
{
 "kind": "Node",
 "apiVersion": "v3",
 "metadata": {
 "name": "test-83773"
 },
 "spec": {
 "flavor": "c3.large.2",
 "az": "ap-southeast-1a",
 "os": "EulerOS 2.9",
 "dataVolumes": [
 {
 "size": 100,
 "volumetype": "SAS"
 }
]
 }
}
```



```
{
 {
 "size": 100,
 "volumetype": "SAS"
 }
],
"billingMode": 0,
"extendParam": {
 "maxPods": 110
},
"nodeNicSpec": {
 "primaryNic": {
 "subnetId": "ca964acf-8468-4735-8229-97940ef6c881"
 }
},
"rootVolume": {
 "size": 50,
 "volumetype": "SAS"
},
"runtime": {
 "name": "docker"
},
"login": {
 "userPassword": {
 "username": "root",
 "password": "*****"
 }
},
"storage": {
 "storageSelectors": [
 {
 "name": "cceUse",
 "storageType": "evs"
 }
],
 "storageGroups": [
 {
 "name": "vgpaas",
 "selectorNames": [
 "cceUse"
],
 "cceManaged": true,
 "virtualSpaces": [
 {
 "name": "runtime",
 "size": "90%",
 "runtimeConfig": {
 "lvType": "striped"
 }
 },
 {
 "name": "kubernetes",
 "size": "10%",
 "lvmConfig": {
 "lvType": "striped"
 }
 }
]
 }
]
},
"count": 1
}
```

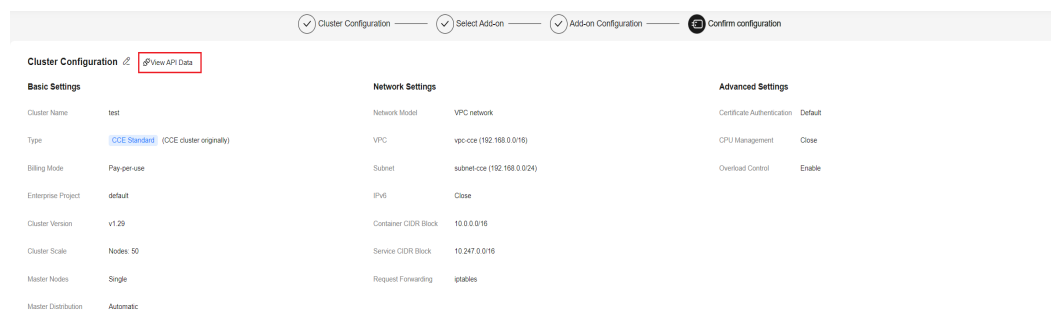
## 8.15 Generating API Parameters on the Console

If you input incorrect API parameters when creating a cluster or node using an API, the API call will fail. To avoid this, you can generate API parameters on the console. The console will automatically generate the correct parameters based on your configuration.

### Generating API Parameters for Creating a Cluster

- Step 1** Log in to the [CCE console](#).
- Step 2** On the **Clusters** page, click **Buy Cluster** in the upper right corner.
- Step 3** Configure cluster parameters as required. For details, see [Buying a CCE Standard/Turbo Cluster](#).
- Step 4** After the configuration is complete, view the API data generated based on the configuration on the **Confirm Configuration** page. You can download or copy the API data.

**Figure 8-19** Generating API parameters to create a cluster



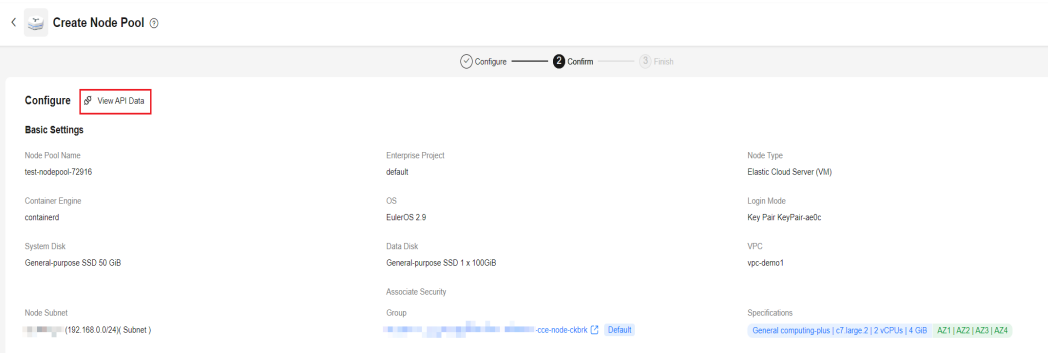
- Step 5** Use the generated API data as the body to call the API for creating a cluster. For details, see [Creating a Cluster](#).

----End

### Generating API Parameters for Creating a Node Pool or Node

- Step 1** Log in to the [CCE console](#).
- Step 2** Log in to the CCE console, choose **Clusters**, and click the name of the target cluster to access the cluster console.
- Step 3** In the navigation pane, choose **Nodes**.
- To create a node pool, configure the parameters by referring to [Creating a Node Pool](#).
  - To create a node, configure the parameters by referring to [Creating a Node](#).
- Step 4** After the configuration is complete, view the API data generated based on the configuration on the **Confirm** page. You can download or copy the API data.

Figure 8-20 Generating API parameters to create a node pool or node



**Step 5** Use the generated API data as the body to call the API for creating a node pool or node. For details, see [Creating a Node Pool](#) or [Creating a Node](#).

----End