SoftWare Repository for Container

User Guide

Issue 07

Date 2025-09-19

HUAWEI

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are trademarks 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 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.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Introduction

SoftWare Repository for Container (SWR) provides easy, secure, and reliable management of container images throughout their lifecycles, including image push, pull, and deletion.

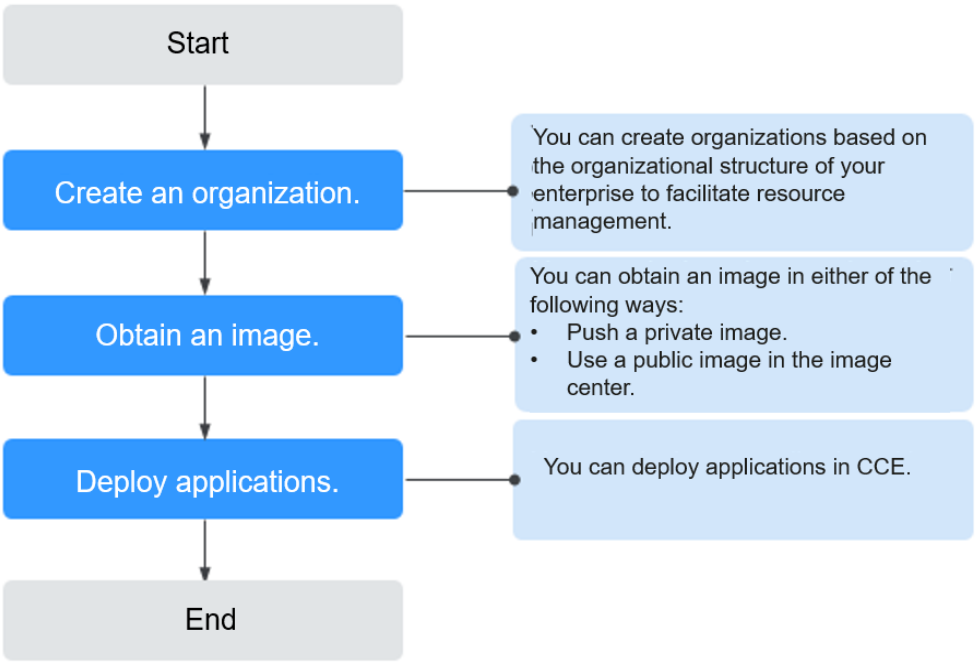
SWR provides private image repositories and fine-grained permissions management, allowing you to grant different access permissions, namely, read, edit, and manage, to different users. Simply set a trigger for an image. Every time the image is updated, the applications deployed in Cloud Container Engine (CCE) with this image will be automatically updated.

You can access SWR on the **Console** or through **APIs**.

NOTE

SWR is a free service.

Figure 1-1 How SWR works



2 Using IAM to Grant Access to SWR

2.1 Using IAM Roles or Policies to Grant Access to SWR

System-defined permissions in role/policy-based authorization provided by **Identity and Access Management (IAM)** let you control access to your SWR resources. With IAM, you can:

- Create IAM users or user groups for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing SWR resources.
- Grant users only the permissions required to perform a given task based on their job responsibilities.
- Entrust other Huawei Cloud account or cloud service to perform efficient O&M on your SWR resources.

If your Huawei Cloud account does not require individual IAM users for permissions management, you can skip this section.

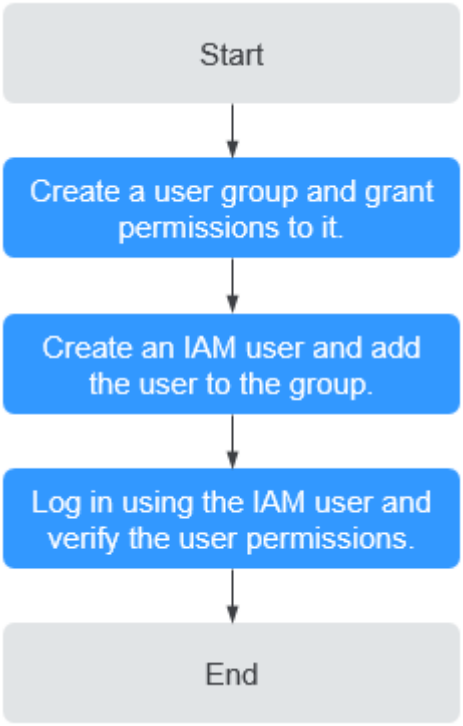
Figure 2-1 shows the process flow of role/policy-based authorization.

Prerequisites

Before granting permissions to user groups, learn about system-defined permissions for SWR. For details, see **Permissions**. To grant permissions for other services, learn about all **system-defined permissions**.

Process Flow

Figure 2-1 Process of granting SWR permissions



1. **Create a user group and assign permissions** to it.

   Create a user group on the IAM console, and grant the **SWR Admin** permissions to the group.

2. **Create an IAM user and add it to the user group**.

   Create a user on the IAM console and add the user to the group created in **1**.

3. **Log in** and verify permissions.

   Log in to the management console using the IAM user. Switch to the authorized region. If the following operations can be performed, the permissions are assigned successfully:

   a. Choose **Service List** > **SoftWare Repository for Container**. The SWR console is displayed.

   b. In the navigation pane, choose **Organizations**. Click **Create Organization** in the upper right corner. Enter an organization name to create an organization.

   c. In the navigation pane, choose **My Images**. Click **Upload Through SWR** in the upper right corner. Select the new organization. Upload a local image file to SWR.

SWR Resources

A resource is an object that exists within a service. SWR resources include organizations and images. When creating a policy, you can select a resource by specifying its path.

Table 2-1 SWR resources and their paths

Resource	Path
namespace	[Format] swr:*:*:namespace:*organization name* [Note] IAM automatically generates the path prefix **SWR:*:*:namespace:**. For the path of a specific organization, add the *organization name* to the end. You can also use a wildcard character (*) to indicate any organization. Example: **swr:*:*:namespace:*** indicates any organization.
repo	[Format] swr:*:*:repo:*image repository name* [Note] IAM automatically generates the path prefix **SWR:*:*:repo:**. For the path of a specific repository, add the *image repository name* to the end. You can also use a wildcard character (*) to indicate any image repository. Example: **SWR:*:*:repo:*** indicates any image repository.

Example 1: If you only allow users to query brief information about image repositories, configure a policy as follows:

```
{
"Version": "5.0"
"Statement": [
{
"Effect": "Allow",
"Action": [
"swr:repo:getRepo"
],
"Resource": [
"swr:*:*:repo:*"
]
}
]
}
```


Example 2: To synchronize an image **test** from the **source** organization in the **cn-north-4** region to the **target** organization in the **cn-north-7** organization, a user needs to have permission to create auto image synchronization tasks and pull images in the **cn-north-4** region, permission to obtain temporary login commands in the **cn-north-4** and **cn-north-7** regions, and permission to push images in the **cn-north-7** region. Configure a policy as follows:

```
{
"Version": "5.0",
"Statement": [
{
"Effect": "Allow",
"Action": [
"swr:repo:createAutoSyncRepoJob",
"swr:repo:download"
],
"Resource": [
"swr:cn-north-4:*:repo:source/test"
]
},
{
"Effect": "Allow",
"Action": [
"swr:repo:upload"
],
"Resource": [
"swr:cn-north-7:*:repo:target"
]
},
{
"Effect": "Allow",
"Action": [
"swr::createLoginSecret"
]
}
]
```



```
}
```

2.2 Using IAM Identity Policies to Grant Access to SWR

System-defined permissions in identity policy-based authorization provided by **Identity and Access Management (IAM)** let you control access to your SWR resources. With IAM, you can:

- Create IAM users or user groups for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing SWR resources.
- Grant users only the permissions required to perform a given task based on their job responsibilities.
- Entrust a Huawei Cloud account or a cloud service to perform efficient O&M on your SWR resources.

If your Huawei Cloud account meets your permissions requirements, you can skip this section.

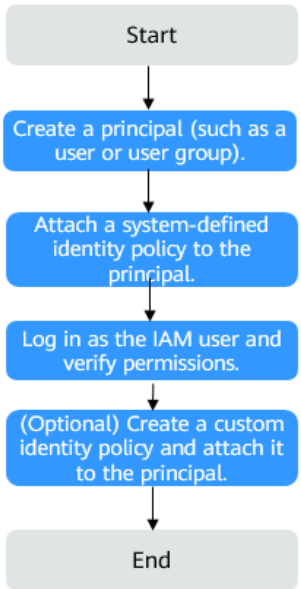
Figure 2-2 shows the process flow of identity policy-based authorization.

Prerequisites

Before granting permissions, learn about system-defined permissions in **Identity Policy-based Authorization**.

Process Flow

Figure 2-2 Process of granting SWR permissions



1. On the IAM console, **create an IAM user** or **create a user group**.
2. **Attach a system-defined identity policy** to the user or user group.

Assign the permissions defined in the system-defined identity policy **SWRReadOnlyPolicy** to the user or group, or attach the system-defined identity policy to it.

3. **Log in as the IAM user** and verify permissions.

   In the authorized region, perform the following operations:

   - Choose **Service List** > **SoftWare Repository for Container**. On the SWR console, click **Create Organization**. If you are not allowed for this operation, the **SWRReadOnlyPolicy** has taken effect (assume that you only have the **SWRReadOnlyPolicy** assigned).
   - Choose any other service from **Service List**. If a message appears indicating insufficient permissions to access the service, the **SWRReadOnlyPolicy** has taken effect.

Example Custom Identity Policies

Custom identity policies can be created to supplement system-defined identity policies. You can add actions in custom identity policies as needed. For details about supported actions, see **Table 2-2**.

To create a custom identity policy, choose either visual editor or JSON.

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Create a JSON policy or edit an existing one.

For details, see **Creating a Custom Identity Policy and Attaching It to a Principal**.

- Example: Grant permission to create, view, and delete organizations.

```
{
"Version": "5.0",
"Statement": [
{
"Effect": "Allow",
"Action": [
"swr:namespace:createNamespace",
"swr:namespace:deleteNamespace",
"swr:namespace:listNamespaces",
"swr:namespace:deleteNamespaceAccess",
"swr:namespace:getNamespaceAccess"
]
}
]
}
```

Actions Supported by System-defined Identity Policies

Table 2-2 Actions supported by system-defined identity policies

Operation	Action	SWR Admin
Creating an organization	swr:namespace:createNamespace	√
Deleting an organization	swr:namespace:deleteNamespace	√
Listing organizations	swr:namespace:listNamespaces	√
Querying details about an organization	swr:namespace:getNamespace	√
Creating a repository in an organization	swr:repo:createRepo	√
Deleting repositories from an organization	swr:repo:deleteRepo	√
Listing repositories	swr:repo:listRepos	√
Listing shared images	swr:repo:listSharedRepos	√
Querying brief information about a repository	swr:repo:getRepo	√
Updating brief information about a repository	swr:repo:updateRepo	√
Deleting images with a specified tag from a repository	swr:repo:deleteRepoTag	√
Listing image tags	swr:repo:listRepoTags	√
Creating an account used for image sharing	swr:repo:createRepoDomain	√
Deleting an account used for image sharing	swr:repo:deleteRepoDomain	√
Listing accounts used for image sharing	swr:repo:listRepoDomains	√
Checking whether an account used for image sharing exists	swr:repo:getRepoDomain	√
Updating an account used for image sharing	swr:repo:updateRepoDomain	√

Operation	Action	SWR Admin
Creating an automatic image synchronization task	swr:repo:createAutoSyncRepoJob	√
Manually synchronizing images	swr:repo:createManualSyncRepoJob	√
Deleting an automatic image synchronization task	swr:repo:deleteAutoSyncRepoJob	√
Listing automatic image synchronization tasks	swr:repo:listAutoSyncRepoJobs	√
Querying details about an automatic image synchronization task	swr:repo:getSyncRepoJobInfo	√
Creating a trigger	swr:repo:createTrigger	√
Deleting a trigger	swr:repo:deleteTrigger	√
Listing triggers in a repository	swr:repo:listTriggers	√
Querying details about a trigger	swr:repo:getTrigger	√
Updating a trigger	swr:repo:updateTrigger	√
Creating an image retention policy	swr:repo:createRetention	√
Deleting an image retention policy	swr:repo:deleteRetention	√
Listing image retention records	swr:repo:listRetentionHistories	√
Listing image retention policies	swr:repo:listRetentions	√
Querying details about an image retention policy	swr:repo:getRetention	√
Updating an image retention policy	swr:repo:updateRetention	√
Generating a temporary login command	swr::createLoginSecret	√
Listing quotas	swr::listQuotas	√

Operation	Action	SWR Admin
Querying the tenant resource overview	swr::getDomainOverview	√
Querying tenant resource statistics	swr::getDomainResourceReports	√
Uploading an image using multipart upload (on the SWR console)	swr:namespace:multipartUpload	√
Pushing an image (docker)	swr:repo:upload	√
Pulling an image (docker)	swr:repo:download	√

NOTE

Both manual and automatic image synchronization require permissions for the actions **swr:repo:download** (pulling images), **swr::createLoginSecret** (obtaining temporary login commands), and **swr:repo:upload** (pushing images).

3 Container Engine Basics

A container engine is one of the most important Kubernetes components. It is used to manage the lifecycle of images and containers. You can use it to create lightweight, portable, and self-sufficient containers for any application easily.

SWR supports two types of container engines: Docker and containerd. This section uses Docker as an example to describe how to install a container engine and use it to create an image file.

Preparations

Before installing Docker, get a basic understanding of what Docker is and how it works. For more information, see **Docker Documentation**.

Selecting a Docker Version

Docker is compatible with almost all operating systems. Select a Docker version that best suits your needs. If you are not sure which version to use, see **https://docs.docker.com/engine/install/**.

NOTE

- You can use SWR to store container images. You are advised to download Docker 18.06 or later.
- Bind an elastic IP address (EIP) first if your server runs in a private network because Docker installation requires Internet connection.

Installing Docker

You can select either of the following installation procedures based on your OS.

Linux

EulerOS

- Linux

  Run the commands to install the latest version. To install a specific version, see **Install Docker Engine**.

  ```
  curl -fsSL get.docker.com -o get-docker.sh
  sh get-docker.sh
  ```



```
sudo systemctl daemon-reload
sudo systemctl restart docker
```

- EulerOS

  Perform the following steps:

  a. Log in to the ECS where you want to install Docker.

  b. Configure a yum repository.

  c. Install and run Docker.

     i. Obtain the **docker-engine** package from the yum repository.

        yum search docker-engine

     ii. Run **yum install -y** to install the **docker-engine** package. The following is an example for x86:

        yum install docker-engine.x86_64 -y

     iii. Enable Docker to start at system startup.

        systemctl enable docker

     iv. Start Docker.

        systemctl start docker

  d. Verify the installation.

     docker --version

     If information similar to the following is displayed, Docker is installed successfully:

     ```
     Docker version 18.09.0, build 384e3e9
     ```

Building a Container Image

This section walks you through the steps of using a Dockerfile to build a container image for a simple web application. Dockerfile is a text file that contains all the instructions a user can call on the command line to build an image. A container image is a stack consisting of multiple layers. Each instruction creates a layer.

When using a browser to access a containerized application built from a Nginx image, you will see the default Nginx welcome page. In this section, you will build a new image from a base Nginx image to change the welcome message to **Hello, SWR!**

Step 1 Log in to the server running the container engine as user **root**.

Step 2 Create a file named **Dockerfile**.

mkdir mynginx

cd mynginx

touch Dockerfile

Step 3 Edit Dockerfile.

vim Dockerfile

Add the following instructions to **Dockerfile**:

```
FROM nginx
RUN echo '<h1>Hello, SWR!</h1>' > /usr/share/nginx/html/index.html
```


In the preceding instructions:

- **FROM**: creates a layer from the base image. A valid Dockerfile must start with a **FROM** instruction. In this example, the **nginx** image is used as the base image.
- **RUN**: executes a command to create a layer. The format is **RUN** *<command>*. In this example, the **echo** command is executed to display "Hello, SWR!"

Press **Esc** and enter **:wq** to save the changes and exit.

Step 4 Run **docker build** *[options] <context path>* to build an image.

docker build -t nginx:v1 .

- **-t nginx:v1**: indicates the image name and tag.
- **.**: indicates the directory where the Dockerfile is located. All content in this directory will be packed and sent to Docker to build an image.

Step 5 Run the command below to view the image. You can see the new image **nginx:v1**.

docker images

----End

Creating an Image Package

This section describes how to compress a container image into a .tar or .tar.gz package.

Step 1 Log in to the server running the container engine as user **root**.

Step 2 Show all images.

docker images

Check the name and tag of the image to be compressed.

Step 3 Compress the image into a package.

docker save *[OPTIONS]* **IMAGE** *[IMAGE...]*

NOTE

OPTIONS: You can set it to **--output** or **-o**, indicating that the image will be exported to a file.

The file should be in either .tar or .tar.gz format.

When using **docker save** to create an image package, use *{image}:{tag}* instead of *image id*. Otherwise, the package cannot be uploaded on the SWR console.

Example:

```
$ docker save nginx:latest > nginx.tar
$ ls -sh nginx.tar
108M nginx.tar

$ docker save php:5-apache > php.tar.gz
$ ls -sh php.tar.gz
372M php.tar.gz

$ docker save --output nginx.tar nginx
$ ls -sh nginx.tar
```



```
108M nginx.tar

$ docker save -o nginx-all.tar nginx # Package the nginx image of all tags.
$ docker save -o nginx-latest.tar nginx:latest
```

----End

Importing an Image File

You can use **docker load** to import an image package.

There are two methods:

docker load < *Path/File name*.tar

docker load --input *Path/File name*.tar or **docker load -i** *Path/File name*.tar

Example:

```
$ docker load --input fedora.tar
```

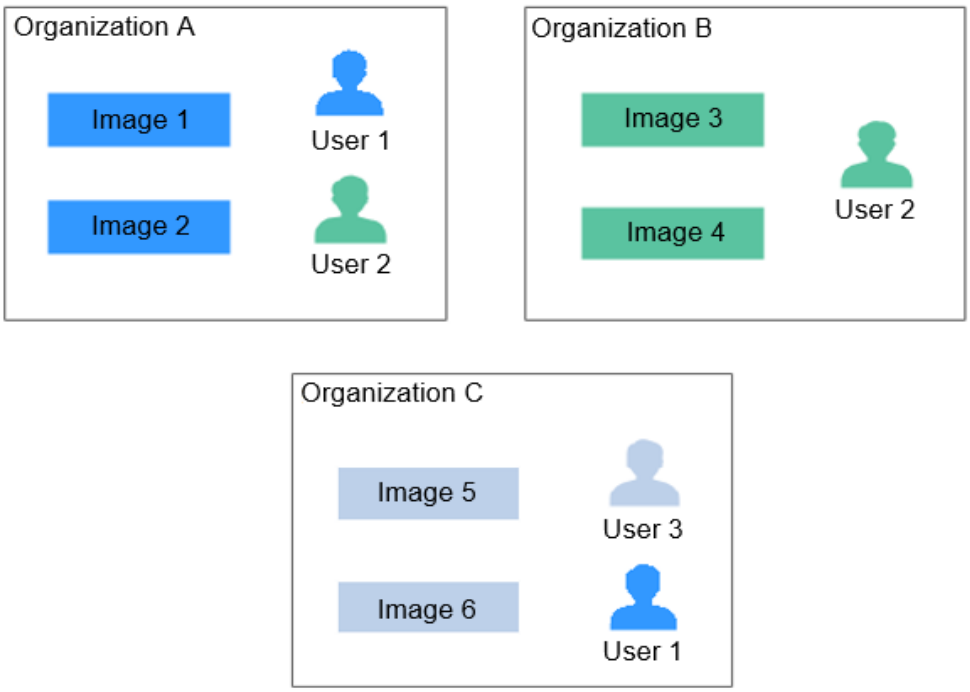

4 Organization Management

Scenarios

Organizations are used to isolate image repositories. With each organization being limited to one company or department, images can be managed in a centralized and efficient manner. An image name needs to be unique within an organization. The same IAM user can access different organizations as long as the user has sufficient permissions, as shown in **Figure 4-1**.

You can grant different permissions, namely, read, edit, and manage, to IAM users under the same account. For details, see **Granting Permissions for Images**.

Figure 4-1 Organizations



Creating an Organization

You can create organizations based on the organizational structure of your enterprise to facilitate image resource management. Create an organization before you push an image.

Step 1 Log in to the **SWR console**.

Step 2 Click in the upper left corner and select a region and project.

Step 3 Choose **Organizations** in the navigation pane.

Step 4 Click **Create Organization** in the upper right corner of the page. In the displayed dialog box, enter a name and click **OK**.

NOTE

- The organization name must be unique in the current region. If a message is displayed indicating that the organization already exists, the organization name may have been used by another user. Use another organization name.
- A user can create organizations only after being assigned with the **SWR Admin** or **Tenant Administrator** policy in IAM.
- The organization name cannot be **library**, which is reserved for the system.

----End

Viewing Images of an Organization

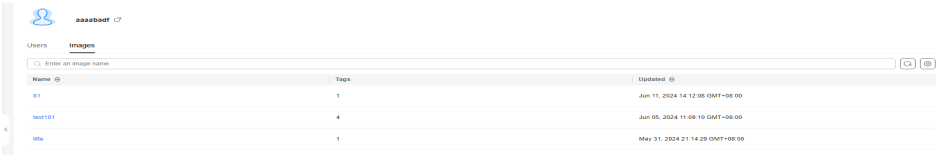
After you create an organization and push images to it, you can view these images in the organization.

Step 1 Log in to the **SWR console**.

Step 2 Click in the upper left corner and select a region and project.

Step 3 In the navigation pane, choose **Organizations**. Then click the name of the target organization in the list.

Step 4 To view the images of this organization, click the **Images** tab.



----End

Deleting an Organization

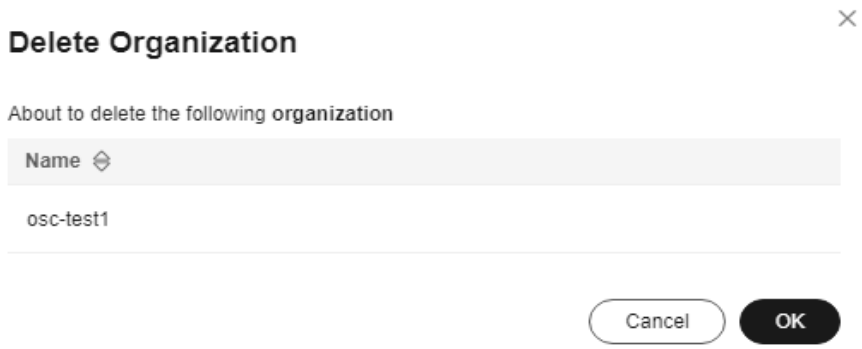
Before deleting an organization, delete all images from the organization.

Step 1 Log in to the **SWR console**.

Step 2 Click in the upper left corner and select a region and project.

Step 3 In the navigation pane, choose **Organizations**.

Step 4 Locate an organization. In the upper right corner of the organization, click **Delete**. Then, click **OK**.



----End

5 Permissions Management

5.1 SWR Permissions Overview

There are three types of SWR permissions:

- **IAM permissions**: Create IAM users and grant them permissions to use SWR.
- **Image permissions**: After creating an IAM administrator, you can grant image access permissions to other IAM users. You can grant these users the permissions to read, edit, or manage images for hierarchical and refined permissions management.
- **Permission dependencies of the SWR console**: Cloud services usually work together to implement a function. SWR triggers and image vulnerability scanning are dependent on HSS, CCE, or CCI. For details, see **Permission Dependencies of the SWR Console**.

5.2 Configuring Permissions in IAM

5.2.1 Creating a User and Granting Permissions

System-defined permissions in role/policy-based authorization provided by **Identity and Access Management (IAM)** let you control access to your SWR resources. With IAM, you can:

- Create IAM users or user groups for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing SWR resources.
- Grant users only the permissions required to perform a given task based on their job responsibilities.
- Entrust other Huawei Cloud account or cloud service to perform efficient O&M on your SWR resources.

If your Huawei Cloud account does not require individual IAM users for permissions management, you can skip this section.

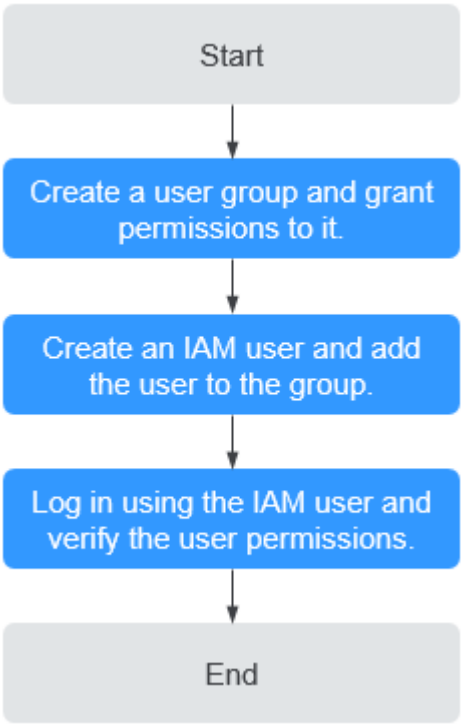
This section describes the procedure for granting user permissions.

Prerequisites

Before granting permissions to user groups, learn about system-defined permissions for SWR. For details, see **Permissions**. To grant permissions for other services, learn about all **system-defined permissions**.

Process Flow

Figure 5-1 Process of granting SWR permissions



1. **Create a user group and assign permissions** to it.

   Create a user group on the IAM console, and grant the **SWR Admin** permissions to the group.

2. **Create an IAM user and add it to the user group**.

   Create a user on the IAM console and add the user to the group created in **1**.

3. **Log in** and verify permissions.

   Log in to the management console using the IAM user. Switch to the authorized region. If the following operations can be performed, the permissions are assigned successfully:

   a. Choose **Service List** > **SoftWare Repository for Container**. The SWR console is displayed.

   b. In the navigation pane, choose **Organizations**. Click **Create Organization** in the upper right corner. Enter an organization name to create an organization.

   c. In the navigation pane, choose **My Images**. Click **Upload Through SWR** in the upper right corner. Select the new organization. Upload a local image file to SWR.

System-defined Roles

Roles are a type of coarse-grained authorization mechanism that defines service-level permissions based on user responsibilities. Only a limited number of service-level roles are available for authorization. Roles are not ideal for fine-grained authorization and least privilege access.

The system-defined role preset for SWR in IAM is **SWR Admin**, which has all permissions for SWR.

System-defined Policies

System-defined policies preset for SWR in IAM include **SWR FullAccess**, **SWR OperateAccess**, and **SWR ReadOnlyAccess**.

- **SWR FullAccess**: all permissions for SWR
- **SWR OperateAccess**: operation permissions for SWR
- **SWR ReadOnlyAccess**: read-only permissions for SWR

Table 5-1 Permissions granted by the SWR FullAccess policy

Action	Specific Action	Description
"swr:*:*"	"vpc:*:get*"	Permission to view details about all VPC resources
	"vpc:*:list*"	Permission to list all VPC resources
	"scm:*:list*"	Permission to list all SSL Certificate Manager (SCM) resources
	"scm:cert:download"	Permission to download SSL certificates from SCM
	"kms:*:get*"	Permission to view details about all KMS resources
	"kms:*:list*"	Permission to list all KMS resources
	"hss:image:*"	All permissions for container images in Host Security Service (HSS)
	"cce:cluster:list*"	Permission to list all CCE resources
	"cci:namespace:list*"	Permission to list all CCI namespace resources

Action	Specific Action	Description
	"cci:deployment:list*"	Permission to list all CCI Deployment resources
	"cci:namespaceSubResource:list*"	Permission to list all CCI Kubernetes resources
	"cci:deployment:get"	Permission to view details about all CCI Deployment resources
	"cci:namespaceSubResource:get"	Permission to view details about all CCI Kubernetes resources

Table 5-2 Permissions granted by the SWR ReadOnlyAccess policy

Action	Specific Action	Description
"swr:*:*"	"swr:*:get*"	Permission to view details about all SWR resources
	"swr:*:list*"	Permission to list all SWR resources
	"swr:*:download*"	Permission to download images from SWR
	"swr:instance:createTempCredential"	Permission to generate temporary login credentials for a repository of SWR Enterprise Edition
	"swr:system:createLoginSecret"	Permission to generate temporary login credentials for shared repositories in SWR
	"vpc:*:get*"	Permission to view details about all VPC resources
	"vpc:*:list*"	Permission to list all VPC resources
	"scm:*:list*"	Permission to list all SCM resources
	"kms:*:get*"	Permission to view details about all KMS resources

Action	Specific Action	Description
	"kms:*:list*"	Permission to list all KMS resources
	"hss:image:list*"	Permission to list container images in HSS
	"hss:image:vulnerabilities"	Permission to list container image vulnerabilities in HSS
	"cce:cluster:list*"	Permission to list all CCE resources
	"cci:namespace:list*"	Permission to list all CCI namespace resources
	"cci:deployment:list*"	Permission to list all CCI Deployment resources.
	"cci:namespaceSubResource:list*"	Permission to list all CCI Kubernetes resources
	"cci:deployment:get"	Permission to view details about all CCI Deployment resources
	"cci:namespaceSubResource:get"	Permission to view details about all CCI Kubernetes resources

Table 5-3 Permissions granted by the SWR OperateAccess policy

Action	Specific Action	Description
"swr:*:*"	"swr:repository:*"	All permissions for managing repositories of SWR Enterprise Edition
	"swr:instance:get*"	Permission to view details about repositories of SWR Enterprise Edition
	"swr:instance:list*"	Permission to list repositories of SWR Enterprise Edition
	"swr:instance:execute*"	Permission to execute asynchronous tasks of repositories of SWR Enterprise Edition

Action	Specific Action	Description
	"swr:instance:createTempCredential"	Permission to generate temporary login credentials for a repository of SWR Enterprise Edition
	"swr:system:createLoginSecret"	Permission to generate temporary login credentials for shared repositories in SWR
	"swr:repo:*"	All permissions for repositories of SWR Basic Edition
	"swr:namespace:get*"	Permission to view all namespace resources of SWR Basic Edition
	"swr:namespace:list*"	Permission to list namespace resources of SWR Basic Edition
	"swr:system:listQuotas"	Permission to view the quota information of SWR Basic Edition
	"swr:system:getDomainOverview"	Permission to view the brief resource information of SWR Basic Edition
	"swr:system:getDomainResourceReports"	Permission to obtain tenant resource statistics of repositories of SWR Basic Edition
	"vpc:*:get*"	Permission to view details about all VPC resources
	"vpc:*:list*"	Permission to list all VPC resources
	"scm:*:list*"	Permission to list all SCM resources
	"kms:*:get*"	Permission to view details about all KMS resources
	"kms:*:list*"	Permission to list all KMS resources

Action	Specific Action	Description
	"hss:image:*"	All permissions for container images in HSS
	"cce:cluster:list*"	Permission to list all CCE resources

Table 5-4 Permissions granted by the SWRFullAccessPolicy policy

Action	Specific Action	Description
"swr:*:*"	"vpc:*:get*"	Permission to view details about all VPC resources
	"vpc:*:list*"	Permission to list all VPC resources
	"scm:cert:list"	Permission to list all SCM certificates
	"kms:cmk:get"	Permission to view details about KMS keys
	"kms:cmk:list"	Permission to list all KMS keys
	"eps:enterpriseProjects:list"	Permission to list all enterprise projects
	"iam:projects:list"	Permission to list all IAM projects

Table 5-5 Permissions granted by the SWRReadOnlyAccessPolicy policy

Action	Specific Action	Description
"swr:*:*"	"swr:*:get*"	Permission to view details about all SWR resources
	"swr:*:list*"	Permission to list all SWR resources
	"swr:*:download*"	Permission to download the list of all SWR resources

Action	Specific Action	Description
	"swr:instance:createTempCredential"	Permission to generate temporary login credentials for a repository of SWR Enterprise Edition
	"vpc:*:get*"	Permission to view details about all VPC resources
	"vpc:*:list*"	Permission to list all VPC resources
	"scm:cert:list"	Permission to list all SCM certificates
	"kms:cmk:get"	Permission to view details about KMS keys
	"kms:cmk:list"	Permission to list all KMS keys
	"eps:enterpriseProjects:list"	Permission to list all enterprise projects
	"iam:projects:list"	Permission to list all IAM projects

Table 5-6 Permissions granted by the SWROperateAccessPolicy policy

Action	Specific Action	Description
"swr:*:*"	"swr:*:get*"	Permission to view details about all SWR resources
	"swr:*:list*"	Permission to list all SWR resources
	"swr:*:download*"	Permission to download the list of all SWR resources
	"swr:instance:createTempCredential"	Permission to generate temporary login credentials for a repository of SWR Enterprise Edition
	"swr:repository:*"	All permissions for managing repositories of SWR Enterprise Edition

Action	Specific Action	Description
	"swr:instance:execute*"	Permission to execute asynchronous tasks of repositories of SWR Enterprise Edition
	"vpc:*:get*"	Permission to view details about all VPC resources
	"vpc:*:list*"	Permission to list all VPC resources
	"scm:cert:list"	Permission to list all SCM certificates
	"kms:cmk:get"	Permission to view details about KMS keys
	"kms:cmk:list"	Permission to list all KMS keys
	"eps:enterpriseProjects:list"	Permission to list all enterprise projects
	"iam:projects:list	Permission to list all IAM projects

Custom Policies

Custom policies can be created as a supplement to the system-defined policies of SWR. For the actions that can be added to custom policies, see **Permissions Policies and Supported Actions**.

You can create custom policies in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Create a JSON policy or edit an existing one.

For details, see **Creating a Custom Policy**.

5.2.2 SWR Custom Policies

Custom policies can be created to supplement the system-defined policies of SWR. For details about actions supported in custom policies, see **Permissions and Supported Actions**.

You can create custom policies in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Create a policy in the JSON format from scratch or based on an existing policy.

For details about how to create a custom policy, see **Creating a Custom Policy**. This section provides examples of common custom policies of SWR.

Example SWR Custom Policies

- Example 1: Allowing image pull

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "swr:repo:download"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "swr::createLoginSecret"
      ]
    }
  ]
}
```

- Example 2: Granting the permission to deny image pull

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "swr:repo:download"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "swr::createLoginSecret"
      ]
    }
  ]
}
```

Example 3: Allowing image pull over a specified source VPC

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "swr:repo:download"
      ],
      "Condition": {
        "StringEquals": {
          "g:SourceVpc": [
            "0bfdf87b-7789-4851-801e-8e726b82beae"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "swr::createLoginSecret"
      ]
    }
  ]
}
```


5.2.3 SWR Resources

A resource is an object that exists within a service. SWR resources include repo and namespace. You can select these resources by specifying their paths.

Table 5-7 SWR resources and their paths

Resource	Resource Name	Path
repo	Image repository	[Format] swr:*:*:repo:*image repository name* The first * is **regionid**, and the second * is **domainid**. [Note] IAM automatically generates the path prefix **SWR:*:*:repo:**. For the path of a specific repository, add the *image repository name* to the end. You can also use a wildcard character (*) to indicate any image repository. Example: swr:*:*:repo:test/nginx*: image repository whose name starts with **nginx** in the **test** organization swr:*:*:repo:test/nginx: image repository whose name is **nginx** in the **tes**t organization

Resource	Resource Name	Path
namespace	Organization	[Format] SWR:*:*:namespace:*organization name* The first * is **regionid**, and the second * is **domainid**. [Note] For the path of an organization, IAM automatically generates the resource path prefix SWR:*:*:namespace:. You can add the organization name to the end. You can also use a wildcard character (*) to indicate any organization. Example: swr:*:*:namespace:test*: organization whose name starts with **test** swr:*:*:namespace:test: organization whose name is **test**

5.3 Configuring Permissions in SWR

5.3.1 Granting Permissions for Images

Scenarios

To manage SWR permissions, you can use Identity and Access Management (IAM). For details, see **Creating a User and Granting Permissions**. If you have the SWR Admin or Tenant Administrator permissions, you become an admin user of SWR and can grant permissions to other IAM users in SWR. To push an image, you must have the edit or manage permission. To pull a private image, you must have the read, edit, or manage permission. To pull a public image, no permission is required.

NOTE

An admin user is granted image management permission of all organizations by default, even if the user is not in the authorized user list of any organization.

If you are not an SWR admin user, you can request an SWR admin user to grant you permissions to read, edit, or manage a specific image or images in a specific organization.

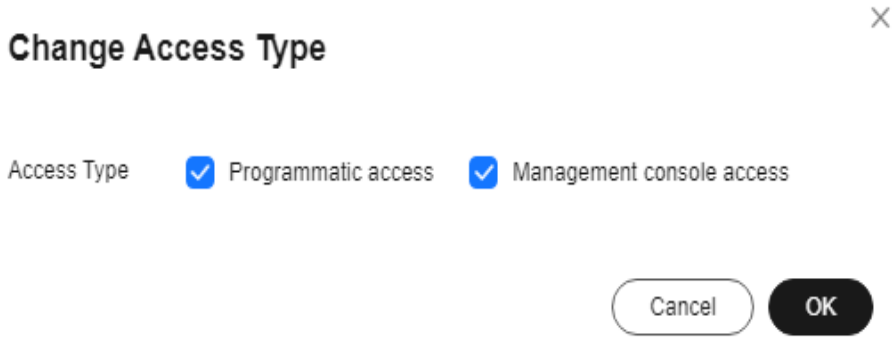
Examples

- Example 1: An IAM user with the ServiceStage Developer permission (SWR read-only permission) wants to pull the **nginx** image created by the SWR administrator in the **group** organization.

  Solution: On the details page of the **nginx** image, the SWR administrator grants the **read** permission to the IAM user.
- Example 2: An SWR administrator wants to grant an external user the permission to push images to the organization, but the user is not allowed to log in to the console and can only push images through the container engine client.

  Solution: On the **Users** tab of the details page of the organization, the SWR administrator grants the **edit** permission to the user. In IAM, the administrator sets **Access Type** to **Programmatic access**.

  Figure 5-2 Changing the access type



Constraints

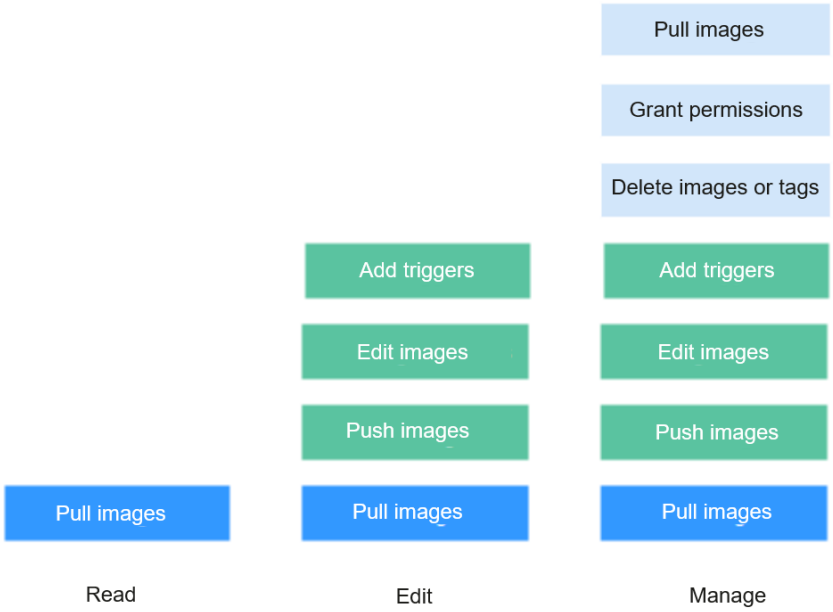
Currently, federated users cannot manage image authorization on the SWR console. You can add a custom policy on the IAM console to manage image authorization. For details, see **SWR Custom Policies**.

Authorization Methods

In SWR, you can grant permissions to IAM users in either of the following ways:

- **Grant permissions for a specific image** to allow IAM users to read, edit, and manage the image.
- **Grant permissions for an organization** to allow IAM users to read, edit, and manage all the images in the organization.

Figure 5-3 User permissions



You can add the following three types of permissions to users:

- Read: Users can only pull images.
- Edit: Users can pull and push images, edit images, and add triggers.
- Manage: Users can pull and push images, delete images or tags, edit images, grant permissions, add triggers, and share images with other users.

NOTE

To upload images to an organization on the SWR console, users need to have permission to edit or manage the organization. The edit and manage permissions granted on image details pages are not sufficient to upload images.

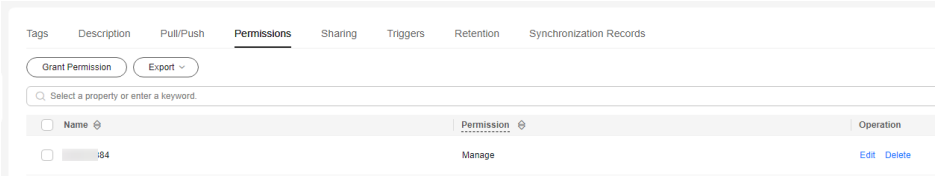
Granting Permissions for a Specific Image

To allow IAM users of your account to read, edit, and manage a specific image, grant the required permissions to the users on the details page of this image.

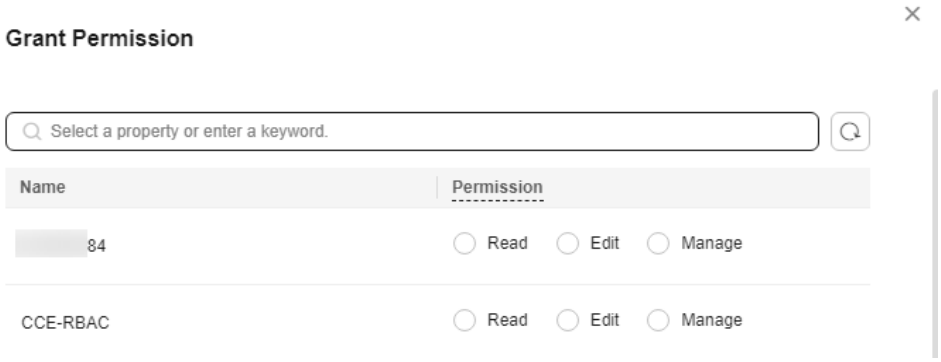
Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then click the name of the target image.

Step 3 On the image details page, click the **Permissions** tab.



Step 4 Click **Grant Permission**. In the displayed dialog box, enter an IAM username, and then select **Read**, **Edit**, or **Manage**. Click **OK**.

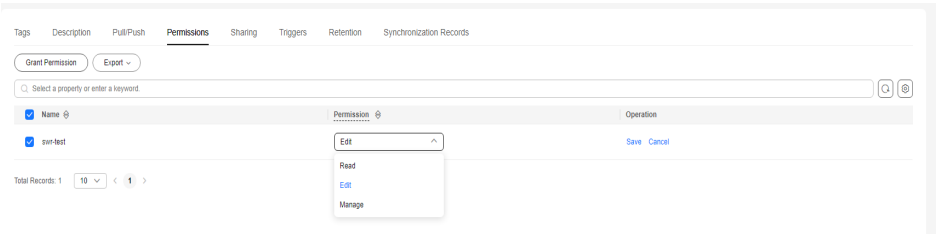


----End

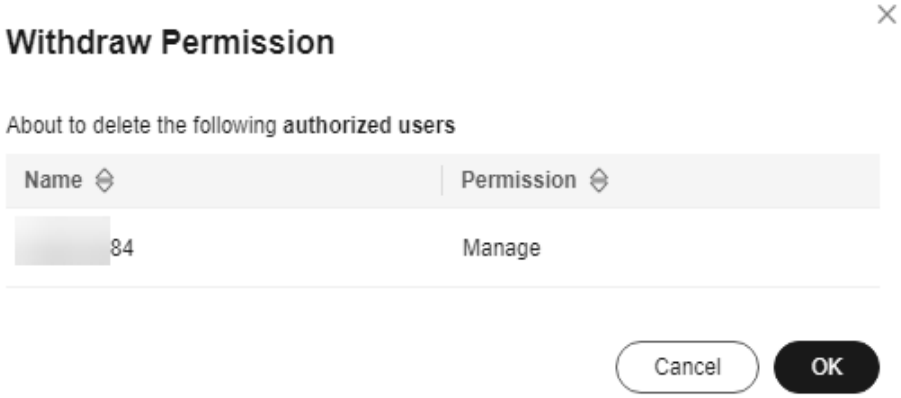
Modifying or Deleting Permissions for a Specific Image

You can modify or delete user permissions on the details page of an image.

- To modify permissions, click the **Permissions** tab on the details page of an image. Locate a user and click **Edit** in the **Operation** column. Select permission in the **Permission** drop-down list and click **Save**.



- To delete permissions, click the **Permissions** tab on the details page of an image. Locate a user and click **Delete** in the **Operation** column. Click **OK**.



Granting Permissions for an Organization

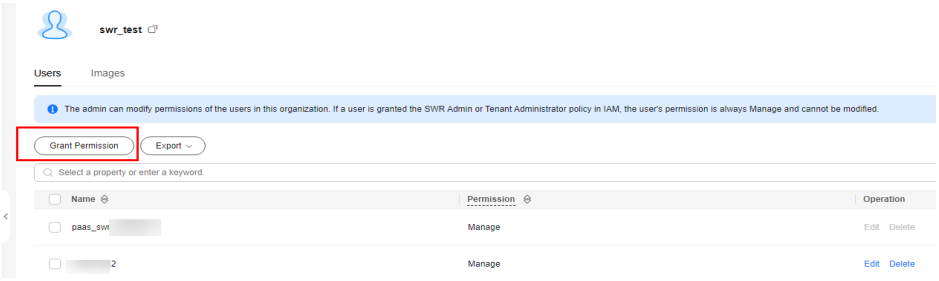
After an IAM user is created, the administrator needs to grant this user the permissions for an organization so that this user can read, edit, and manage images in the organization.

Only accounts and IAM users who have the **Manage** permission can grant permissions to other users.

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **Organizations**. Locate the target organization and click its name.

Step 3 On the **Users** tab, click **Grant Permission**. In the displayed dialog box, enter an IAM username, and then select permission for the user. Click **OK**.

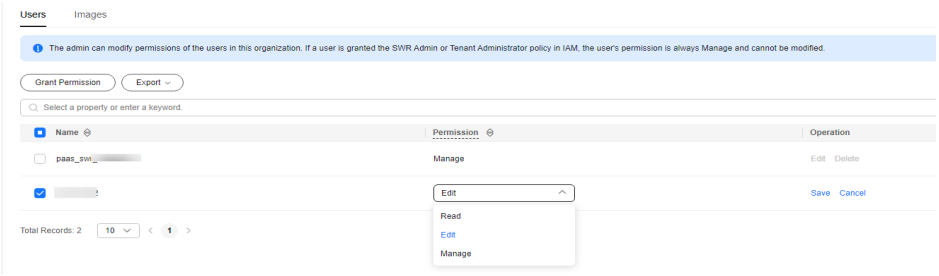


----End

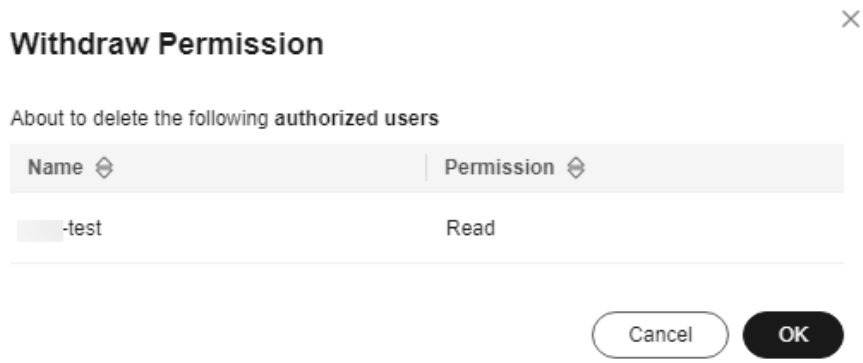
Modifying or Deleting Permissions for an Organization

You can modify or delete user permissions for an organization.

- To modify permissions, on the **Users** tab, locate a user and click **Edit** in the **Operation** column. Select permission in the **Permission** drop-down list and click **Save**.



- To delete permissions, on the **Users** tab, locate a user and click **Delete** in the **Operation** column. Click **OK**.



5.4 Permission Dependencies of the SWR Console

Some SWR functions depend on other cloud services. You can use IAM to grant permission to access these services.

Before granting an IAM user permission to access dependent cloud services of SWR, ensure that this user already has SWR Administrator, SWR FullAccess, SWR OperateAccess, or SWR ReadOnlyAccess permissions as needed.

Granting Permissions to Access Dependent Cloud Services

To use basic SWR functions, an IAM user must already have SWR Administrator, SWR FullAccess, SWR OperateAccess, or SWR ReadOnlyAccess permissions. Then, you can add roles or policies for them to use the SWR functions that depend on other cloud services.

Table 5-8 Role/Policy dependencies of the SWR console

Console Function	Dependency	Role/Policy Required
Image vulnerability scanning	Host Security Service (HSS)	Custom policy: **SWR HSS Access**
Triggers	Cloud Container Engine (CCE) Cloud Container Instance (CCI)	If your applications are deployed in CCE, you need policies to access CCE. For details, see the custom policy **SWR CCE Access**. If your applications are deployed in CCI, you need policies to access CCI. For details, see the custom policy **SWR CCI Access**.

NOTE

To grant an IAM user permission to access dependent cloud services of SWR, you must have the IAM role Security Administrator.

Fine-grained HSS Authorization

Step 1 Log in to the **management console**.

Step 2 Select a region, click ☰ in the upper left corner, and choose **Management & Governance** > **Identity and Access Management**.

Step 3 In the navigation pane, choose **Permissions** > **Policies/Roles**. Click **Create Custom Policy**. Set **Policy Name** to **SWR HSS Access** and **Policy View** to **JSON**. Configure the policy as follows and click **OK**.

```
{
  "Version": "1.1",
```



```
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "HSS:*:*"
      ]
    }
  ]
}
```

Step 4 In the navigation pane, choose **User Groups**. Select the user group the IAM user belongs to and click **Authorize**.

Step 5 Select the **SWR HSS Access** policy. Select **All resources** and click **OK**.

Step 6 After the authorization is successful, click **Finish**. The policy will be in effect after about 15 minutes.

----End

Fine-grained CCE Authorization

Step 1 Log in to the **management console**.

Step 2 Select a region, click ☰ in the upper left corner, and choose **Management & Governance** > **Identity and Access Management**.

Step 3 In the navigation pane, choose **Permissions** > **Policies/Roles**. Click **Create Custom Policy**. Set **Policy Name** to **SWR CCE Access** and **Policy View** to **JSON**. Configure the policy as follows and click **OK**.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cce:cluster:list"
      ]
    }
  ]
}
```

Step 4 In the navigation pane, choose **User Groups**. Select the user group your IAM user belongs to and click **Authorize**.

Step 5 Select the **SWR CCE Access** policy. Select **All resources** and click **OK**.

Step 6 After the authorization is successful, click **Finish**. The policy will be in effect after about 15 minutes.

Step 7 Click ☰ in the upper left corner. Choose **Containers** > **Cloud Container Engine**. In the navigation pane, choose **Permissions**. Select the cluster to access. In the upper right corner of the page, click **Add Permission**.

Step 8 Configure the following parameters and click **OK**.

- **User/User Group**: Select the user group the IAM user belongs to.
- **Namespace**: Select **All namespaces**.
- **Permission Type**: Select **viewer**.

Step 9 When a dialog box is displayed indicating the permission is added successfully, click **OK**. Wait for 3 to 5 seconds for the authorization to take effect.

----End

Fine-grained CCI Authorization

Step 1 Log in to the **management console**.

Step 2 Select a region, click ☰ in the upper left corner, and choose **Management & Governance** > **Identity and Access Management**.

Step 3 In the navigation pane, choose **Permissions** > **Policies/Roles**. Click **Create Custom Policy**. Set **Policy Name** to **SWR CCI Access** and **Policy View** to **JSON**. Configure the policy as follows and click **OK**.

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
           "cci:namespace:list",
           "cci:deployment:list",
           "cci:namespaceSubResource:list",
           "cci:deployment:get",
           "cci:namespaceSubResource:get"
         ]
      }
   ]
}
```

Step 4 In the navigation pane, choose **User Groups**. Select the user group your IAM user belongs to and click **Authorize**.

Step 5 Select the **SWR CCI Access** policy. Select **All resources** and click **OK**.

Step 6 After the authorization is successful, click **Finish**. The policy will be in effect after about 15 minutes.

----End

5.5 Control Policies Supported by SWR

5.5.1 Overview of Control Policies

SWR supports multiple control policies, including IAM-based access control, SCP-based access control, RCP-based access control, NCP-based access control, and VPC Endpoint policy-based access control. You can use different control policies based on security requirements.

IAM-based Access Control

Identity and Access Management (IAM)) provides permissions management for secure access to your Huawei Cloud services and resources. For details about how to use IAM to control access to SWR, see **Configuring Permissions in IAM**.

SCP-based Access Control

Service Control Policies (SCPs) are guardrail policies provided by Organizations. The management account can use SCP to limit the permissions that can be assigned to member accounts in an organization. You can attach an SCP to your organization, OUs, or member accounts. Any SCP attached to an organization or OU affects all the accounts within the organization or under the OU. For details, see **Service Control Policies Overview**.

NOTE

The organization here refers to the organization in the Organizations service, not the organization in SWR.

RCP-based Access Control

Resource Control Policies (RCPs) are guardrail policies provided by Organizations. RCPs limit the maximum permissions allowed for a resource. Access to resources of an organization member account is restricted by RCPs. An organization administrator can set RCPs in an organization to meet the security and compliance requirements for access control of resources in organization member accounts.

NOTE

The organization here refers to the organization in the Organizations service, not the organization in SWR.

NCP-based Access Control

Network Control Policies (NCPs) are guardrail policies provided by Organizations. An NCP policy limits the maximum permissions allowed for access from a VPC endpoint. NCP policies restrict requests initiated from a VPC endpoint created by the member accounts of an organization. An organization administrator can set NCPs in an organization to meet the security and compliance requirements for controlling the access initiated from the VPC endpoints created by member accounts of an organization.

NOTE

The organization here refers to the organization in the Organizations service, not the organization in SWR.

VPC Endpoint Policy-based Access Control

VPC endpoint policies are a type of resource-based policies. You can configure a policy to control which principals can use the VPC endpoint to access VPC endpoint services. For details, see **Managing Policies for VPC Endpoints**.

Virtual Private Cloud (VPC) is used to control the network border security. If the API access point of a resource is within the VPC of your account, the access is within the VPC and the security is controllable (the VPC can be considered as a network security domain). If the API access point is on a public network, the network attack surface is large and security is hard to control.

NOTE

After a control policy is configured, anonymous download of public images is also controlled by the control policy.

5.5.2 SCPs

This section uses an example to describe how to configure SCPs.

Example: **Forbid an account to download images in an organization.**

The following describes how to configure an SCP to forbid an account in an organization in the Organizations service to download images from the **test-repo** image repository in the **test-namespace** organization of SWR.

Configuration method

Step 1 Log in to the management console as the organization administrator or using the management account, and navigate to the Organizations console.

Step 2 On the **Policies** page, click **Service control policies** and then **Create Policy**.

Step 3 Enter the policy name and description. On the left of the policy content, you can copy and paste the following JSON policy content: Click **Save**.

```
{
  "Version": "5.0",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "swr:repo:download"
      ],
      "Resource": [
        "swr:*:*:repo:test-namespace/test-repo"
      ]
    }
  ]
}
```

Step 4 Bind the policy to an OU or account of the organization to apply the policy.

1. Log in to Huawei Cloud as the organization administrator or using the management account, navigate to the Organizations console, and access the **Organization** page.
2. Select the OU or account you want to attach the SCP to.
3. On the details page, click the **Policies** tab. On the displayed tab, expand **Service control policies** and click **Attach**.
4. Select the policy to be added and enter "Confirm" in the text box. Then, click **Attach**.

----End

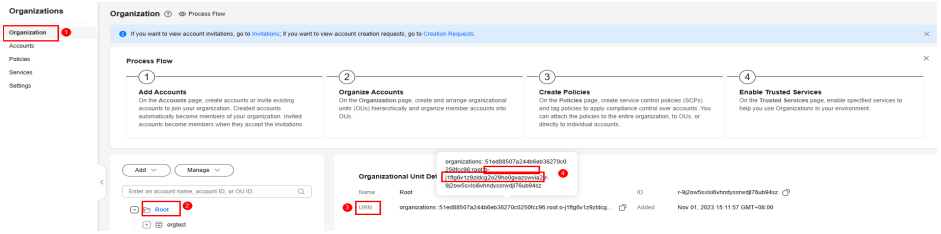
5.5.3 RCPs

Example 1: **Images in an organization can only be downloaded by accounts in that organization.**

The following policy defines that images in the OU or account bound to the policy cannot be downloaded by accounts outside the **o-j1ftg6v1z9zldcg2o29ho0gvazswvia2** organization. They can only be downloaded by accounts in the organization.

NOTE

The organization here refers to the organization in the Organizations service, not the organization in SWR. To obtain the ID of the organization, follow the steps in the figure.



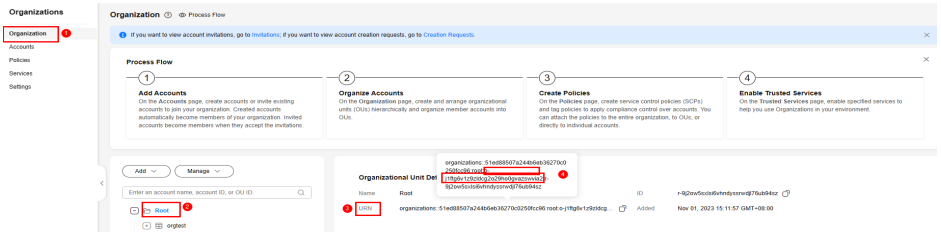
```
{
  "Version": "5.0",
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": [
        "swr:repo:download"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "StringNotEquals": {
          "g:PrincipalOrgId": [
            "o-j1ftg6v1z9zldcg2o29ho0gvazswvia2"
          ]
        },
        "Bool": {
          "g:PrincipalIsService": [
            "false"
          ]
        }
      }
    }
  ]
}
```

Example 2: **Images in an organization can only be downloaded by accounts in that organization, except public images.**

The following policy defines that private images in the OU or account bound to the policy cannot be downloaded by accounts outside the **o-j1ftg6v1z9zldcg2o29ho0gvazswvia2** organization. They can only be downloaded by accounts in the organization. Public images can be downloaded by any account.

NOTE

The organization here refers to the organization in the Organizations service, not the organization in SWR. To obtain the ID of the organization, follow the steps in the figure.



```
{
  "Version": "5.0",
```



```
"Statement": [
  {
    "Effect": "Deny",
    "Principal": "*",
    "Action": [
      "swr:repo:download"
    ],
    "Resource": [
      "*"
    ],
    "Condition": {
      "StringNotEquals": {
        "g:PrincipalOrgId": [
          "o-j1ftg6v1z9zldcg2o29ho0gvazswvia2"
        ]
      },
      "Bool": {
        "g:PrincipalIsService": [
          "false"
        ],
        "swr:RepositoryIsPublic": [
          "false"
        ]
      }
    }
  }
]
}
```

NOTE

The configuration method is the same as that described in **SCPs**.

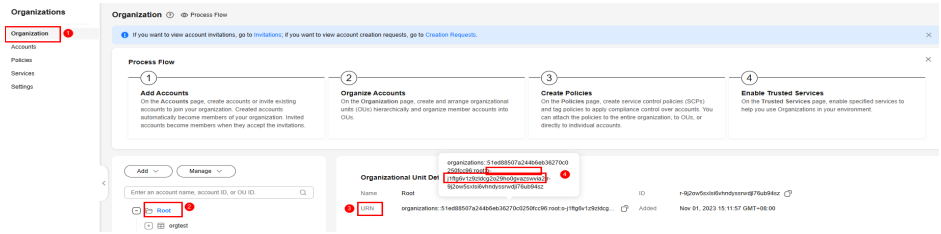
5.5.4 NCPs

Example 1: **Accounts in an organization can only download private images in that organization through VPC Endpoint, and they can download any public images.**

The following policy defines that images in the OU or account bound to the policy cannot be downloaded by accounts outside the **o-j1ftg6v1z9zldcg2o29ho0gvazswvia2** organization through VPC Endpoint. They can only be downloaded by accounts in the organization.

NOTE

The organization here refers to the organization in the Organizations service, not the organization in SWR. To obtain the ID of the organization, follow the steps in the figure.




```
{
  "Version": "5.0",
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": [
        "swr:repo:download"
```



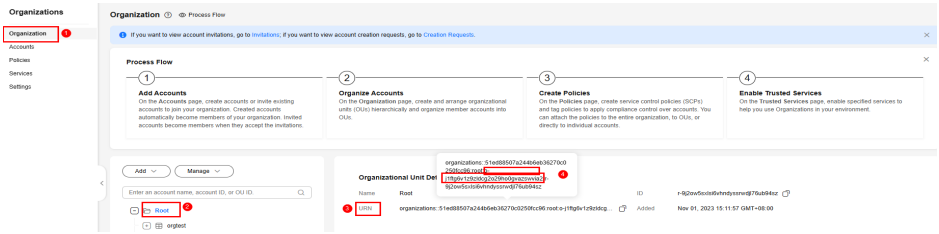
```
    ],
    "Resource": [
      "*"
    ],
    "Condition": {
      "Bool": {
        "g:PrincipalIsService": [
          "false"
        ]
      },
      "StringNotEquals": {
        "g:ResourceOrgId": [
          "o-j1ftg6v1z9zldcg2o29ho0gvazswvia2"
        ]
      }
    }
  }
 ]
}
```

Example 2: **Accounts in an organization can only download private images in that organization through VPC Endpoint, and they can download any public images.**

The following policy defines that private images in the OU or account bound to the policy cannot be downloaded by accounts outside the **o-j1ftg6v1z9zldcg2o29ho0gvazswvia2** organization through VPC Endpoint. They can only be downloaded by accounts in the organization. Public images can be downloaded by any account.

NOTE

The organization here refers to the organization in the Organizations service, not the organization in SWR. To obtain the ID of the organization, follow the steps in the figure.




```
{
 "Version": "5.0",
 "Statement": [
  {
    "Effect": "Deny",
    "Principal": "*",
    "Action": [
      "swr:repo:download"
    ],
    "Resource": [
      "*"
    ],
    "Condition": {
      "Bool": {
        "g:PrincipalIsService": [
          "false"
        ],
        "swr:RepositoryIsPublic": [
          "false"
        ]
      },
      "StringNotEquals": {
        "g:ResourceOrgId": [
```

```
          "o-j1ftg6v1z9zldcg2o29ho0gvazswvia2"
        ]
      }
    }
  }
 ]
}
```

NOTE

The configuration method is the same as that described in **SCPs**.

5.5.5 VPC Endpoint Policies

In SWR Basic Edition, you can upload and download images through VPC endpoints. The VPC endpoint policy can be configured to control the upload and download of the images. For details about how to create a VPC endpoint, see **Access Through VPC Endpoint**. For details about how to configure a VPC endpoint policy, see **Managing the Policy of a VPC Endpoint**.

Example 1: **Configure a VPC endpoint policy to allow the upload or download of only specified images.**

The following policy only allows the servers in VPC1 to upload images to or download images from the **test-repo** repository in the **test-namespace** organization of SWR.

```
{
 "Version": "5.0",
 "Statement": [
  {
   "Action": [
    "swr:repo:upload",
    "swr:repo:download"
   ],
   "Resource": [
    "swr:*:*:repo:test-namespace/test-repo"
   ],
   "Effect": "Allow",
   "Principal": "*"
  }
 ]
}
```

Example 2: **Configure a VPC endpoint policy to allow the download of only specified private images and all public images.**

The following policy only allows the servers in VPC1 to download images from the **test-repo** repository in the **test-namespace** organization of SWR. Public images are not restricted.

```
{
 "Version": "5.0",
 "Statement": [
  {
   "Action": [
    "swr:repo:download"
   ],
   "Resource": [
    "swr:*:*:repo:test-namespace/test-repo"
   ],
   "Effect": "Allow",
   "Principal": "*"
  },
```



```
  {
    "Action": [
      "swr:repo:download"
    ],
    "Resource": [
      "*"
    ],
    "Effect": "Allow",
    "Principal": "*",
    "Condition": {
      "Bool": {
        "swr:RepositoryIsPublic": [
          "true"
        ]
      }
    }
  }
 ]
}
```


6 Image Management

6.1 Image Management Overview

SWR provides free, shared image repositories. These repositories offer full-lifecycle image management for your images.

- **Pushing images**: Pushing images (also called uploading images) helps you push local images to an SWR image repository, so that you can manage images more conveniently. You can use either a container engine client or the SWR console to push your images. Currently, there are two types of container engine clients: **Docker** and **containerd**.
- **Pulling images**: Pulling images (also called downloading images) is the process of obtaining images from an image repository. Then, you can use the images to deploy containerized applications in CCE or CCI.
- **Logging in to or connecting to SWR**: Before pushing or pulling images using a container engine client, you need to connect to SWR. You can use a temporary or long-term command to connect to SWR.
- **Modifying images**: After images are uploaded, they are private by default. You can modify image attributes, including image type (public or private), category, and description.
- **Sharing images**: After images are uploaded, you can share **private images** with other accounts and grant them the permission to download the shared images.
- **Adding image triggers**: SWR often works with CCE or CCI to enable automatic application updates. You can add a trigger to automatically update the application that uses the image when the image tag is updated.
- **Adding image retention policies**: After images are pushed, you can add retention policies to automatically delete any unused images. There are policies based on the number of image retention days and policies based on the number of image tags.
- **Synchronizing images**: After images are uploaded, you can synchronize images of the latest tags to the image repository in another region. Both manual synchronization and automatic synchronization are supported.

- **Scanning images**: You can scan your private images to check for vulnerabilities in just a few clicks and follow the suggestions to keep images safe.
- **Image Center**: SWR provides a large number of public images. You can add public container images to your favorites and push them to your repository.

6.2 Pushing an Image

Scenarios

You can use either a container engine client or the SWR console to push your images to SWR for easier management.

- Method 1: Run **docker push** (Docker) or **ctr push** (containerd) on the server where the container engine client is installed to push an image to SWR. This is applicable to the push of large images.

  NOTE

  The push can be implemented over an intranet or the Internet. For details, see **Configuring Access Network**.
- Method 2: Upload an image on the SWR console. This is applicable to the push of small images.

Prerequisites

- An organization has been created in SWR. For details, see **Creating an Organization**.
- A container engine client is available.
- The image has been saved as a **.tar** or **.tar.gz** file. For details, see **Creating an Image Package**.

Constraints

- If a Docker container engine client is used to push images, the Docker version is 18.06 or later.
- The size of each image layer cannot exceed 10 GB.
- A single tenant can push up to 20 image layers concurrently.
- If you use the console, up to 10 files can be uploaded at a time. The size of a single file (including each decompressed file) cannot exceed 2 GB. If the image package to be uploaded is created using a Docker container engine client, the Docker version must be 18.06 or later.
- A single tenant can push up to 500 images and 300 tags for each image. If the quotas are exceeded, the push will fail.

Methods

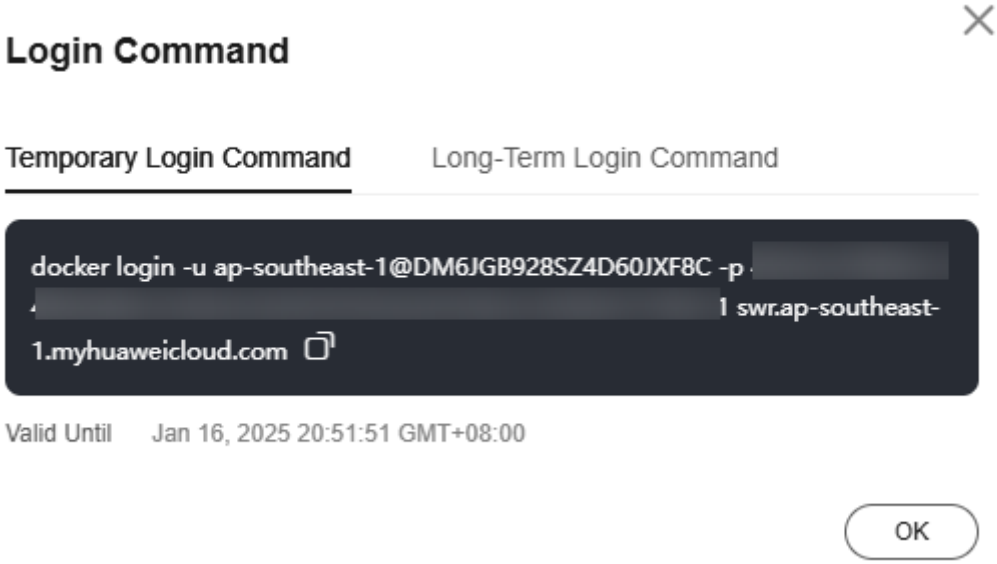
- For small images, upload them on the console. Resumable upload is not supported. For large images, push them using a container engine client.
- SWR has a quota on the number of images that can be stored. You can delete unused images in a timely manner.

(Recommended) Pushing an Image Using a Container Engine Client

Docker

1. **Build a container image** or **import an image file**.
2. Connect to SWR.
   a. Log in to the **SWR console**.
   b. In the navigation pane, choose **Dashboard**. Click **Generate Login Command** in the upper right corner. In the displayed dialog box, click to copy the login command.

      Figure 6-1 Generating a login command

      NOTE

      - A temporary login command is valid for 6 hours. For details about how to obtain a login command that will remain valid for a long term, see **Obtaining a Long-Term Login or Image Push/Pull Command**. After you obtain a long-term login command, your temporary login commands will still be valid as long as they are in their validity periods.
      - After a temporary login command expires, clear the browser cache before you generate a new one.
      - The domain name at the end of the login command is the image registry address. Record the address for later use.

   c. Run the login command on the server where Docker is installed.

      The message **Login Succeeded** will be displayed upon a successful login.
3. Run the following command on the server running Docker to tag the **nginx** image:

   docker tag *[image-name 1:tag 1][image-registry-address]/[organization-name]/[image-name 2:tag 2]*

   In the command:

- *[image-name 1:tag 1]*: Replace it with the actual name and tag of the image to push.
- *[image-registry-address]*: You can obtain the address on the SWR console, that is, the domain name at the end of the login command in **2.b**.
- *[organization-name]*: Replace it with the name of the organization created.
- *[image-name 2:tag 2]*: Replace it with the desired image name and tag.

Example:

docker tag nginx:v1 swr.ap-southeast-1.myhuaweicloud.com/cloud-develop/nginx:v1

4. Run the following command to push the image to SWR:

   docker push *[image-registry-address]/[organization-name]/[image-name 2:tag 2]*

   Example:

   docker push swr.ap-southeast-1.myhuaweicloud.com/cloud-develop/nginx:v1

   The following information will be returned upon a successful push:

   ```
   The push refers to repository [swr.ap-southeast-1.myhuaweicloud.com/cloud-develop/nginx:v1]
   fbce26647e70: Pushed
   fb04ab8effa8: Pushed
   8f736d52032f: Pushed
   009f1d338b57: Pushed
   678bbd796838: Pushed
   d1279c519351: Pushed
   f68ef921efae: Pushed
   v1: digest: sha256:0cdfc7910db531bfa7726de4c19ec556bc9190aad9bd3de93787e8bce3385f8d size: 1780
   ```

   To view the pushed image, refresh the **My Images** page.

containerd

1. Log in to the **SWR console**.
2. In the navigation pane, choose **My Images**. Then click the name of the target image.
3. On the **Pull/Push** tab, click **Generate Push Command** and copy the command.

   NOTE

   The command is only valid for six hours after it is generated. To obtain a push command that will remain valid for a long term, see **Obtaining a Standard Push/Pull Command for containerd**.

4. Log in to the VM running containerd as **root**.
5. On the server, run the command copied in **3**.

6. Check whether the image is pushed successfully.

NOTE

- If the number of pushed images reaches the quota, new images cannot be pushed. However, if there are existing images but the number of image tags does not reach the quota, more images can be pushed.
- If the number of tags of an image reaches the quota, the image cannot be pushed, but other images can still be pushed.
- When images or image tags are pushed concurrently, the number of pushed images or image tags may exceed the quota. To release the quota, you need to delete all images or image tags that exceed the quota and then delete unused images or image tags.

FAQ

Why Does an Image Fail to Be Pushed Using a Container Engine Client?

Other

Uploading an Image on the SWR Console

Procedure

Step 1 Log in to the **SWR console**.

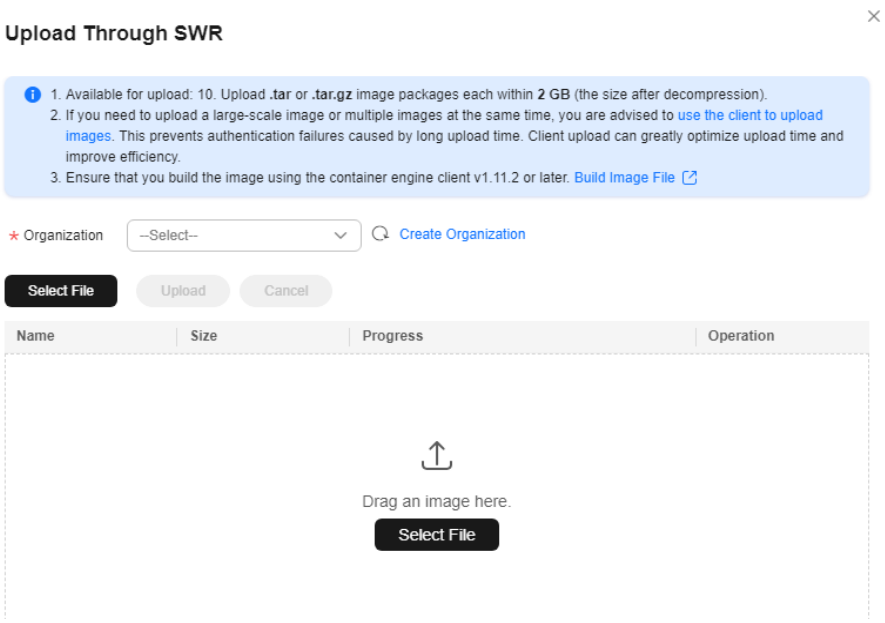
Step 2 In the navigation pane, choose **My Images**. Then click **Upload Through SWR**.

Step 3 In the displayed dialog box, select an organization. Then, click **Select File** to upload an image file.

NOTE

If you select multiple images, they will be uploaded one by one. Concurrent upload is not supported.

Figure 6-2 Uploading an image on the SWR console



Step 4 Click **Upload**.

If **Completed** is displayed, the image is uploaded successfully.

----End

NOTE

- If the number of pushed images reaches the quota, new images cannot be pushed. However, if there are existing images but the number of image tags does not reach the quota, more images can be pushed.
- If the number of tags of an image reaches the quota, the image cannot be pushed, but other images can still be pushed.
- When images or image tags are pushed concurrently, the number of pushed images or image tags may exceed the quota. To release the quota, you need to delete all images or image tags that exceed the quota and then delete unused images or image tags.

FAQ

Why Does an Image Fail to Be Uploaded Through the SWR Console?

Other

6.3 Obtaining a Long-Term Login or Image Push/Pull Command

Scenarios

When you use a container engine client to push or pull images, you can use a temporary or long-term command to connect to SWR.

- For Docker, you can **obtain a long-term login command**. If you just push or pull images occasionally, a temporary command will be enough for you.
- For containerd, you can **obtain a long-term pull/push command**. If you just push or pull images occasionally, a temporary command will be enough for you.

CAUTION

Keep the long-term login command of Docker containers and the long-term push/pull command of containerd containers secure to prevent information leakage.

For details, see **What Are the Differences Between Long-Term and Temporary Login Commands?**

This section describes how to obtain a long-term Docker login command and containerd push or pull command.

NOTE

A temporary login command will expire in six hours. If the command expires, clear the browser cache before you generate a new one.

To be better compatible with the new IAM console, enhanced commands are introduced. To distinguish from them, existing commands are renamed as

standard commands. An enhanced command simplifies permission control by removing SWR-based authorization. Only IAM-based authorization is used for easier but more comprehensive permission control over image push and pull. For more information, see **Table 6-1**.

Table 6-1 Comparison between a standard login command and an enhanced login command

Difference	Standard Login Command	Enhanced Login Command
Scope	Unlimited	Available only on the new IAM console.
Validity period	There are temporary and long-term login commands. For details about their differences, see **What Are the Differences Between Long-Term and Temporary Login Commands?**	For security purposes, only temporary login commands are provided, with a validity period of 24 hours.
Permissions control	IAM-based permission control. For details, see **Configuring Permissions in IAM**. SWR-based permission control. For details, see **Configuring Permissions in SWR**.	Only IAM-based permission control is provided. For details, see **Configuring Permissions in IAM**. For enhanced login commands, IAM condition keys are enhanced. For details, see **Table 6-2**.
Interconnection with log auditing	Yes	Yes
Condition keys	See **Table 6-2**.	See **Table 6-2**.

Table 6-2 IAM condition keys for push and pull using standard and enhanced login commands

Condition Key	Standard Login Command		Enhanced Login Command
	Temporary Login Command	Long-Term Login Command	
g:CalledVia	No	No	No
g:CalledViaFirst	No	No	No

Condition Key	Standard Login Command		Enhanced Login Command
	Temporary Login Command	Long-Term Login Command	
g:CalledViaLast	No	No	No
g:PrincipalTag/tag-key	No	No	Yes
g:MFAAge	No	No	Yes
g:MFAPresent	No	No	Yes
g:SourceIdentity	No	No	Yes
g:TokenIssueTime	No	No	Yes
g:ViaService	No	No	Yes
g:DomainId	Yes	No	Yes
g:DomainName	Yes	No	Yes
g:PrincipalAccount	Yes	Yes	Yes
g:PrincipalUrn	Yes	Yes	Yes
g:PrincipalIsService	Yes	Yes	Yes
g:PrincipalIsRootUser	Yes	Yes	Yes
g:PrincipalServiceName	Yes	No	Yes
g:PrincipalType	Yes	Yes	Yes
g:PrincipalId	Yes	Yes	Yes
g:UserName	Yes	Yes	Yes
g:UserId	Yes	Yes	Yes
g:PrincipalOrgPath	Yes	Yes	Yes
g:PrincipalOrgId	Yes	Yes	Yes
g:PrincipalOrgManagementAccountId	Yes	Yes	Yes
g:ResourceOrgId	Yes	Yes	Yes
g:ResourceOrgPath	Yes	Yes	Yes
g:Referer	Yes	Yes	Yes

Condition Key	Standard Login Command		Enhanced Login Command
	Temporary Login Command	Long-Term Login Command	
g:SecureTransport	Yes	Yes	Yes
g:SourceIp	Yes	Yes	Yes
g:SourceVpc	Yes	Yes	Yes
g:SourceVpce	Yes	Yes	Yes
g:UserAgent	Yes	Yes	Yes
g:RequestedRegion	Yes	Yes	Yes
g:RequestTag/tag-key	No	No	No
g:ResourceAccount	Yes	Yes	Yes
g:ResourceTag/tag-key	No	No	No
g:TagKeys	No	No	No
g:EnterpriseProjectId	No	No	No
g:SourceAccount	No	No	No
g:SourceUrn	No	No	No
g:CurrentTime	Yes	Yes	Yes

How to Obtain

Perform the following operations to obtain a long-term Docker login command and containerd push or pull command.

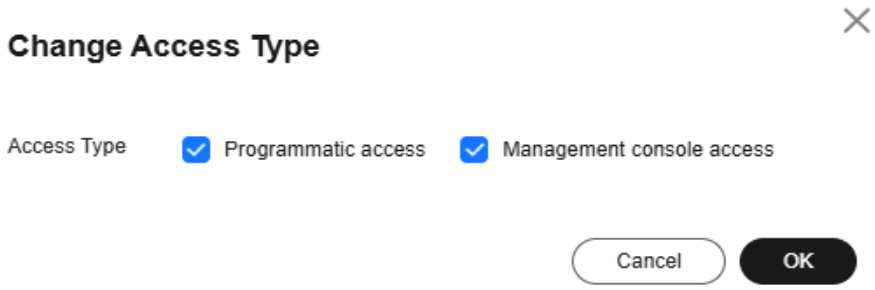
Obtaining a Standard Long-Term Login Command for Docker

Step 1 Obtain the programming access permission. (If you already have the programming access permission, skip this step.)

1. Log in to the **management console** as an administrator.
2. Click  in the upper left corner and select a region and a project.
3. Click  in the navigation pane and choose **Management & Governance** > **Identity and Access Management**.

4. In the user list, search for the user you want to grant programmatic access to.
5. Click the user to go to its details page. Click next to **Access Type**. Select **Programmatic access**. (You can also select both.)

Figure 6-3 Changing the access type



Step 2 Obtain an AK/SK. (If you already have an AK/SK, skip this step.)

NOTE

Ensure that you have permission to access the IAM service. For details about the authorization, see **Creating a User Group and Assigning Permissions**.

1. Log in to the IAM console, hover the cursor on the username, and click **My Credentials**.
2. In the navigation pane, choose **Access Keys** and click **Create Access Key**.
3. Enter a description, and click **OK**.
4. In the displayed dialog box, click **Download**.
5. After the credential is downloaded, obtain the AK and SK from the **credentials.csv** file.

Table 6-3 Example of credentials.csv

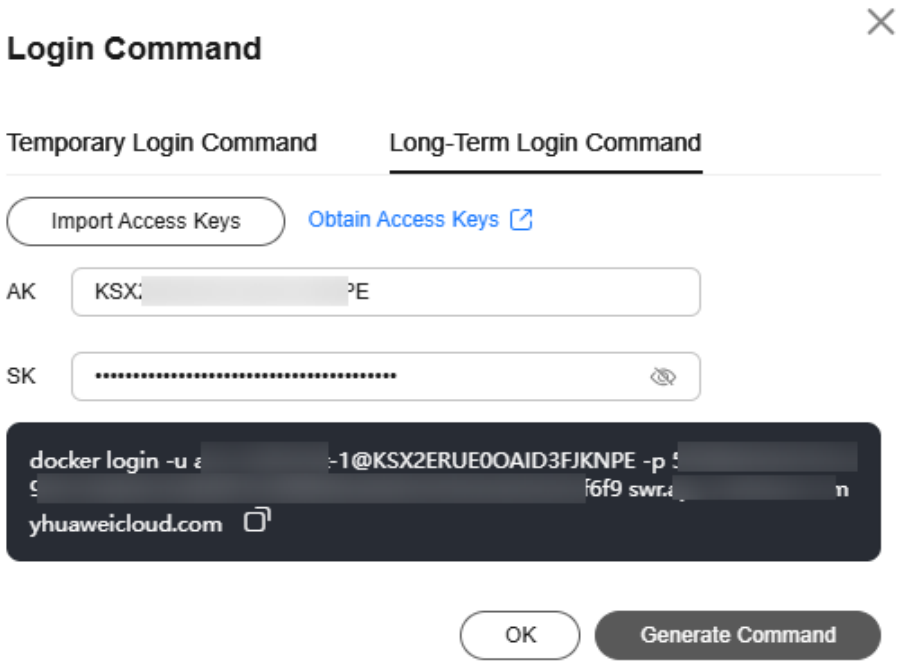
Username	Access Key ID	Secret Access Key
*a*****	*RVHVMX******	*H3nPwzgZ******

NOTE

Each credential file can be downloaded only once. Keep it secure.

Step 3 Log in to the **SWR console**.

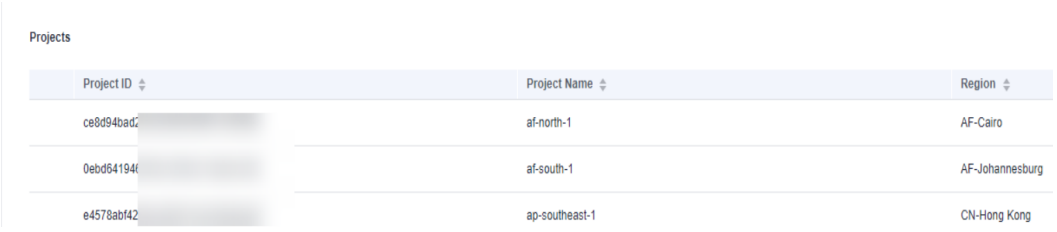
Step 4 Click **Generate Login Command**. On the **Long-Term Login Command** tab, click **Import Access Keys** to upload **credentials.csv** or enter the **Access Key ID** and **Secret Access Key** contained in **credentials.csv** that is obtained in ***Step 2***. Click **Generate Command**. If your console does not have the **Long-Term Login Command** tab, skip this step and manually concatenate a long-term login command by performing the following steps.



Step 5 Obtain the project name and image registry address.

1. Log in to the IAM console.
2. Hover the cursor over the username in the upper right corner.
3. Choose **My Credentials** from the drop-down list.
4. In the project list, find the region and project your VM belongs to.

Figure 6-4 Region and project



Step 6 Log in to a Linux PC and run the following command to obtain a login key:

printf "*$AK*" | openssl dgst -binary -sha256 -hmac "*$SK*" | od -An -vtx1 | sed 's/[\n]//g' | sed 'N;s/\n//'

Replace *AK* with the access key ID and *SK* with the secret access key in the **credentials** file obtained in **Step 2**.

Example:

```
printf "RVHVMX******" | openssl dgst -binary -sha256 -hmac "H3nPwzgZ******" | od -An -vtx1 | sed 's/[ \n]//g' | sed 'N;s/\n//'
```

After the command is executed, the following login key is obtained:

```
cab4ceab4a1545**************
```


> **NOTE**
>
> The login key is an example only.

Step 7 Use all the information you obtained to generate a long-term login command in the following format:

docker login -u *[project-name]*@*[AK]* **-p** *[login-key] [image-registry-address]*

The project name and registry address are obtained in **Step 5**, the AK in **Step 2**, and the login key in **Step 6**.

Example:

```
docker login -u ap-southeast-3@RVHVMX****** -p cab4ceab4a1545********** swr.ap-southeast-3.myhuaweicloud.com
```

If "Login Succeeded" is displayed, the login is successful.

> **NOTE**
>
> - The login key is encrypted and cannot be decrypted. So, other users cannot obtain your SK from **-p**.
> - The login command can be executed on other Docker clients.

Step 8 (Optional) When you log out of the registry, run the following commands to delete your authentication information:

```
cd /root/.docker/
rm -f config.json
```

Step 9 (Optional) Run **history -c** to delete the operation records.

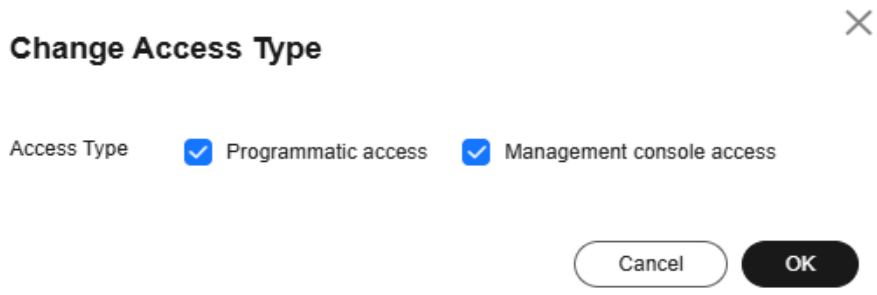
----End

Obtaining a Standard Push/Pull Command for containerd

Step 1 Gain programmatic access. If you already have it, skip this step.

1. Log in to the **management console** as an administrator.
2. Click in the upper left corner and select a region and a project.
3. Click in the navigation pane and choose **Management & Governance** > **Identity and Access Management**.
4. In the user list, search for the user you want to grant programmatic access to.
5. Click the user to go to its details page. Click next to **Access Type**. Select **Programmatic access**. (You can also select both.)

Figure 6-5 Changing the access type



Step 2 Obtain an AK/SK. If you already have it, skip this step.

NOTE

Ensure that you have permission to access the IAM service. For details about the authorization, see **Creating a User Group and Assigning Permissions**.

1. Log in to the IAM console, hover the cursor on the username, and click **My Credentials**.
2. In the navigation pane, choose **Access Keys** and click **Create Access Key**.
3. Enter a description, and click **OK**.
4. In the displayed dialog box, click **Download**.
5. After the credential is downloaded, obtain the AK and SK from the **credentials.csv** file.

Table 6-4 Example of credentials.csv

Username	Access Key ID	Secret Access Key
*a*****	*RVHVMX******	*H3nPwzgZ******

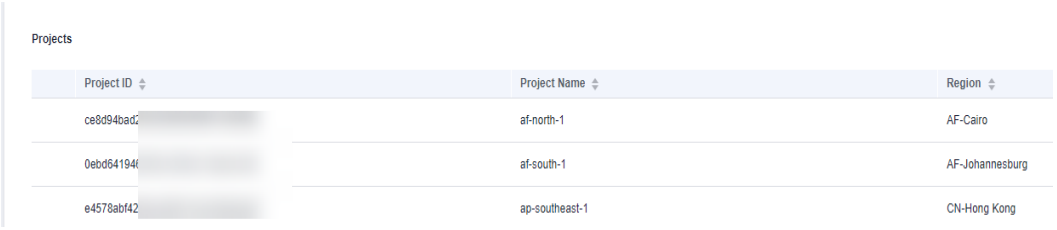
NOTE

Each credential file can be downloaded only once. Keep it secure.

Step 3 Obtain the project name and image registry address.

1. Log in to the **IAM console**.
2. Hover the cursor over the username in the upper right corner.
3. Choose **My Credentials** from the drop-down list.
4. In the project list, find the region and project your VM belongs to.

Figure 6-6 Region and project



5. Use the project information you obtained to generate an image registry address in the format of **swr.***project-name***.myhuaweicloud.com**.

   For example, if the VM of user a***** belongs to AP-Singapore, the image registry address will be **swr.ap-southeast-3.myhuaweicloud.com**.

Step 4 Log in to a Linux PC and run the following command to obtain a login key:

printf "*$AK*" | openssl dgst -binary -sha256 -hmac "*$SK*" | od -An -vtx1 | sed 's/[\n]//g' | sed 'N;s/\n//'

Replace **AK** with the access key ID and **SK** with the secret access key in the **credentials.csv** file in **Step 2**.

Example:

```
printf "RVHVMX******" | openssl dgst -binary -sha256 -hmac "H3nPwzgZ******" | od -An -vtx1 | sed 's/[ \n]//g' | sed 'N;s/\n//'
```

After the command is executed, the following login key is obtained:

```
cab4ceab4a1545***************
```

NOTE

The login key is an example only.

Step 5 Concatenate the obtained information to form a long-term containerd command.

1. Image pull command

ctr image pull --user [project-name]**@**[*AK*]:[*login-key*] [*image-repository-address*]/{organization-name}/{image-name}:{tag}

In the command, the project name and image registry address are obtained in **Step 3**, the AK is the *Access Key Id* field in the credentials.csv file obtained in **Step 2**, and the login key is the execution result in **Step 4**.

2. Image push command

ctr image push --user [project-name]**@**[*AK*]:[*login-key*] [*image-repository-address*]/{organization-name}/{image-name}:{tag}

In the command, the project name and image registry address are obtained in **Step 3**, the AK is the *Access Key Id* field in the credentials.csv file obtained in **Step 2**, and the login key is the execution result in **Step 4**.

NOTE

- The login key is encrypted and cannot be decrypted into an SK.
- The commands can be executed on other servers running containerd to pull and push images.

----End

6.4 Uploading an Image

Scenarios

This section describes how to upload an image on the SWR console.

Constraints

- Up to 10 files can be uploaded at a time. The size of a single file (including each decompressed file) cannot exceed 2 GB.
- If the image package to be uploaded is created using a Docker container engine client, the Docker version must be 18.06 or later.
- A single tenant can push up to 500 images and 300 tags for each image. If the quotas are exceeded, the push will fail.

Prerequisites

- You have created an organization in SWR. For details, see **Creating an Organization**.
- The image has been saved as a **.tar** or **.tar.gz** file. For details, see **Creating an Image Package**.

Procedure

Step 1 Log in to the **SWR console**.

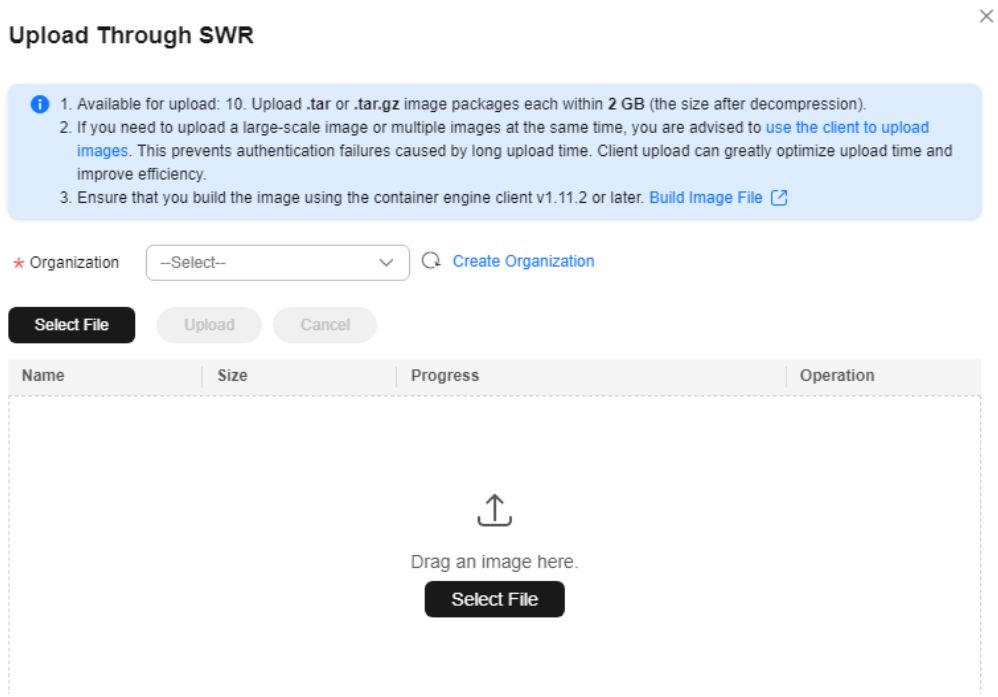
Step 2 In the navigation pane, choose **My Images**. Then click **Upload Through SWR**.

Step 3 In the displayed dialog box, select an organization. Then, click **Select File** to upload an image file.

NOTE

If you select multiple images, they will be uploaded one by one. Concurrent upload is not supported.

Figure 6-7 Uploading an image on the SWR console



Step 4 Click **Upload**.

If **Completed** is displayed, the image is uploaded successfully.

----End

NOTE

- If the number of pushed images reaches the quota, new images cannot be pushed. However, if there are existing images but the number of image tags does not reach the quota, more images can be pushed.
- If the number of tags of an image reaches the quota, the image cannot be pushed, but other images can still be pushed.
- When images or image tags are pushed concurrently, the number of pushed images or image tags may exceed the quota. To release the quota, you need to delete all images or image tags that exceed the quota and then delete unused images or image tags.

FAQ

Why Does an Image Fail to Be Uploaded Through the SWR Console?

6.5 Pulling an Image

Scenarios

To use an image stored in a repository, you need to pull it from the repository first. Then, you can use this image to deploy containerized applications in CCE or CCI. Pulling an image is actually downloading an image. Images are classified as public or private.

- Public images can be pulled by all users in any account. To improve image management security, SWR also supports permission control over the download of public images through **related control policies**.
- Private images are controlled by specific permission management. You can grant permissions to allow users to read, edit, or manage private images. For details, see **Granting Permissions for a Specific Image**.

You can use Docker or containerd to pull images from SWR.

Prerequisites

- Your network is normal.
- If you are an IAM user, you have obtained the read and edit permissions on images in the organization from the administrator. For details, see **Granting Permissions**.
- On the **My Images** page, **Private Images** list your own images in your organization and **Shared Images** list private images shared by other users in the organization.

Pulling My Image

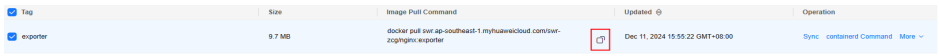
You can use Docker or containerd to pull images from SWR.

Docker

1. Log in to the server running the container engine as user **root**.
2. Obtain a login command by referring to **(Recommended) Pushing an Image Using a Container Engine Client** and connect to SWR.

3. Log in to the **SWR console**.
4. In the navigation pane, choose **My Images**. Then click the name of the target image.
5. On the **Tags** tab, locate the image tag you desire and click in the **Image Pull Command** column to copy the command.

   Figure 6-8 Obtaining the image pull command

6. On the server, run the command copied in **5**.

   Example: **docker pull swr.cn-east-3.myhuaweicloud.com/group/nginx:v2.0.0**

   Run the **docker images** command to check whether the image is successfully pulled.

   ```
   # docker images
   REPOSITORY                                TAG       IMAGE ID       CREATED         SIZE
   swr.*******.*******.com/group/nginx       v2.0.0    22f2bf2e2b4f   5 hours ago     22.8MB
   ```

7. (Optional) Save the image to an archive file.

   docker save [*image-name:tag-name*] **>** [*archive-file-name*]

   Example: **docker save swr.cn-east-3.myhuaweicloud.com/group/nginx:v2.0.0 > nginx.tar**

containerd

1. Log in to the **SWR console**.
2. In the navigation pane, choose **My Images**. Then click the name of the target image.
3. On the **Tags** tab, click **containerd Command** in the **Operation** column to copy the image pull command. Alternatively, go to the **Pull/Push** tab to copy the image pull command.

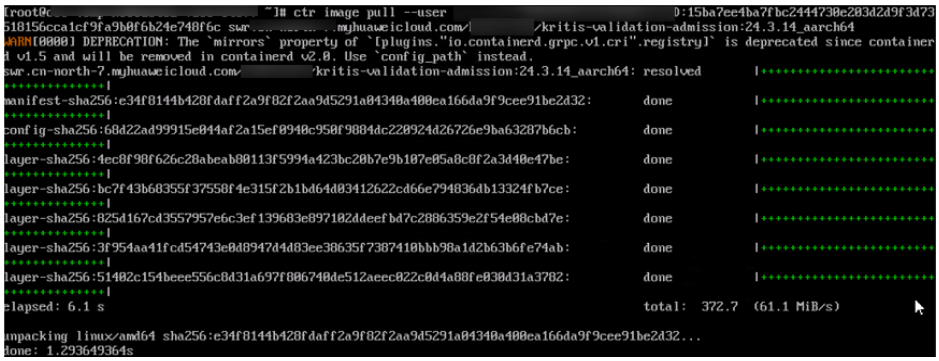
   NOTE

   The command is only valid for six hours after it is generated. To obtain a pull command that will remain valid for a long term, see **Obtaining a Standard Push/Pull Command for containerd**.

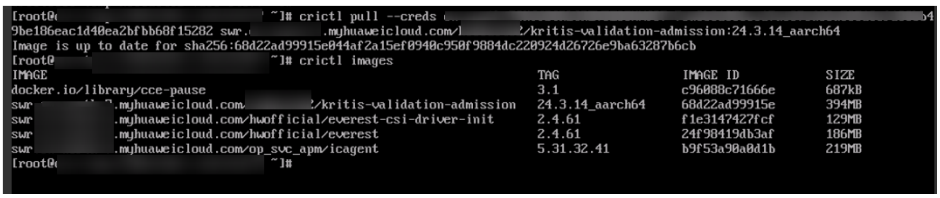
4. Log in to the VM running containerd as **root**.
5. On the server, run the command copied in **3**.
   - If the command was copied from the **Operation** column, run it as follows.

   - If the command was copied from the **Pull/Push** tab, run it as follows.



6. Check whether the image is pulled successfully.
   - If the command was copied from the **Operation** column, run **crictl images** to check whether the pull is successful.



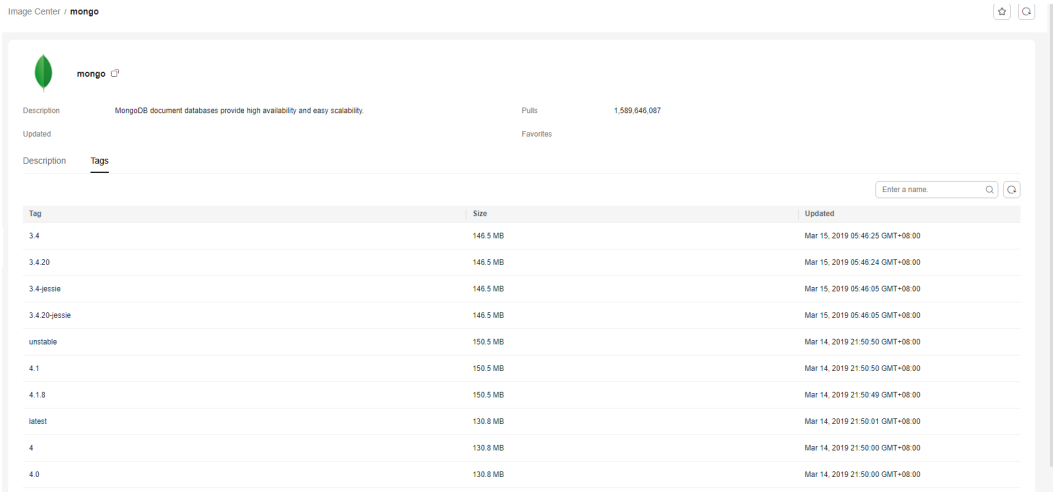
   - If the command was copied from the **Pull/Push** tab, run **ctr images list** to check whether the pull is successful.

Pulling an Image from the Image Center

You can pull an image from the Image Center without specifying a registry address. For example, you can **connect the VM running the container engine to SWR** and run the following command to pull the **mongo** image easily:

docker pull mongo:*4.1*

Figure 6-9 mongo image details



6.6 Modifying an Image

Scenarios

After pushing an image, you can modify it, including its type (private by default), category, and description.

Public images can be downloaded by all users. To improve image management security, SWR also supports permission control over the download of public images through **related control policies**.

Private images are controlled by specific permission management. You can grant permissions to allow users to read, edit, or manage private images. For details, see **Granting Permissions for a Specific Image**.

Procedure

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then click the name of the target image.

Step 3 On the details page, click **Edit** in the upper right corner. In the displayed dialog box, set **Sharing Type** (**Public** or **Private**), **Category**, and **Description**, and click **OK**.

Figure 6-10 Editing an image

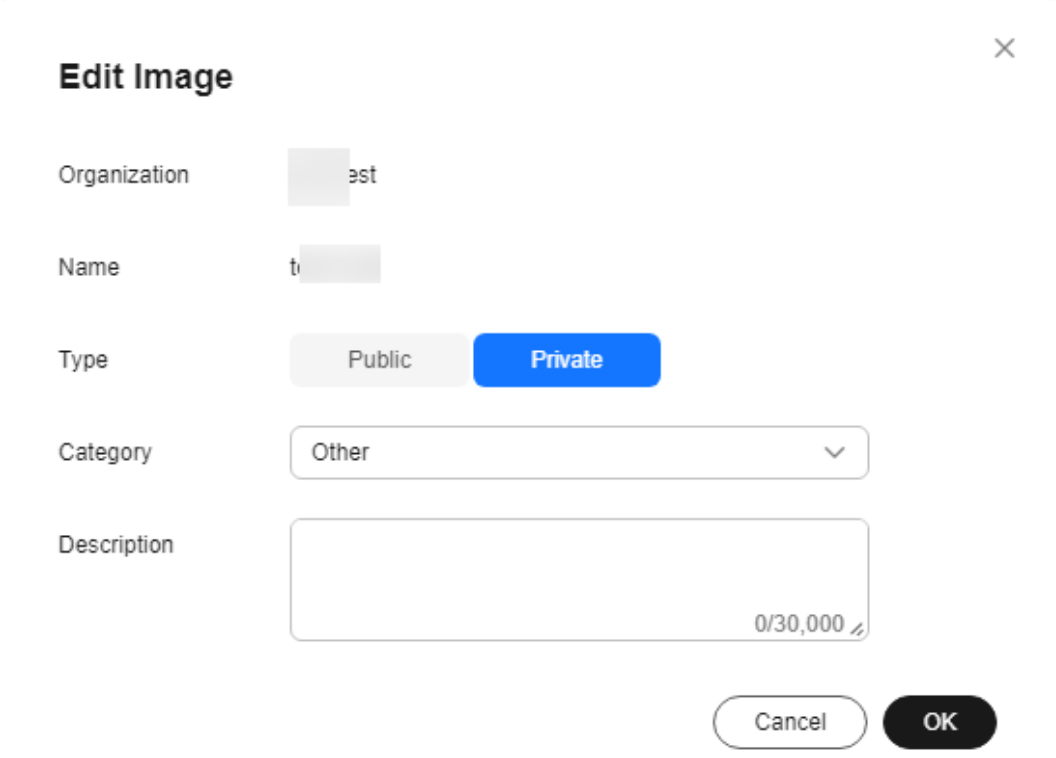


Table 6-5 Editing an image

Parameter	Description
Organization	The organization that the image belongs to
Image	Image name
Type	The following options are available: • Public • Private **NOTE** Public images can be pulled and used by all users. • If your container engine client and the image repository are in the same region, you can access the image repository through private networks. • If your container engine client and the image repository are in different regions, you need to pull images over the Internet.
Category	The following options are available: • Application server • Linux • Arm • Framework & Application • Database • Language • Other
Description	0 to 30,000 characters are allowed.

----End

6.7 Sharing a Private Image

Scenarios

You can share your private images with other accounts and grant them permission to pull the images.

A user under these accounts can then log in to the **SWR console** to check these images on the **My Images** > **Images From Others** page. The user can also click the name of an image to check its details, including the image tag and image pull command.

Constraints

- Only private images can be shared. Public images cannot be shared.

- Only IAM users authorized to manage the private images can share images. The users with whom you share your images only have the read-only permission, which only allows them to pull the images.
- You can share images only with accounts in the same region. Cross-region image sharing is not supported.
- A private image can be shared with a maximum of 500 tenants.

Procedure

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then click the name of the target image.

Step 3 On the details page, click the **Sharing** tab.

Step 4 Click **Share Image**. Set parameters based on **Table 6-6**, and click **OK**.

Figure 6-11 Sharing an image

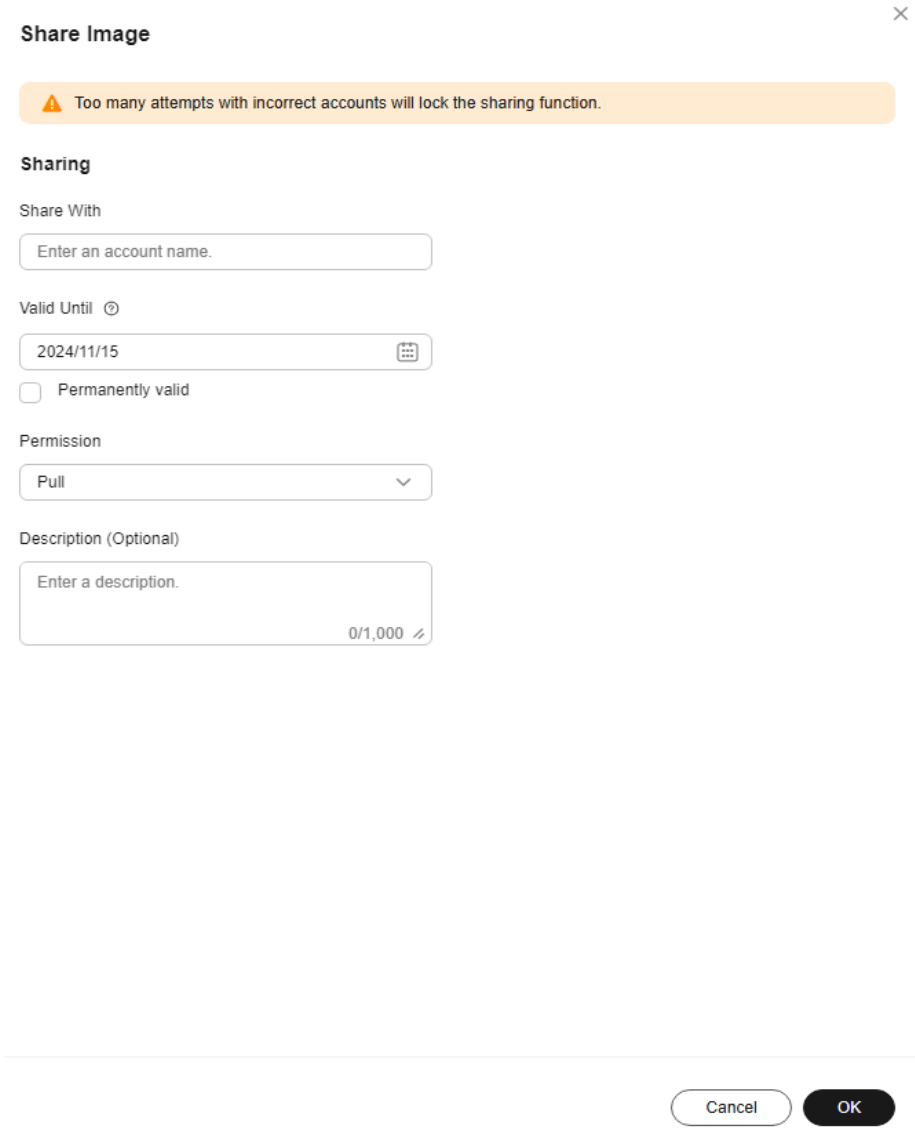


Table 6-6 Sharing an image

Parameter	Description
Share With	Enter an account name.
Valid Until	Set a validity period. If you want the image to be permanently accessible to the account, select **Permanently valid**.
Permission	Only **Pull** is available currently.
Description	0 to 1,000 characters are allowed.

Step 5 To view all the shared images, choose **My Images** in the navigation pane, click the **Private Images** tab, and select **My shared images**.

NOTE

To share multiple images, choose **My Images** > **Private Images**, select the images, and click **Share**.

----End

6.8 Adding a Trigger

Scenarios

SWR works with Cloud Container Engine (CCE) to enable automatic application updates. This can be achieved if a trigger is added for images.

Prerequisites

- You have permission to access CCE. For details, see **Fine-grained CCE Authorization** or **Fine-grained CCI Authorization**.

  NOTE

  You will be billed for image scanning in CCE or CCI based on their pricing. For details about CCE pricing, see **CCE Billing Overview**.

  For details about CCI billing, see **CCI Billing Overview**.

- A containerized application has been created in CCE by using an image from SWR.

  If no applications are created, log in to the CCE console and create one. For details, see **Creating a Deployment** or **Creating a StatefulSet**. To create an application in CCI, see **Creating a Deployment**.

Procedure

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then click the image name to go to the details page.

Step 3 Click the **Triggers** tab, and then click **Add Trigger**. Configure parameters based on **Table 6-7** and click **OK**.

Figure 6-12 Adding a trigger

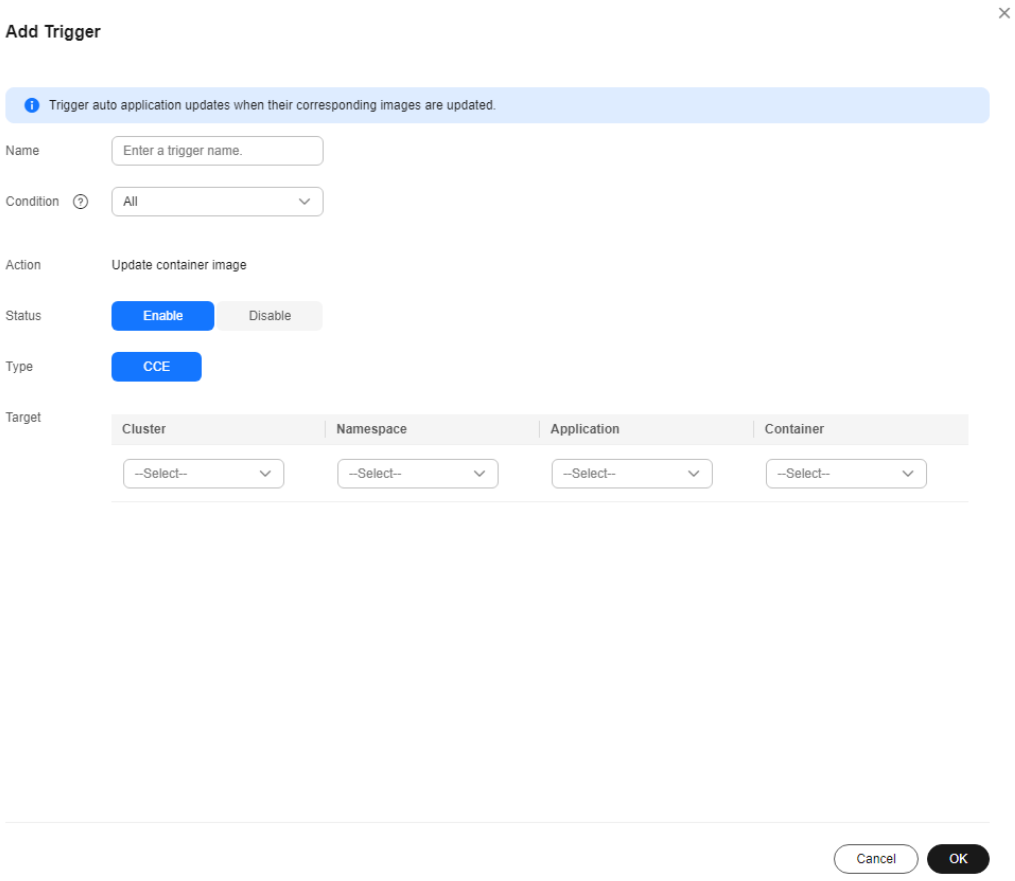


Table 6-7 Trigger

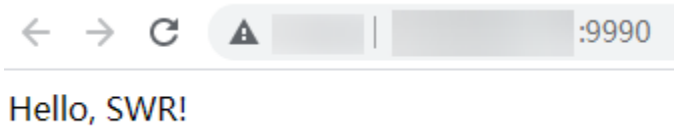
Parameter	Description
Name	The name of a trigger. The name can contain 1 to 64 characters, and must start with a letter. Only letters, digits, underscores (_), and hyphens (-) are allowed. The name cannot end with an underscore or hyphen. Consecutive underscores or hyphens are not allowed and an underscore cannot be placed next to a hyphen.

Parameter	Description
Condition	The following trigger conditions are supported: • **All**: Deployment is triggered when a new image tag is generated. • **Specified**: Deployment is triggered when a specific image tag is generated or updated. • **RegEx**: Deployment is triggered when an image tag that matches the regular expression is generated or updated. The regular expression rules are as follows: – *: matches any field that does not contain the path separator /. – **: matches any field that contains the path separator /. – ?: matches any single character except /. – *{option 1, option 2, ...}*: matches any of the options.
Action	Currently, only the action "update" can be a trigger. You need to specify the application to be updated and containers of the application.
Status	Select **Enable**.
Type	Select **CCE**.
Target	Select the containers whose image you want to update.

----End

Example 1: Trigger Condition Is All

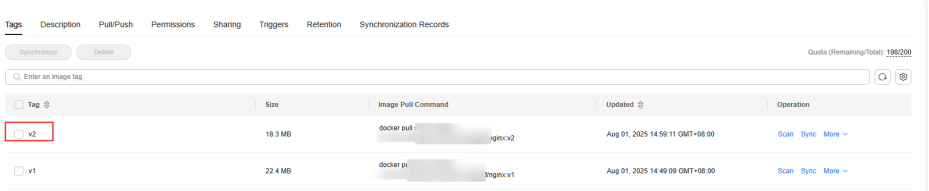
A Deployment named **nginx** is created using the Nginx image v1. The Deployment provides services to external systems with a welcome page displaying **Hello, SWR!**



1. Add a trigger to the Nginx image.

   Set **Name** to **All_tags** and **Condition** to **All**. Select the application and all its containers that use the Nginx image.

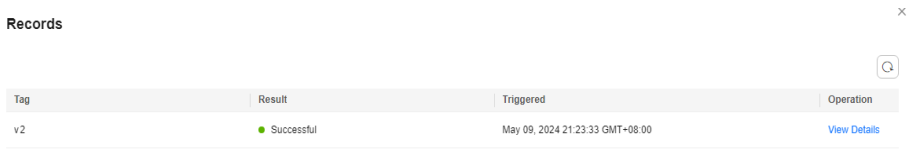
2. Push the Nginx image v2 to SWR. The welcome page of the Deployment created using this new image should display **Hello, SoftWare Repository for Container!**



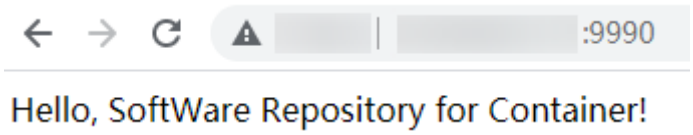
3. Check whether the deployment is triggered successfully.

   On the **Triggers** tab, locate the trigger and click **Records** to check whether the trigger is successful.

   Figure 6-13 Result

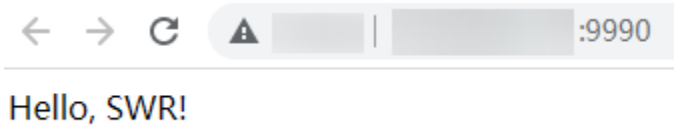


   The welcome page of the Deployment displays **Hello, SoftWare Repository for Container!**



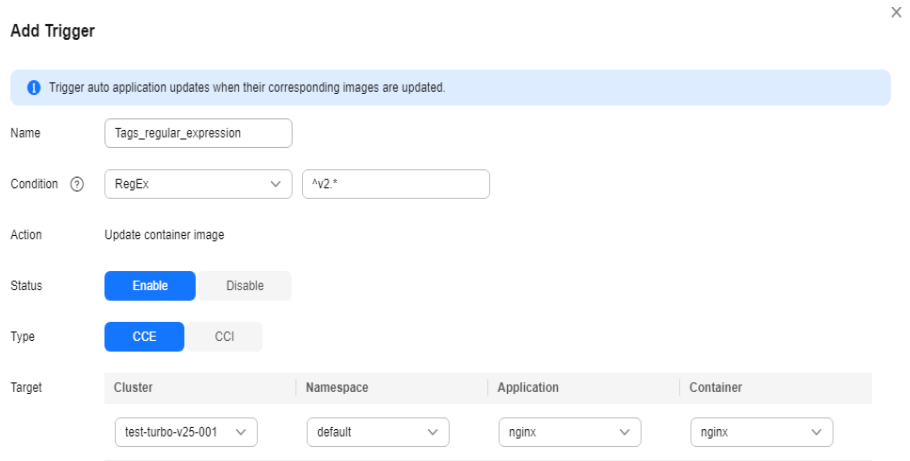
Example 2: Trigger Condition Is RegEx

A Deployment named **nginx** is created using the Nginx image v0. The Deployment provides services to external systems with a welcome page displaying **Hello, SWR!**

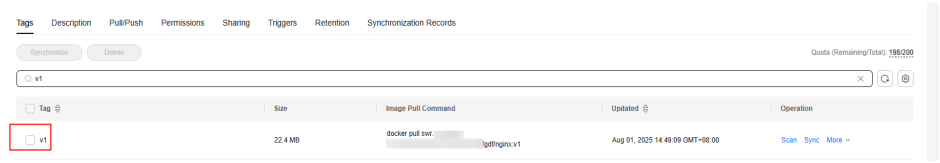


1. Add a trigger to the Nginx image.

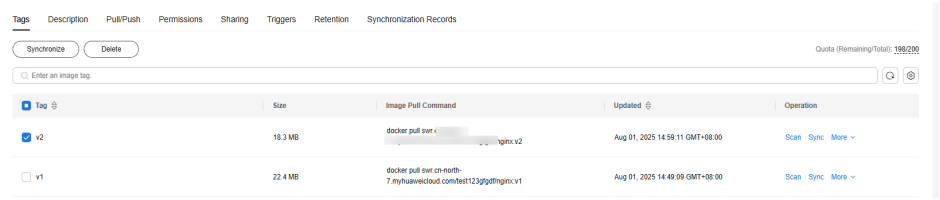
   Set **Name** to **Tags_regular_expression**, **Condition** to **RegEx**, and regular expression to **^v2.***. Select the application and all its containers that use the Nginx image.



2. Push the Nginx image v1 to SWR. The welcome page of the Deployment created using this new image should display **Hello, SWR! (v1)**.



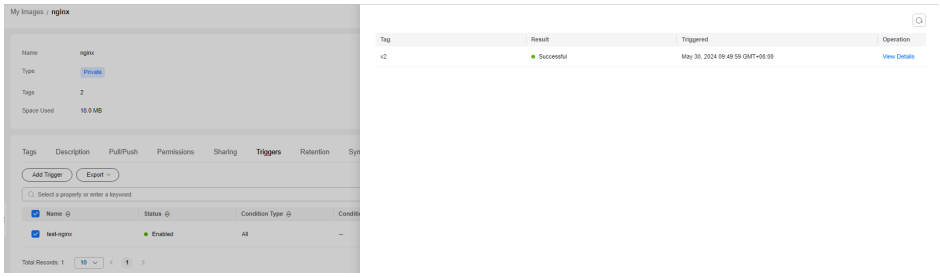
3. Push the Nginx image v2 to SWR. The welcome page of the Deployment created using this new image should display **Hello, SWR! (v2)**.



4. Check whether the deployment is triggered successfully.

   On the **Triggers** tab, click ⌄ to check the result. Only the deployment of the **nginx** image tagged with **v2** is triggered, because the image name matches the regular expression **^v2.***.

Figure 6-14 Result



The welcome page of the Deployment displays **Hello, SWR! (v2)**.

6.9 Adding an Image Retention Policy

Scenarios

You can add a retention policy to an image in SWR to automatically delete any unused image tags. The policy takes effect immediately after you set it. There are two types of policies:

- Number of days: keeping only image tags that have been pushed to SWR within a certain number of days.
- Number of tags: keeping only a certain number of the most recent image tags.

You can configure filters for your retention policy to prevent certain image tags from being affected by the retention policy.

Constraints

- Only one retention rule can be added to an image. If you want to add a new retention policy, you must delete the existing policy.
- You can view only the retention logs generated in the last three months.

Procedure

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then click the image name to go to the details page.

Step 3 On the **Retention** tab, click **Add Retention Policy**. Configure the policy based on **Table 6-8** and click **OK**.

Figure 6-15 Adding a retention policy

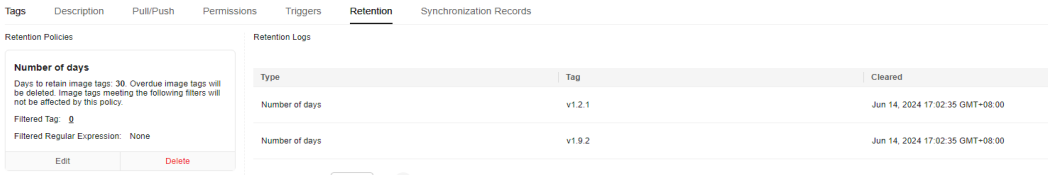
Table 6-8 Parameters for adding an image retention policy

Parameter	Description
Type	There are two types of retention policies: • Number of days: keeping only image tags that have been pushed to SWR within a certain number of days. • Number of tags: keeping only a certain number of the most recent image tags.
Retention Days	This parameter is available only when **Type** is set to **Number of days**. It indicates the number of the most recent days for which the images will be retained. The value must be an integer ranging from 1 to 365.

Parameter	Description
Count Limit	This parameter is available only when **Type** is set to **Number of tags**. It indicates the number of the most recent image tags to be retained. The value must be an integer ranging from 1 to 1,000.
Filter with Tags	Enter the image tags that need to be excluded from the retention policy.
Filter with Regular Expression	Enter a regular expression. Image tags matching this regular expression will be excluded from the retention policy.

After the retention policy is added, SWR immediately applies the policy and displays removed image tags (if any) in the **Retention Logs** area.

Figure 6-16 Checking the retention policy and logs

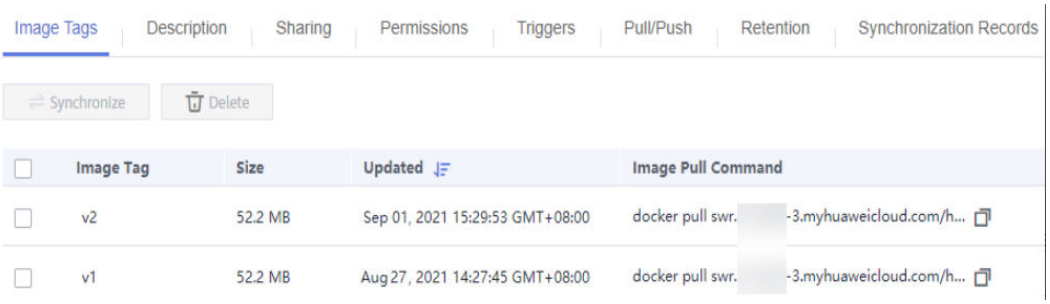


----End

Example 1: Set Policy Type to Number of Days

The image **nginx** has two tags: **v1** and **v2**. The following figure shows when they were updated.

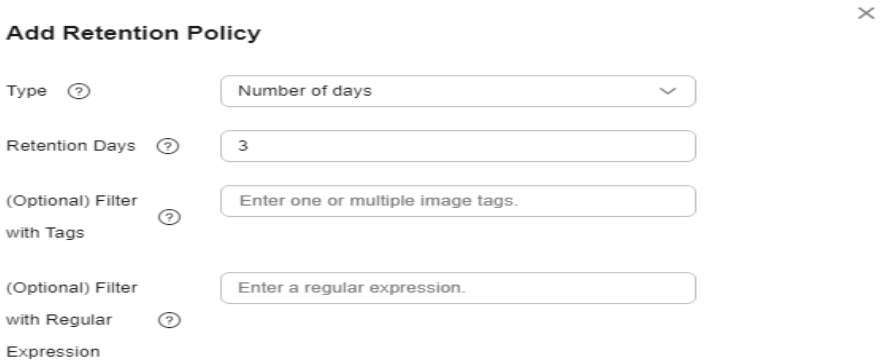
Figure 6-17 Image tags



1. Add a retention policy.

   Set **Type** to **Number of days**, and **Retention Days** to **3**.

Figure 6-18 Adding a retention policy

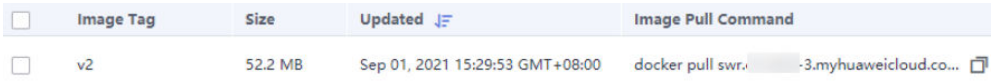


2. Check whether the retention policy has taken effect.

   Check **Retention Logs**. Nginx image v1 has been stored for more than three days (the current time is 2021/09/01 16:00:00). So, it is automatically removed.

   Check **Image Tag**. Only Nginx image v2 is left.

   Figure 6-19 Image v2

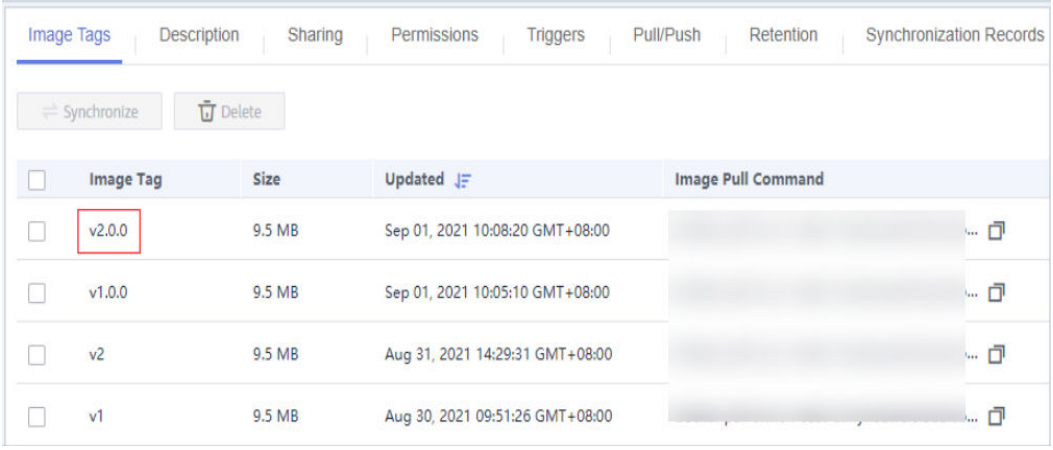


   The retention policy has taken effect.

Example 2: Set Policy Type to Number of Tags and Use a Regular Expression Filter

Click the **Image Tags** tab. The image **nginx** has four tags: **v1**, **v2**, **v1.0.0**, and **v2.0.0** as shown in the following figure.

Figure 6-20 Nginx image tags



1. Add a retention policy.

Set **Type** to **Number of tags**, **Count Limit** to **1**, and **Filter with Regular Expression** to **^v2.***.

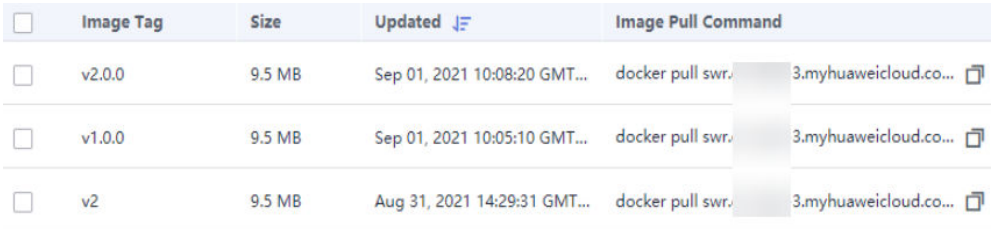
Figure 6-21 Adding a retention policy

2. Check whether the retention policy has taken effect.

   Based on the regular expression, Nginx image v2 and v2.0.0 will be excluded from the policy. In the policy, the count is 1, so only one of Nginx image v1 and v1.0.0 can be retained. Nginx image v1 will be removed because it is older.

   Check **Image Tag** in **Retention Logs**. If Nginx image v1 is removed, the retention policy has taken effect.

   Figure 6-22 Image tags



   The following regular expressions are for your reference:

   - ^[0-9]*$: filters out tags consisting of numbers.
   - ^.{2,5}$: filters out tags with a length ranging from 2 to 5 characters.
   - ^[a-z]+$: filters out tags consisting of lowercase letters.
   - ^[A-Za-z0-9]+$: filters out tags consisting of letters and numbers.

CAUTION

If there is an OR (|) operator in a regular expression, enclose the OR part in parentheses, or the regular expression will be parsed incorrectly and all tags of the image will be removed.

For example, if you only want to retain tags containing a or s, the regular should be (.*a.*|.*s.*).

6.10 Synchronizing an Image to Other Regions

Scenarios

You can synchronize newly pushed images to other regions either automatically or manually. Automatic synchronization can be performed for a single image or multiple images.

Table 6-9 Comparison between automatic and manual synchronization

Synchronization	Automatic	Manual
Scope	Automatic synchronization is applicable when images need to be updated frequently. Automatic synchronization cannot be applied to existing images.	Manual synchronization is applicable when images are not frequently updated. For example, they only need to be updated once or twice occasionally. Manual synchronization can be applied to existing images.
Execution	An image is synchronized automatically every time it is updated.	An image is synchronized when you click **Sync**.

NOTE

After you configure automatic image synchronization, new and updated images will be automatically synchronized to repositories of other regions. However, images that were pushed before automatic image synchronization was configured will not be automatically synchronized.

You can manually synchronize them by referring to **Can Existing Images be Automatically Synchronized?**

Constraints

- Only accounts and administrative users can use automatic image synchronization.
- A single tenant can push up to 500 images and 300 tags for each image. If the quotas are exceeded, the push will fail.
- Cross-region image synchronization is only available in the following regions: CN North-Beijing1, CN North-Beijing4, CN North-Ulanqab1, CN East-Shanghai1, CN East-Shanghai2, CN South-Guangzhou, CN Southwest-Guiyang1, CN-Hong Kong, AP-Bangkok, AP-Singapore, AP-Jakarta, and AF-Johannesburg.

Automatically Synchronizing a Single Image

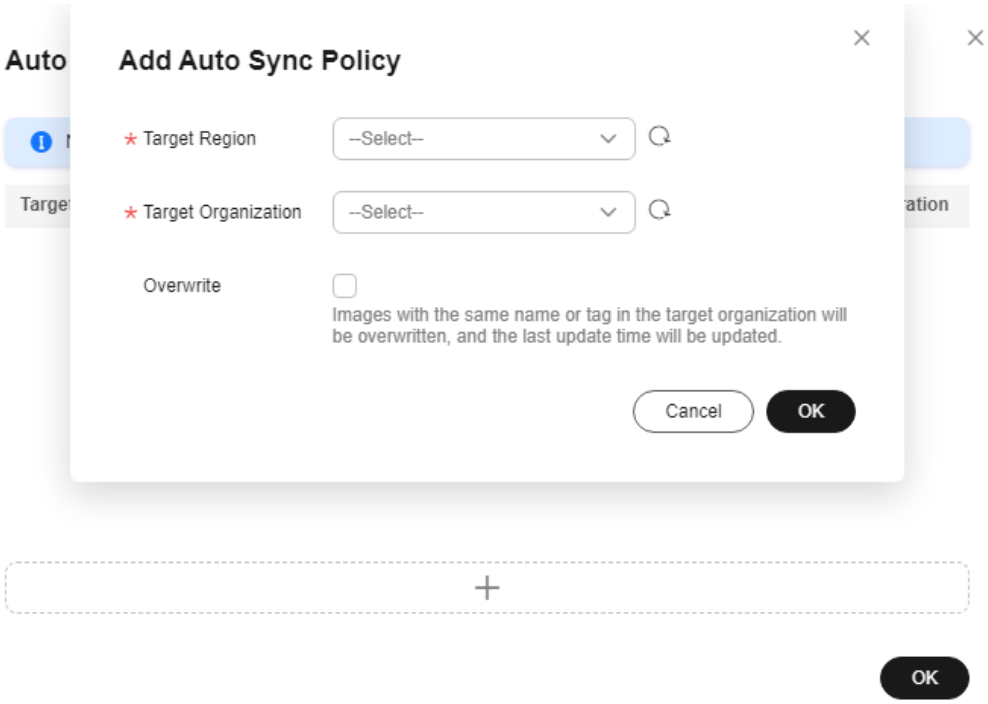
Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then click the name of the target image.

Step 3 On the image details page, click **Set Auto Sync** in the upper right corner.

Step 4 Click ⊕. Select a target region and a target organization. Click **OK**.

Figure 6-23 Configuring automatic image synchronization



- **Target Region**: The target region for image synchronization, for example, CN-Hong Kong.
- **Target Organization**: The target organization to which the image will be synchronized.
- **Overwrite**:

  Select this option if you want to overwrite any image with the same name and tag in the target organization.

  Deselect this option if you do not want to overwrite images in the target organization. If the image you are synchronizing has a duplicate name and tag with an image in the target organization, the synchronization will be canceled and you will receive a notification.

Step 5 On the **Synchronization Records** tab of the image details page, you can view the details of each synchronization task, including the start time, image tag, task status, type, and duration.

----End

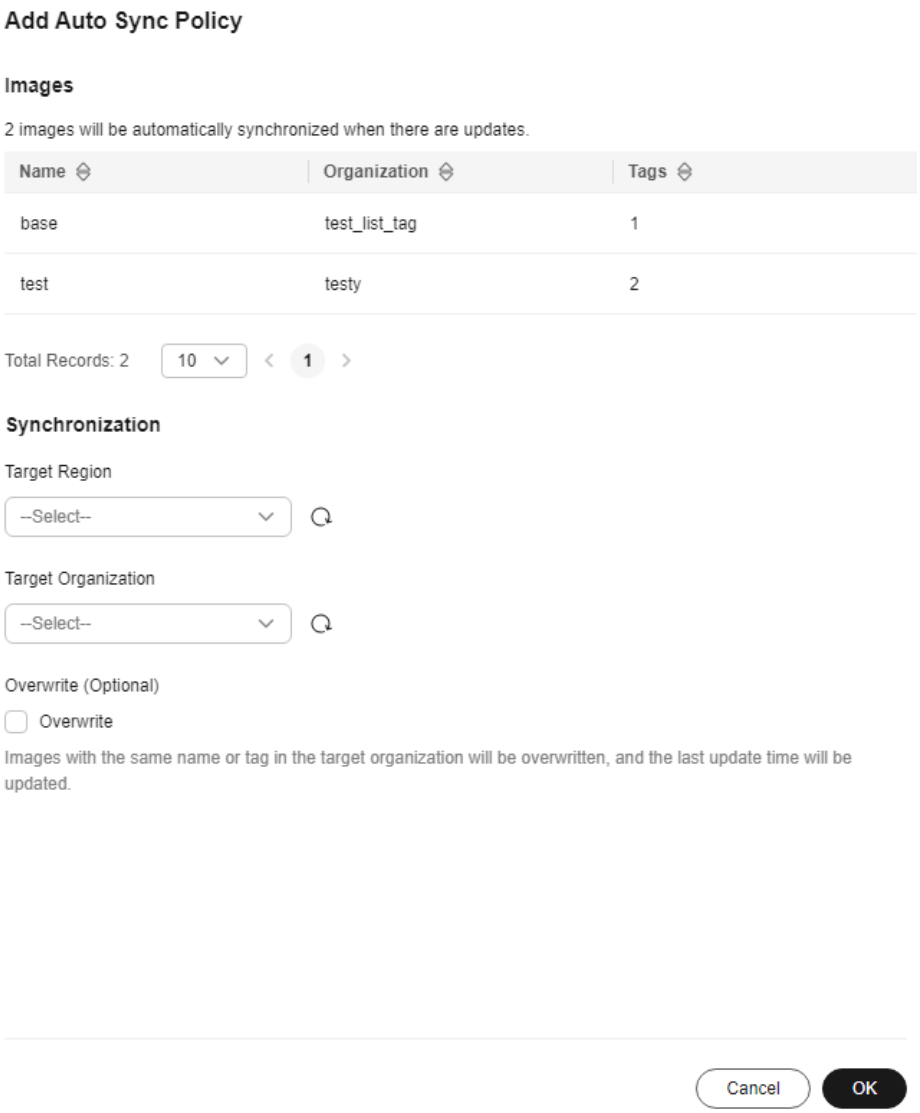
Automatically Synchronizing Multiple Images in a Batch

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Select the images and click **Auto Synch**.

> **CAUTION**
>
> A maximum of 50 images can be synchronized at a time.

Step 3 In the right pane, confirm the images to be synchronized, select the target region and organization, and determine whether to overwrite existing images. Click **OK**.



Step 4 Click an image name to go to the details page. On the **Synchronization Records** tab, you can view the details of each synchronization task, including the start time, image tag, task status, type, and duration.

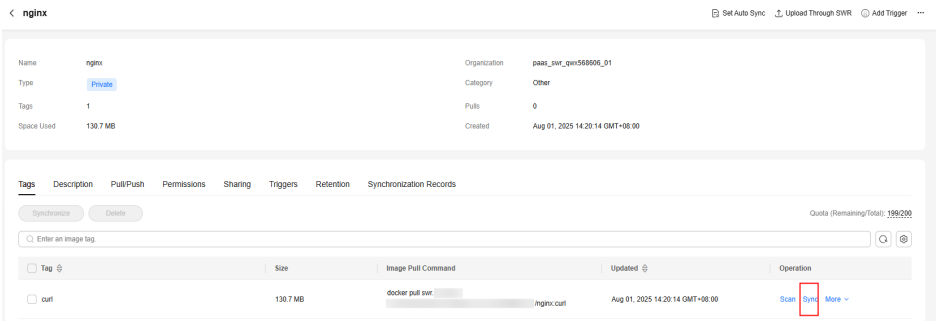
----End

Manually Synchronizing an Image

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then, click the image name to go to the details page.

Step 3 On the **Tags** tab, locate the image tag to be synchronized and click **Sync** in the **Operation** column. Select a target region and organization and determine whether to overwrite existing image tags. Click **OK**.



Step 4 On the **Synchronization Records** tab of the image details page, you can view the details of each synchronization task, including the start time, image tag, task status, type, and duration.

----End

Common Causes of Image Synchronization Failures

Table 6-10 Image synchronization failures

Status	Possible Cause	Solution
Failed	• The management plane network between the regions is abnormal. • The quota of images that can be pushed to the target repository is exceeded.	• Contact O&M engineers to check the network. • **Submit a service ticket**. • Try again later.
Failed: Sync timed out	The management plane network between the regions is abnormal.	• Contact O&M engineers to check the network. • Try again later.
Failed: Image already exists	The **Overwrite** option is not selected for image synchronization, but an image with the same name as the source exists in the target organization.	• If overwriting is not needed, ignore this failure. • If overwriting is needed, delete the synchronization rule and create another one with **Overwrite** selected.

6.11 Scanning an Image

Scenarios

You can scan your private images in SWR to check for vulnerabilities in just a few clicks and use the recommended solutions to mitigate the vulnerabilities, if any.

Constraints

- Currently, image scanning is only supported in CN North-Beijing1, CN North-Beijing4, CN East-Shanghai1, CN East-Shanghai2, and CN South-Guangzhou.
- You must have HSS permissions. For details, see **Fine-grained HSS Authorization**.

  NOTE

  You will be billed by HSS for image scanning. For details, see **HSS Billing Overview**.
- Multi-architecture images cannot be scanned.

Procedure

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **My Images**. Then click the image you want to scan.

NOTE

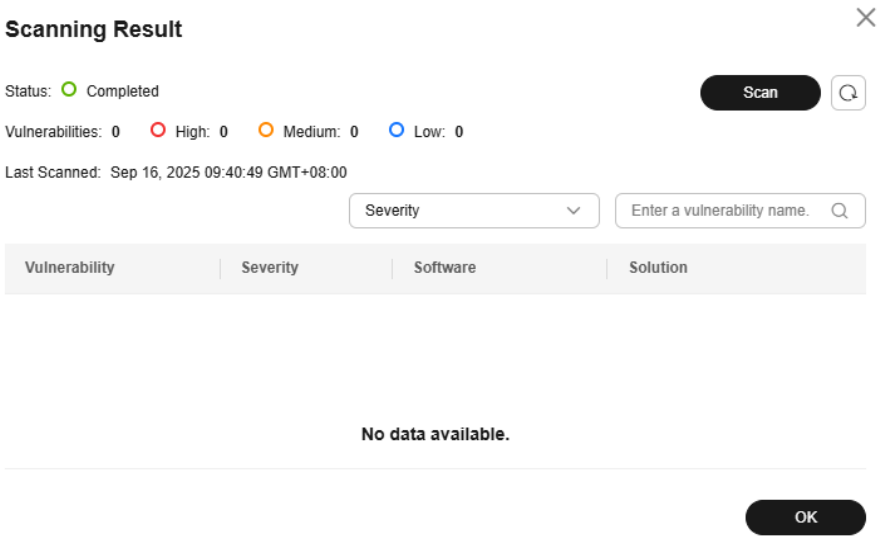
Before executing an image scanning task, ensure that at least one private image is available in **My Images**. If no private image is available, push one by referring to **Pushing an Image**.

Step 3 On the **Tags** tab, locate the image tag you want to scan and click **Scan** in the **Operation** column.

NOTE

You can scan an image only after it is synchronized to HSS. If the image to be scanned has not been synchronized to HSS, a dialog box is displayed. Click **Synchronize**.

Step 4 Click **Scan** and wait for the result.



- **Vulnerability Name**: the name of the vulnerability found on the image
- **Severity Level**: the severity of the vulnerability. You can determine whether immediate action is required based on the severity level.
- **Software Information**: version information about the software affected by the vulnerability
- **Solution**: solution to the vulnerability. Click the link in the **Solution** column to view the solution.

----End

If the image scanning fails, rectify the fault by referring to the following table.

Table 6-11 Causes and solutions for repository image scan failures

Failure Cause	Solution
Access to SWR failed.	**Submit a service ticket** to request technical support.
Insufficient SWR permissions.	Complete the authorization. For details, see **Authorization Methods**.
The image details could not be obtained. The image was not found in the repository.	On the **Host & Container Security Service** console, choose **Risk Management** > **Container Images** > in the navigation pane, click the **Repository Images** tab, and click **Synchronize Images** to update the image list and check whether the image exists.
Failed to download the image.	**Submit a service ticket** to request technical support.
The image is oversized.	The total image size cannot exceed 50 GB. You are advised to simplify images.

Failure Cause	Solution
The image has too many layers.	An image can contain a maximum of 127 layers, and each layer cannot exceed 10 GB. You are advised to simplify images.
Schema v1 images cannot be scanned.	You are advised to upgrade schema v1 images to v2.
If the image scanning duration exceeds 3 hours, the system automatically stops the scanning.	You are advised to simplify images.

For more information about container image scanning, see **Repository Image Security Scan**.

6.12 Image Center

Scenarios

SWR provides a large number of public images. You can add public container images to your favorites and push them to your repository.

CAUTION

Images in Image Center are provided and maintained by the open-source communities and are only used for testing by developers. They cannot be used in commercial environments.

Constraints

Image Center is only available in the following regions:

CN North-Beijing1, CN North-Beijing4, CN East-Shanghai1, CN East-Shanghai2, CN South-Guangzhou, CN Southwest-Guiyang1, CN-Hong Kong, AP-Bangkok, AP-Singapore, AF-Johannesburg, and LA-Santiago

Adding an Image to Favorites

Step 1 Log in to the **SWR console**.

Step 2 In the navigation pane, choose **Image Resources** > **Image Center**.

Step 3 Locate the desired image and click ☆ on the right.

You can view all your favorite images on the **My Favorites** page.

----End

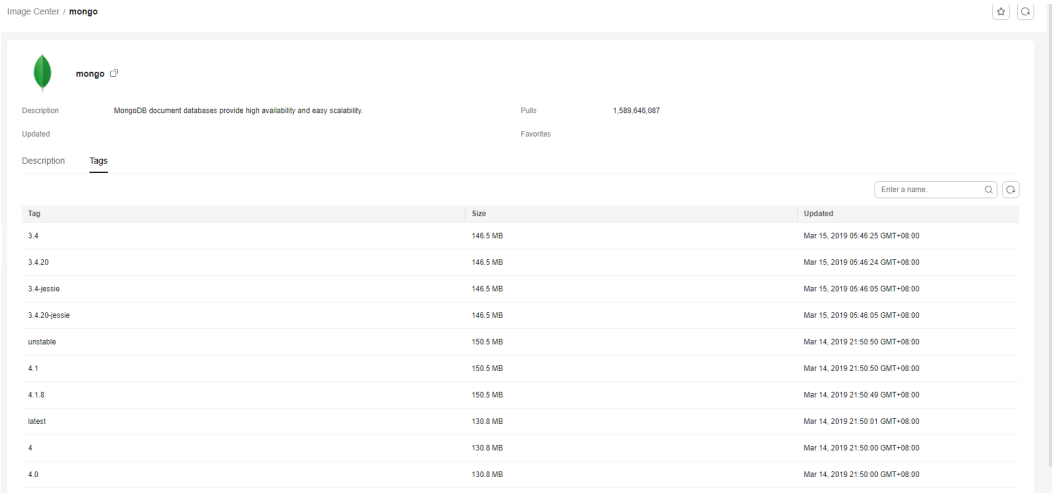
Pulling an Image from the Image Center

The following uses the **mongo** image in the Image Center as an example.

CAUTION

You need to ensure that your container engine client can access the Internet.

Figure 6-24 mongo image details



- If your container engine client node is in any of the following regions, you can run the command to download the image:

  CN-Hong Kong, AP-Bangkok, AP-Singapore, AF-Johannesburg, and LA-Santiago

  docker pull mongo:*4.1*

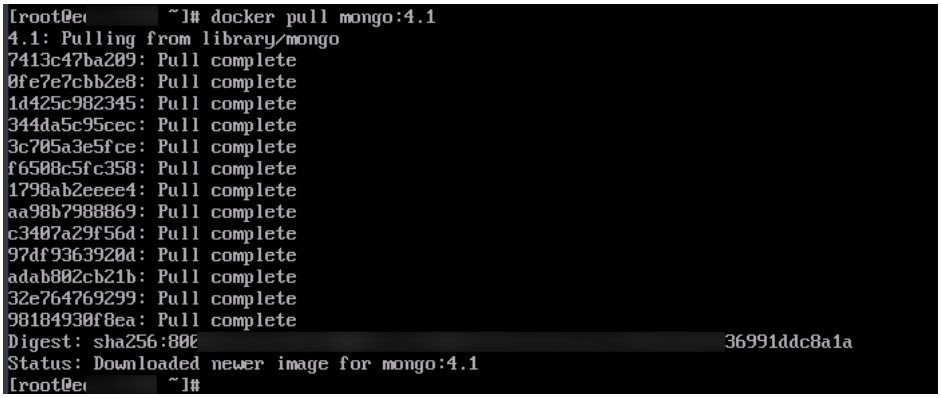
  ```
  [root@ec        ~]# docker pull mongo:4.1
  4.1: Pulling from library/mongo
  7413c47ba209: Pull complete
  0fe7e7cbb2e8: Pull complete
  1d425c982345: Pull complete
  344da5c95cec: Pull complete
  3c705a3e5fce: Pull complete
  f6508c5fc358: Pull complete
  1798ab2eeee4: Pull complete
  aa98b7988869: Pull complete
  c3407a29f56d: Pull complete
  97df9363920d: Pull complete
  adab802cb21b: Pull complete
  32e764769299: Pull complete
  98184930f8ea: Pull complete
  Digest: sha256:800                                                    36991ddc8a1a
  Status: Downloaded newer image for mongo:4.1
  [root@ec        ~]#
  ```

  NOTE

  The part following **docker pull** is *<image-name>:<image-tag>*. You can click next to the image name on the image details page to copy the name. To obtain the image tag, go to the **Tags** tab.

- If the node where the container engine client is deployed is in any of the following regions, you need to set an image accelerator before running the **docker pull** command:

CN North-Beijing1, CN North-Beijing4, CN East-Shanghai1, CN East-Shanghai2, CN South-Guangzhou, and CN Southwest-Guiyang1



6.13 Configuring a Pull Accelerator

Due to carrier network issues, pulling images from third-party image registry (such as Docker Hub) may take a long time or even fail. SWR provides image download acceleration to speed up the download of some common open-source images.

CAUTION

- The accelerator is only intended for individual developers and cannot be encapsulated for external use or used for commercial purposes.
- In a production environment, to prevent image pull failures caused by Docker network problems, you are advised to synchronize the required images from Docker Hub to SWR and then pull them from SWR.
- The image accelerator cannot pull images of all tags. Only common open-source images can be pulled using the image accelerator. You are advised to synchronize the required images to SWR.
- Pull acceleration is not available for containerd.

Constraints

- Only Huawei Cloud users can use image pull acceleration in Huawei Cloud container products.
- Only common open-source images can be pulled using the image accelerator. The acceleration may not work for all images. Use this feature with caution in production environments.
- If a Docker container engine client is used, the Docker version must be 18.06 or later.
- This feature is available in CN South-Guangzhou, CN North-Beijing4, CN East-Shanghai1, and CN Southwest-Guiyang1.

Procedure

Step 1 Log in to the **SWR console**.

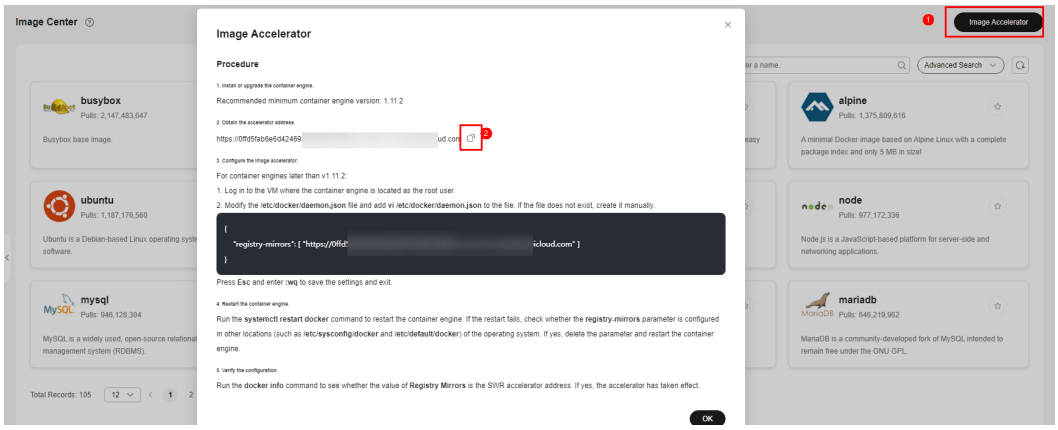
Step 2 In the navigation pane, choose **Image Resources** > **Image Center**.

NOTE

Ensure that the Image Center is available in the current region. For details, see .

Step 3 Click **Image Accelerator**. In the displayed dialog box, click to copy the accelerator address.

Figure 6-25 Image accelerator



Step 4 Log in to the server running the container engine as user **root**.

Step 5 Modify the **/etc/docker/daemon.json** file. If the file does not exist, manually create one. Add the following content to the file:

vi /etc/docker/daemon.json

```
{
"registry-mirrors":[accelerator-address]
}
```

Replace *[accelerator-address]* with the image accelerator address copied in **Step 3**.

Press **Esc** and enter **:wq** to save the settings and exit.

Step 6 Run **systemctl restart docker** to restart Docker.

If the restart fails, check whether the **registry-mirrors** parameter is set in other locations (such as **/etc/sysconfig/docker** or **/etc/default/docker**) of the operating system. If yes, delete the parameter and restart the container engine.

Step 7 To verify whether the SWR image accelerator has been successfully configured, run the **docker info** command to check whether the value of **Registry Mirrors** is the SWR accelerator address. If yes, the accelerator has taken effect.

Figure 6-26 Registry Mirrors information

```
Registry Mirrors:
 https://                                .mirror.swr.myhuaweicloud.com/
```

----End

7 Auditing

7.1 SWR Operations Supported by CTS

Scenarios

Cloud Trace Service (CTS) is a log audit service provided by Huawei Cloud and intended for cloud security. It allows you to collect, store, and query cloud resource operation records and use these records to analyze security, audit compliance, track resources, and locate faults.

With CTS, you can record operations related to SWR for future query, audit, and backtrack.

Key Operations Recorded by CTS

Table 7-1 SWR Basic Edition operations recorded by CTS

Operation	Resource Type	Trace Name
Pushing an image	images	postMultipartImagePackage
Obtaining a login command	dockerlogincmd	createDockerConfig
Querying the overview	overview	getDomainOverview
Query monitoring metrics	overview	getDomainResourceReports
Querying quotas	quota	listQuotas
Querying details about a trigger	trigger	showTrigger
Listing triggers	trigger	listTriggers
Creating a trigger	trigger	createTrigger

Operation	Resource Type	Trace Name
Updating a trigger	trigger	updateTrigger
Deleting a trigger	trigger	deleteTrigger
Creating an organization	usernamespace	createUserNamespace
Deleting an organization	usernamespace	deleteUserNamespace
Listing organizations	usernamespace	listUserNamespaces
Querying details about an organization	usernamespace	showUserNamespace
Creating an organization authorization	usernamespaceauth	createUserNamespaceAuth
Deleting an organization authorization	usernamespaceauth	deleteUserNamespaceAuth
Querying details about an organization authorization	usernamespaceauth	showUserNamespaceAuth
Updating an organization authorization	usernamespaceauth	updateUserNamespaceAuth
Creating a repository	imagerepository	createImageRepository
Deleting a repository	imagerepository	deleteImageRepository
Query details about a repository	imagerepository	showImageRepository
Updating a repository	imagerepository	updateImageRepository
Listing repositories	imagerepository	listImageRepositories
Creating an image tag	imagetag	createImageTag
Deleting an image tag	imagetag	deleteImageTag
Listing image tags	imagetag	listImageTags
Creating an image authorization	userrepositoryauth	createUserRepositoryAuth

Operation	Resource Type	Trace Name
Deleting an image authorization	userrepositoryauth	deleteUserRepositoryAuth
Querying details about an image authorization	userrepositoryauth	showUserRepositoryAuth
Updating an image authorization	userrepositoryauth	updateUserRepositoryAuth
Listing image authorizations	userrepositoryauth	listSharedReposDetails
Sharing an image with other accounts	imagerepositoryaccess-domain	createImageRepositoryAccess-Domain
Deleting an account from the sharing list	imagerepositoryaccess-domain	deleteImageRepositoryAccess-Domain
Querying details about an account that an image is shared with	imagerepositoryaccess-domain	showImageRepositoryAccess-Domain
Updating an account that an image is shared with	imagerepositoryaccess-domain	updateImageRepositoryAc-cessDomain
Listing the accounts an image is shared with	imagerepositoryaccess-domain	listImageRepositoryAccessDo-main
Sharing an image	reposhare	createRepoShare
Stopping sharing an image	reposhare	deleteRepoShare
Querying details about sharing an image	reposhare	getRepoShare
Updating details about sharing an image	reposhare	updateRepoShare
Viewing image sharing list	reposhare	listRepoShares
Creating an automatic image synchronization task	image-sync	setRImageSync
Deleting an automatic image synchronization task	image-sync	unsetRImageSync

Operation	Resource Type	Trace Name
Listing automatic image synchronization tasks	image-sync	listRImageSync
Manually synchronizing an image	image-sync	manualImageSync
Querying details about a manual image synchronization job	image-sync	showSyncJob
Creating an image retention policy	retention	createRetention
Deleting an image retention policy	retention	deleteRetention
Querying details about an image retention policy	retention	showRetention
Updating an image retention policy	retention	updateRetention
Listing image retention policies	retention	listRetentions
Listing image retention records	retention	listRetentionHistories
Creating an image layer	blob	createBlob
Updating an image layer by data chunk	blob	updateImageLayerChunk
Updating an image layer	blob	updateImageLayer
Pulling an image layer	blob	downloadImageLayer
Pushing image manifest	manifest	uploadManifest

7.2 Viewing Logs in CTS

Scenarios

After you enable CTS, the system starts recording operations performed on SWR resources. CTS stores operation records generated within a week.

This section describes how to view the records on the CTS console.

Procedure

Step 1 Log in to the CTS console. In the upper right corner, click **Go to Old Edition**.

Step 2 In the navigation pane, choose **Trace List**.

Step 3 Set the filter criteria and click **Query**.

The following filters are available:

- **Trace Type**, **Trace Source**, **Resource Type**, and **Search By**

  Select the desired filter criteria from the drop-down lists, and set **Trace Type** to **Management** and **Trace Source** to **SWR**.

  If you set **Search By** to **Resource ID**, you need to enter a resource ID. Only whole word match is supported.

- **Operator**: Select a specific operator from the drop-down list.
- **Trace Status**: Select **All trace statuses**, **Normal**, **Warning**, or **Incident**.
- Time range: You can select **Last 1 hour**, **Last 1 day**, **Last 1 week**, or **Customize** in the upper right corner.

Step 4 Locate a record and click ⌄ to view its details.

Step 5 Click **View Trace** in the **Operation** column. The trace structure details are displayed.

----End

