CodeArts Governance

FAQs

Issue 01

Date 2025-06-05

HUAWEI

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 Binary Software Composition Analysis (SCA)

1.1 What Objects Can Be Scanned?

- Binary software packages and firmware that have been compiled can be scanned.

  For example, Linux installation packages, Windows installation packages, web deployment packages, Android applications, HarmonyOS applications, iOS applications, and embedded firmware.
- Note that source code files cannot be scanned.

1.2 What Are the Precautions of Binary SCA?

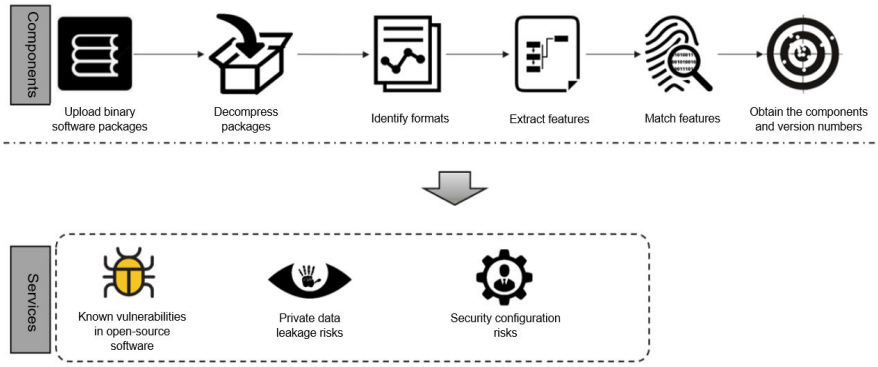
- Files programmed in the following languages can be uploaded: C, C++, Java, Go, JavaScript, Python, Rust, Swift, C#, and PHP
- Files in .7z, .arj, .cpio, .phar, .rar, .tar, .xar, .zip, .jar, .apk, .war, .rpm, and .deb formats and firmware such as Android OTA Images, Android sparse, Intel HEX, RockChip, and U-Boot can be uploaded.
- Each file you upload cannot exceed five GB.
- It usually takes six minutes to scan a 100-MB file. The scanning duration depends on the format and type of the file.
- Binary SCA scans software vulnerabilities by version. Even if vulnerabilities are fixed with patches, they will still be detected.

1.3 How Does Binary SCA Work and What Risks Can be Identified?

CodeArts Governance decompresses and scans your software packages and firmware. It performs component feature analysis based on the bill of materials (BOM) to identify possible rule violations. The following lists the vulnerabilities that can be identified.

- **Open source software's** known vulnerabilities and license compliance risks.
- **Security configuration risks** in hard-coded credentials, sensitive files (keys, certificate, and debugging tools), OS authentication, and access control.
- **Disclosure risks** of IP addresses, hard-coded keys, passwords, and Git/SVN repositories.
- **Compiler security option risks** in binary program compilation.

Figure 1-1 Risk items

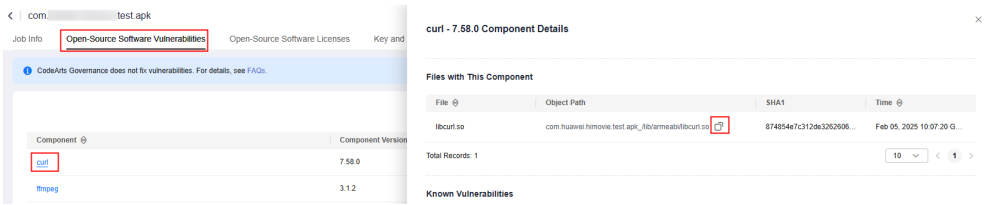


1.4 How Do I Handle Open-Source Software Vulnerabilities?

CodeArts Governance decompresses and scans your software packages and firmware. It performs static analysis based on the software BOM to identify vulnerabilities and license risks. After the scan is complete, you can perform the following operations to handle the vulnerabilities.

1. Check the **Open Source Vulnerability Analysis** tab page.

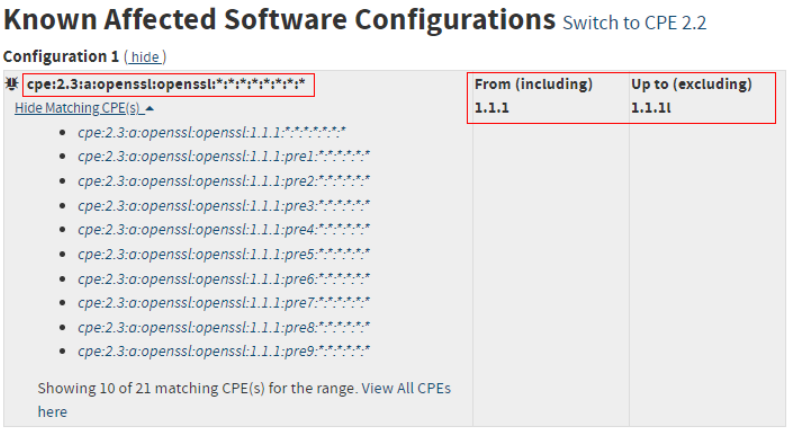
   Locate the files according to the file path displayed on the report details page or in the report. If the detected software does not exist or the software version number is incorrect, no further analysis is required.



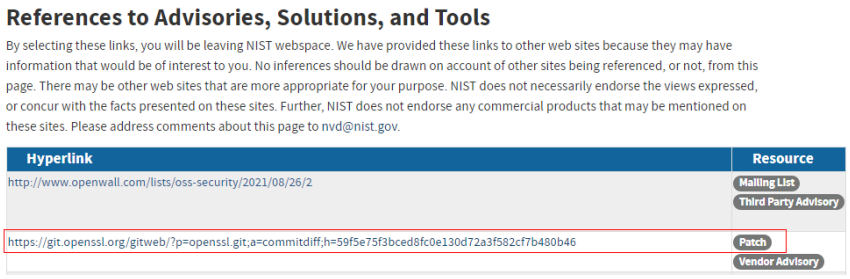
2. Check the **Known vulnerabilities** area.

   Search for the CVE number in National Vulnerability Database (NVD), Common Vulnerabilities and Exposures (CVE), and China National Vulnerability Database (CNVD) to check the vulnerability details.

   – **General analysis:** Confirm the scope of each vulnerability. For example, **CVE-2021-3711** is listed in the **Known Affected Software Configurations** area by NVD, as the following figure shows. You can check whether the vulnerability affects the software version that you are using.



- **Refined analysis:** Since some vulnerabilities exist in functions, you can refer to the patches listed in the communities mentioned above to learn the vulnerability details, affected functions, and fixing methods, as shown in the following figure, and confirm whether the vulnerability affects the software that you are using.



3. Check the license compliance. Identify the risky licenses and confirm whether they can meet your service requirements.
4. Handle the vulnerabilities according to the following instructions.
   - **Known vulnerabilities:** Upgrade the software to the version recommended by the communities mentioned above. Alternatively, install patches to fix the vulnerabilities temporarily.
   - **Risky licenses:** Replace the risky licenses with compliant ones that function the same.

1.5 How Do I Handle Secure Complier Option Vulnerabilities?

CodeArts Governance checks the C, C++, and Go files to see whether there are secure compiler options to defend against attacks.

Handle the secure compiler option issues according to the following instructions.

1. Export the Excel report and view the secure compiler option sheet.
2. Obtain the file source according to the **filepath** column.
3. View the check items and handle the vulnerabilities accordingly.
   - If a check item is passed, its result is in green and it does not require further actions. For files passed **Rpath**, their results are **No** or **N/A**. For other passed check items, their results are **YES** or **N/A**.

– For files failed the check items, obtain their building scripts, and add the secure compiler options accordingly. Note that **Ftrapv** and **FS** may affect the files and you can add them based on your needs.

Table 1-1 Secure complier options

Item	Description	Parameter
BIND_NOW	Immediate binding	-Wl, -z, now
NX	Non-executable stack	-WI, -z, noexecstack
PIC	Position-independent	-fPIC
PIE	Position-independent executable	-fPIE or -pie
Relocation read-only (RELRO)	Global Offset Table (GOT) protection	-WI, -z, relro
SP	Stack protection	-fstack-protector-strong or -fstack-protector-all
NO Rpath/Runpath	Dynamic library search path (forbidden)	Delete **--rpath** from the script.
FS	Fortify Source (buffer overflow check)	-D_FORTIFY_SOURCE=2
Ftrapv	Integer overflow check	-ftrapv
Strip	Symbol table deletion	-s

1.6 How Do I Handle Security Configuration Issues?

CodeArts Governance checks whether the security configuration items are compliant. Specifically, it detects the following issues in the uploaded software packages or firmware.

- Sensitive information, for example, key files, certificate files, source code files, and debugging tools
- Problems in the user and group configurations, hard-coded credentials, and authorization and access control. Note that some check items apply to operation systems (OSs) only.

Handle the security configuration issues according to the following instructions.

Export the PDF report and search for security configuration issues. The result of each check item can be **pass**, **failed**, or **NA** (for software packages or firmware that does not include OSs).

The report includes the following items:

- Check items, or, the check methods
- Files that have issues, if any

- Recommended fix suggestions
- Description of check items

1.7 How Do I Handle Information Leakage Risks?

CodeArts Governance decompresses and scans your software packages and firmware. It identifies information leakage risks, for example, sensitive IPs, Git/SVN repositories, weak passwords, and hard-coded keys.

By exporting reports, for example, PDF reports, you can view the risk details. The following information will be displayed for each risk, you can determine whether to mask or fix them based on your service requirements.

- Risk type: For example, IP address leakage, hard-coded passwords, or Git address leakage
- File path: Full path of the file where information leakage is detected in the package
- Context: Risky lines and context lines
- Content: Contents that are risky
- Location: The exact lines and places of the risks

1.8 Why Is the Component Version Not Identified or Incorrectly Identified?

The possible reasons include:

1. The feature library does not cover the version of the open-source software.
2. The source code of the open-source software has been modified, or only a few functions of the open-source software are referenced. As a result, the software does not have enough features in the compiled or released file for CodeArts Governance to scan.
3. The open-source software contains dependent software. Only a few functions in the dependent software are used. As a result, CodeArts Governance cannot detect the software or version correctly.

1.9 Why Can't I Buy a CodeArts Governance Package?

Check your permissions.

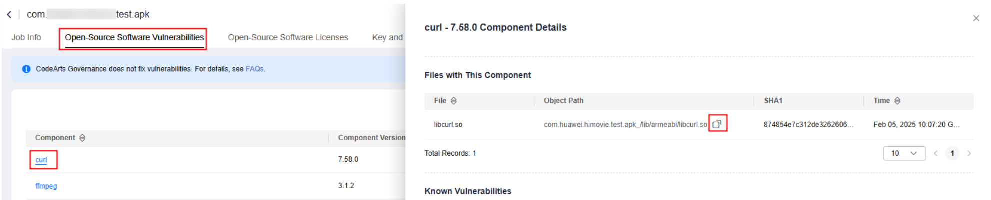
To buy CodeArts Governance packages, you must have the **te_admin**, **bss_adm**, **bss_pay**, or **bss_ops** permission.

Otherwise, contact a user with the Tenant Administrator permission to obtain the permission.

1.10 How Do I View the Path of a File that Has Vulnerabilities?

You can view the patch in one of the following ways.

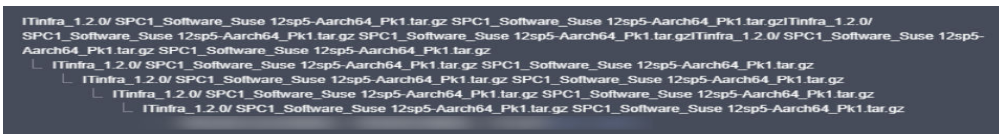
- Method 1: Go to the report details page, click the **Open-Source Software Vulnerabilities** tab page, and click a component name. In the displayed drawer, hover the cursor over the path of the component to check it. You can also click the button on the right to copy the path.



- Method 2: Open the report details page and choose **Download Report** > **PDF Report**. After the file is generated, choose **Download Report** > **Export to PDF**. You can view the file path of the corresponding component by referring to chapter 3 in the PDF report.
- Method 3: On the report details page and choose **Download Report** > **Excel Report**. After the file is generated, choose **Download Report** > **Export to Excel**. Go to the component report or vulnerability report sheet in the Excel to view the file path of the corresponding component.

  Directories at each level are separated by a slash (/). If a directory has been scanned, an underscore and a slash (_/) will be displayed. For example, this file path is displayed: **scrm-service-weixin.jar_/BOOT-INF/classes/libWeWorkFinanceSdk_Java.so**. In this case, decompress **scrm-service-weixin.jar** and view the **BOOT-INF/classes/libWeWorkFinanceSdk_Java.so** and to check whether the open-source software exists or whether it is correct.

  CodeArts Governance will also check the files in subdirectories. Method 1 shows subdirectories by level with indentation at the beginning of each line. For methods 2 and 3, subdirectories are separated by colons (:).



1.11 What Can I Do If a Binary SCA Task Fails?

The following table lists some possible reasons that may lead a binary SCA task to fail.

Table 1-2 Possible causes

Cause	Solution
File parsing failed	The file is incomplete or the structure is abnormal. As a result, it cannot be parsed. In this case, upload a valid package or firmware and start a new task.
File damaged during upload	The file is damaged during upload. As a result, the file cannot be parsed. In this case, start a new task.
Other reasons	If the task keeps failing after multiple retries, you need to contact the service O&M team.

1.12 How Do I Check User Group Permissions and Grant Permissions?

1. Log in to Huawei Cloud and click **Console** in the upper right corner.
2. Hover the mouse over the username in the upper right corner, and select **Identity and Access Management** from the drop-down list.
3. Click **User Groups** and click a group name to check authorization records.
4. Click **Authorize** to grant permissions to the user group.
5. Search for **Tenant Administrator** or **CodeArtsInspector Administrator** and select required policies.
6. Click **Next**, select the authorization scope, and click **OK**.

1.13 What Should I Do If a Role Permission Error (Roles with READONLY_USER) Is Reported?

To perform binary SCA, you must have the Tenant Administrator or CodeArtsInspector Administrator permission. Check users who have such permission by referring to **How Do I Check User Group Permissions and Grant Permissions?** Ask them to grant you the permission under the following instructions.

1. Log in to Huawei Cloud as a user with the Tenant Administrator or CodeArtsInspector Administrator permission, and click **Console** in the upper right corner.
2. Hover the mouse over the username in the upper right corner, and select **Identity and Access Management** from the drop-down list.
3. On the **Users** page, click **Authorize** in the **Operation** column of the row that contains the target user.
4. Select **Inherit permissions from user groups** for **Authorization Method**, then select user groups with the Tenant Administrator or CodeArtsInspector Administrator permission.

5. Click **OK**.