CodeArts Build

Service Overview

Issue 01

Date 2023-11-15

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 What Is CodeArts Build?.......... 1

2 Advantages.......... 2

3 Use Cases.......... 3

4 Build Environments.......... 5

5 Security.......... 22

5.1 Shared Responsibilities.......... 22

5.2 Authentication and Access Control.......... 23

5.3 Data Protection Technologies.......... 24

5.4 Cloud Trace Service (CTS).......... 24

5.5 Service Resilience.......... 25

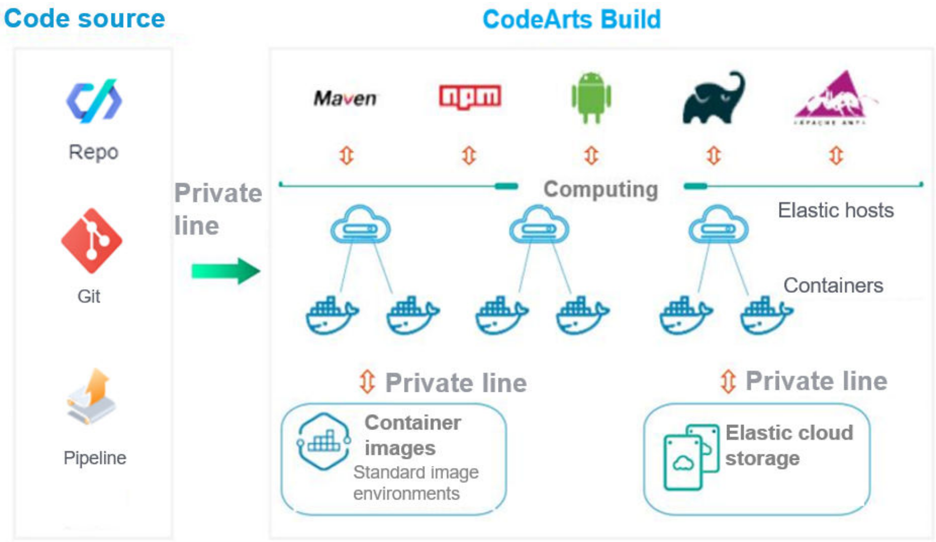
5.6 Certificates.......... 25

6 Constraints and Restrictions.......... 27

1 What Is CodeArts Build?

Building entails compiling source code into one or more target files, and packaging these target files along with configuration and resource files.

CodeArts Build provides an easy-to-use, cloud-based build platform that supports multiple programming languages, helping you achieve continuous delivery with higher efficiency. With CodeArts Build, you can create, configure, and run build tasks with a few clicks. CodeArts Build also supports automated code retrieval, build, and packaging, as well as real-time status monitoring.



Container-based build

CodeArts Build provides container-based build environments and supports two types of container images:

- System images: provided by CodeArts Build for compilation and packaging with popular programming languages.
- Custom images: customized for compilation and packaging with multiple languages.

2 Advantages

All-scenario

CodeArts Build supports multiple programming languages and frameworks to suit popular software development scenarios.

- Supported languages: C, C++, C#, Java, Python, JavaScript, Go, PHP, .NET, and Groovy.
- Supported frameworks: Maven, Gradle, Ant, npm, CMake and Android. For details, see **Build Environment**.

Fast

Massive build resources are available on the cloud, and various cloud-hosted build acceleration methods are used to achieve a speed that is impossible for local builds.

- Elastic resources in the cloud and task execution in parallel
- Global- and tenant-level cache
- Direct Connect for faster transmission

Scalable

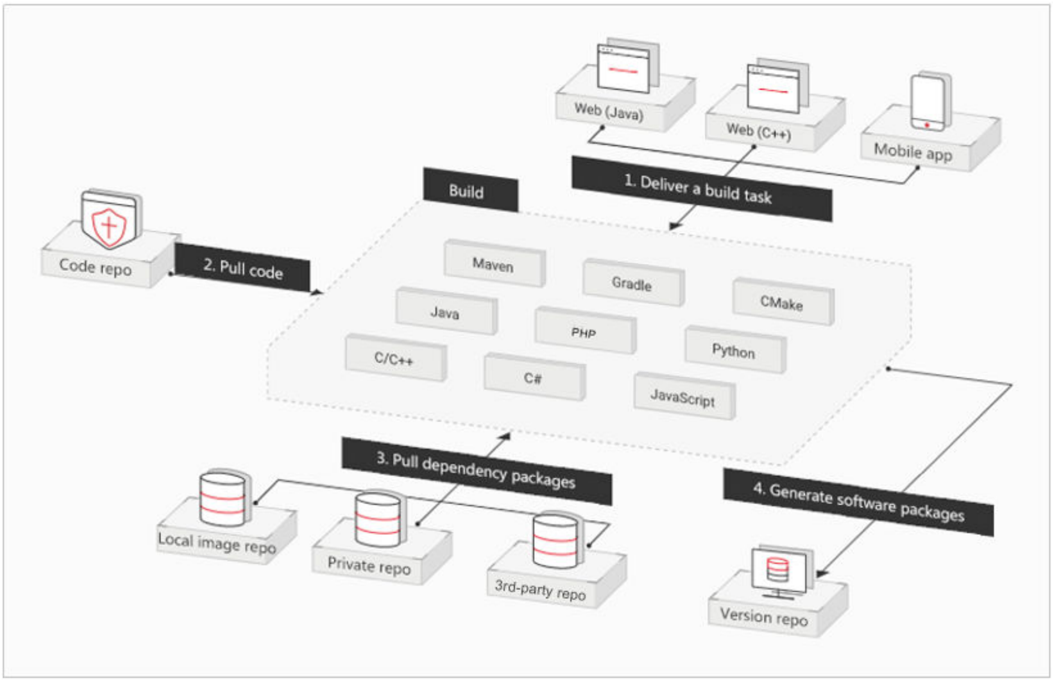
CodeArts Build can flexibly interconnect with different code hosting services and provides the following features:

- Customized build environments
- Diverse source code hosting services
- Continuous integration clusters

Cost-effective

- You are billed for the actual resource usage and duration, so CodeArts Build eliminates the need to invest in hardware resources and maintenance, greatly reducing build costs.
- Servers are centrally maintained by CodeArts, greatly reducing maintenance costs.

3 Use Cases



Internet Web Applications

- Requirements and challenges: As the number of Internet service types increases, cross-platform services and multi-language programming have become a trend.
- Benefits: CodeArts Build can be used to build frontend and backend programs of web applications. It supports languages and frameworks such as Java, Angular, and Node.js, and build standards such as Maven and Ant.

Computer Applications

- Requirements and challenges: Legacy computer applications run locally at a large scale. Services are complex, and the build process is time-consuming.
- Benefits: On-demand resource allocation speeds up the build. Using CodeArts Build, you can build C and C++ applications in Linux and C, C++, and C# applications in Windows.

Mobile Apps

- Requirements and challenges: Services on mobile apps change frequently, requiring quick and smooth delivery.
- Benefits: Cloud-based parallel compilation technology is used to shorten the delivery period. Using CodeArts Build, you can quickly build Android mobile apps.

4 Build Environments

This section describes the basic environments (images and tool versions) for each build action.

Building with Maven

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Tool
maven3.9.5-jdk17	Maven 3.9.5, OpenJDK 17, Zip 3.0, UnZip 6.0, curl 7.29.0, and Wget 1.14
maven3.8.5-jdk17	Maven 3.8.5, OpenJDK 17, Zip 3.0, UnZip 6.0, curl 7.29.0, and Wget 1.14
maven3.6.3-jdk11	Maven 3.6.3, OpenJDK 11, Zip 3.0, UnZip 6.0, curl 7.29.0, and Wget 1.14
maven3.6.1-jdk10	Maven 3.6.1, OpenJDK 10.0.2, Zip 3.0, UnZip 6.0, curl 7.29.0, and Wget 1.14
maven3.5.3-jdk8-open	Maven 3.5.3, OpenJDK 1.8.0_40, curl 7.29.0, and Wget 1.14
maven3.5.3-jdk7	Maven 3.5.3, OpenJDK 1.7.0_75, Zip 3.0, and UnZip 6.0

Building with Android

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Build Environment	Built-in Tool
Basic environment	
Android	Git, Zip 3.0, UnZip 6.0, curl 7.29.0, and Wget 1.14
Optional environments	
Gradle	Gradle 2.13, 2.14.1, 3.3, 4.1–5.6, 6.0–7.3, 8.0, and 8.2.1
JDK	OpenJDK 1.7–1.9 and OpenJDK 10–16
NDK	NDK 16–23, and NDK 25

Building with npm

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Image Tool
Node.js 8.11.2	• Node v8.11.2, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 • curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 • libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.1.0	• Node v10.1.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 • make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 • autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.15.3	• Node v10.15.3, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 • make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 • autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Tool Version	Built-in Image Tool
Node.js 12.7.0	● Node v12.7.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 13	● Node v13, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 14	● Node v14, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 15	● Node v15, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 16	● Node v16, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 18	● Node v18, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 20	● Node v20, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Building with Yarn

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Image Tool
Node.js 8.11.2	● Node v8.11.2, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.1.0	● Node v10.1.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.15.3	● Node v10.15.3, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 12.7.0	● Node v12.7.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 13	● Node v13, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 14	● Node v14, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Tool Version	Built-in Image Tool
Node.js 15	● Node v15, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 16	● Node v16, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 18	● Node v18, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 20	● Node v20, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Building with Gradle

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Build Environment	Built-in Tool
Basic environment	Git, Zip 3.0, UnZip 6.0, curl 7.29.0, and Wget 1.14
Optional environments	
Gradle	Gradle 2.13, 2.14.1, 3.3, 4.1–5.6, 6.0–7.3, 8.0, and 8.2.1

Build Environment	Built-in Tool
JDK	OpenJDK 1.7, OpenJDK 1.8, OpenJDK 1.9, OpenJDK 10, OpenJDK 11, OpenJDK 12, OpenJDK 13, OpenJDK 14, OpenJDK 15, OpenJDK 16, and OpenJDK 17

Building with Mono

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Description
mono6-msbuild16-dotnetcoresdk6.0	Preinstalled with common tools such as MSBuild 16.1.76, NuGet, and .NET Framework 4.8. Select this image when the project uses .NET Framework 4.0 or later or .NET Core 6.0 or later. (This image is compatible with historical tasks but not compatible with the MSBuild-all image.)
mono6-msbuild16-dotnetcoresdk5.0	Preinstalled with common tools such as MSBuild 16.1.76, NuGet, and .NET Framework 4.8. Select this image when the project uses .NET Framework 4.0 or later or .NET Core 5.0 or later. (This image is compatible with historical tasks but not compatible with the MSBuild-all image.)
mono6-msbuild16-dotnetcoresdk3.1	Preinstalled with common tools such as MSBuild 16.1.76, NuGet, and .NET Framework 4.8. Select this image when the project uses .NET Framework 4.0 or later or .NET Core 3.1 or later. (This image is compatible with historical tasks but not compatible with the MSBuild-all image.)
mono6-msbuild16-dotnetcoresdk3.0	Preinstalled with common tools such as MSBuild 16.1.76, NuGet, and .NET Framework 4.8. Select this image when the project uses .NET Framework 4.0 or later or .NET Core 3.0 or later. (This image is compatible with historical tasks but not compatible with the MSBuild-all image.)
mono6-msbuild16-dotnetcoresdk2.2	Preinstalled with common tools such as MSBuild 16.1.76, NuGet, and .NET Framework 4.8. Select this image when the project uses .NET Framework 4.0 or later or .NET Core 2.2 or later. (This image is compatible with historical tasks but not compatible with the MSBuild-all image.)
mono6-msbuild16-dotnetcoresdk2.1	Preinstalled with common tools such as MSBuild 16.1.76, NuGet, and .NET Framework 4.8. Select this image when the project uses .NET Framework 4.0 or later or .NET Core 2.1 or later. (This image is compatible with historical tasks but not compatible with the MSBuild-all image.)

Building with Grunt

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Image Tool
Node.js 8.11.2	● Node v8.11.2, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.1.0	● Node v10.1.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.15.3	● Node v10.15.3, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 12.7.0	● Node v12.7.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 13	● Node v13, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 14	● Node v14, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Tool Version	Built-in Image Tool
Node.js 15	- Node v15, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 - curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 - libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 16	- Node v16, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 - curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 - libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 18	- Node v18, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 - curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 - libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 20	- Node v20, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 - curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 - libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Building with Setuptools/PyInstaller

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Tool
Python 2.7	- Python 2.7, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 - make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 - automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k

Tool Version	Built-in Tool
Python 3.5	● Python 3.5, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 ● automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k
Python 3.6	● Python 3.6, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 ● automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k
Python 3.7	● Python 3.7, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 ● automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k
Python 3.8	● Python 3.8, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 ● automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k
Python 3.9	● Python 3.9, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 ● automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k
Python3.10	● Python 3.10, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 ● automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k

Tool Version	Built-in Tool
Python3.11	- Python 3.11, Setuptools 39.1.0, pip 10.0.1, PyInstaller, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 - make 3.82, GCC 4.8.5, GCC-C++ 4.8.5, libgcc.x86_64 4.8.5, libgcc.i686 4.8.5, and libtool 2.4.2 - automake 1.13.4, autoconf 2.69, zlib 1.2.7, zlib-devel 1.2.7, and openssl-devel 1.0.2k

Running Shell Commands

Use the general image **shell4.2.46-git1.8.3-zip6.00** to run the shell commands. The image is based on EulerOS. By default, Zip, UnZip, sudo, Git, and Wget are installed.

Image	Built-in Image Tool
shell4.2.46-git1.8.3-zip6.00	Git, Wget 1.14, Zip 3.0, UnZip 6.0, and sudo 1.8.19p2

Building with Gulp

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Image Tool
Node.js 8.11.2	- Node v8.11.2, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 - curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 - libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.1.0	- Node v10.1.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 - make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 - autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Tool Version	Built-in Image Tool
Node.js 10.15.3	● Node v10.15.3, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 12.7.0	● Node v12.7.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 13	● Node v13, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 14	● Node v14, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 15	● Node v15, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 16	● Node v16, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 18	● Node v18, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Tool Version	Built-in Image Tool
Node.js 20	● Node v20, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Building with GNU Arm

The EulerOS base image is used. Software such as make, GCC, G++, 32-bit C runtime library, and binary tool library is pre-installed. The following tool versions are provided:

- gnuarm201405

  The Arm toolchain developed by CodeSourcery based on GCC is installed. This toolchain can be used to cross-compile all code in the Arm system, including the bare-metal program, U-Boot, Linux kernel, file system, and application program.

- gnuarm-linux-gcc-4.4.3

  This image complies with the EABI standard and is a cross-compiler for earlier Linux versions.

- gnuarm-7-2018-q2-update

  It has Arm embedded GCC compilers, libraries, and other GNU tools required for bare metal software development on devices based on Arm Cortex-M and Cortex-R processors.

Tool Version	Built-in Tool
gnuarm201405	arm-2014.05-29-arm-none-linux-gnueabi-i686-pc-linux-gnu, Git, Wget 1.14, Zip 3.0, Unzip 6.0, GCC-C++ 4.8.5, bzip2 1.0.6, file 5.11, glibc.i686 2.17, ncurses-devel.i686 2.9, binutils-devel.i686 2.27, and zlib 1.2.7
gnuarm-linux-gcc-4.4.3	arm-linux-gcc-4.4.3-20100728, Git, Wget 1.14, Zip 3.0, Unzip 6.0, GCC-C++ 4.8.5, bzip2 1.0.6, file 5.11, glibc.i686 2.17, ncurses-devel.i686 5.9, binutils-devel.i686 2.27, zlib 1.2.7, and gzip 1.5
gnuarm-7-2018-q2-update	gcc-arm-none-eabi-7-2018-q2-update-linux2, Git, Wget 1.14, Zip 3.0, Unzip 6.0, GCC-C++ 4.8.5, bzip2 1.0.6, file 5.11, glibc.i686 2.17, ncurses-devel.i686 5.9, binutils-devel.i686 2.27, zlib 1.2.7, and gzip 1.5

Building with CMake

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Image Tool
cmake3.16.5-gcc7.3.0	● CMake 3.16.5, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 7.3.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, zlib-devel 1.2.7, libgcc.i686 4.8.5, and libgcc.x86_64 4.8.5
cmake3.16.0-gcc9.2.0	● CMake 3.16.0, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 9.2.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, zlib-devel 1.2.7, libgcc.i686 4.8.5, and libgcc.x86_64 4.8.5
cmake3.15.5-gcc8.3.0	● CMake 3.15.5, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 8.3.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, zlib-devel 1.2.7, libgcc.i686 4.8.5, and libgcc.x86_64 4.8.5
cmake2.8.12-gcc4.8.5	● CMake 2.8.12, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 4.8.5, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, zlib-devel 1.2.7, libgcc.i686 4.8.5, and libgcc.x86_64 4.8.5
cmake2.8.12-gcc5.5.0	● CMake 2.8.12, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 5.5.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, and zlib-devel 1.2.7 ● libgcc.i686 4.8.5 and libgcc.x86_64 4.8.5
cmake2.8.12-gcc6.4.0	● CMake 2.8.12, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 6.4.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, and zlib-devel 1.2.7 ● libgcc.i686 4.8.5 and libgcc.x86_64 4.8.5
cmake2.8.12-gcc7.3.0	● CMake 2.8.12, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 7.3.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, and zlib-devel 1.2.7 ● libgcc.i686 4.8.5 and libgcc.x86_64 4.8.5
cmake3.10.1-gcc4.8.5	● CMake 3.10.1, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 ● GCC 4.8.5, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, zlib-devel 1.2.7, libgcc.i686 4.8.5, and libgcc.x86_64 4.8.5

Tool Version	Built-in Image Tool
cmake3.10.1-gcc5.5.0	- CMake 3.10.1, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 - GCC 5.5.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, and zlib-devel 1.2.7 - libgcc.i686 4.8.5, libgcc.x86_64 4.8.5, isl 0.15, mpfr 2.4.2, gmp 4.3.2, and mpc 0.8.1
cmake3.10.1-gcc6.4.0	- CMake 3.10.1, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 - GCC 6.4.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, and zlib-devel 1.2.7 - libgcc.i686 4.8.5, libgcc.x86_64 4.8.5, isl 0.15, mpfr 2.4.2, gmp 4.3.2, and mpc 0.8.1
cmake3.10.1-gcc7.3.0	- CMake 3.10.1, OpenJDK 1.8.0_191, Git, Wget 1.14, bzip2 1.0.6, and make 3.82 - GCC 7.3.0, GCC-C++ 4.8.5, libstdc++-devel 4.8.5, and zlib-devel 1.2.7 - libgcc.i686 4.8.5, libgcc.x86_64 4.8.5, isl 0.16.1, mpfr 3.1.4, gmp 6.1.0, and mpc 1.0.3

Building with Ant

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Tool
ant1.9.4-jdk1.8	Ant 1.9.4, OpenJDK 1.8.0_40, Git, Wget 1.14, and bzip2 1.0.6
ant1.10.1-jdk1.8	Ant 1.10.1, OpenJDK 1.8.0_40, Git, Wget 1.14, and bzip2 1.0.6
ant1.10.3-jdk1.8	Ant 1.10.3, OpenJDK 1.8.0_40, Git, Wget 1.14, and bzip2 1.0.6
ant1.10.13-jdk1.8	Ant 1.10.13, OpenJDK 1.8.0_40, Git, Wget 1.14, and bzip2 1.0.6

Building with Go

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Tool
Go 1.10.3	Go 1.10.3, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.11.6	Go 1.11.6, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.12.1	Go 1.12.1, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.13.1	Go 1.13.1, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.14	Go 1.14, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.15	Go 1.15, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.16	Go 1.16, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.17	Go 1.17, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.18	Go 1.18, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.19	Go 1.19, Git, GCC 4.8.5, and GCC-C++ 4.8.5
Go 1.20	Go 1.20, Git, GCC 4.8.5, and GCC-C++ 4.8.5

Building Android Quick App

The EulerOS base image is used. Different versions of build environments are provided in the following table.

Tool Version	Built-in Image Tool
Node.js 8.11.2	● Node v8.11.2, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 10.1.0	● Node v10.1.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Tool Version	Built-in Image Tool
Node.js 10.15.3	● Node v10.15.3, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 12.7.0	● Node v12.7.0, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, and UnZip 6.0 ● make 3.82, curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, and automake 1.13.4 ● autoconf 2.69, libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 13	● Node v13, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 14	● Node v14, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 15	● Node v15, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 16	● Node v16, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0
Node.js 18	● Node v18, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

Tool Version	Built-in Image Tool
Node.js 20	● Node v20, OpenJDK 1.8.0_40, Git, Wget 1.14, Zip 3.0, UnZip 6.0, and make 3.82 ● curl 7.29.0, GCC 4.8.5, GCC-C++ 4.8.5, Ant 1.9.4, automake 1.13.4, and autoconf 2.69 ● libtool 2.4.2, python-devel 2.7.5, and kernel-headers 3.10.0

5 Security

5.1 Shared Responsibilities

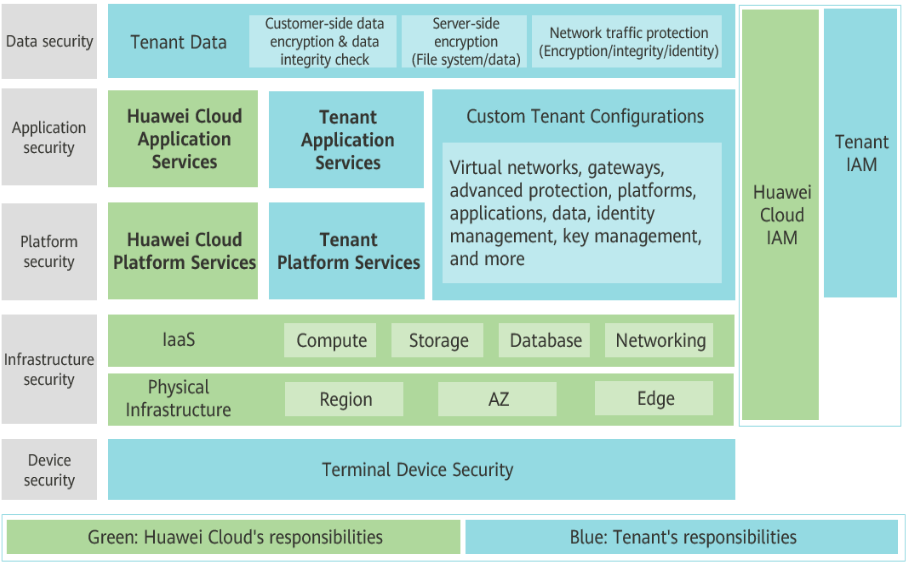
Huawei guarantees that its commitment to cyber security will never be outweighed by the consideration of commercial interests. To cope with emerging cloud security challenges and pervasive cloud security threats and attacks, Huawei Cloud builds a comprehensive cloud service security assurance system for different regions and industries based on Huawei's unique software and hardware advantages, laws, regulations, industry standards, and security ecosystem.

Figure 5-1 illustrates the responsibilities shared by Huawei Cloud and users.

- **Huawei Cloud**: Ensure the security of cloud services and provide secure clouds. Huawei Cloud's security responsibilities include ensuring the security of our IaaS, PaaS, and SaaS services, as well as the physical environments of the Huawei Cloud data centers where our IaaS, PaaS, and SaaS services operate. Huawei Cloud is responsible for not only the security functions and performance of our infrastructure, cloud services, and technologies, but also for the overall cloud O&M security and, in the broader sense, the security and compliance of our infrastructure and services.
- **Tenant**: Use the cloud securely. Tenants of Huawei Cloud are responsible for the secure and effective management of the tenant-customized configurations of cloud services including IaaS, PaaS, and SaaS. This includes but is not limited to virtual networks, the OS of virtual machine hosts and guests, virtual firewalls, API Gateway, advanced security services, all types of cloud services, tenant data, identity accounts, and key management.

Huawei Cloud Security White Paper elaborates on the ideas and measures for building Huawei Cloud security, including cloud security strategies, the shared responsibility model, compliance and privacy, security organizations and personnel, infrastructure security, tenant service and security, engineering security, O&M security, and ecosystem security.

Figure 5-1 Huawei Cloud shared security responsibility model


Data security
Tenant Data
Customer-side data encryption & data integrity check
Server-side encryption (File system/data)
Network traffic protection (Encryption/integrity/identity)
Application security
Huawei Cloud Application Services
Tenant Application Services
Custom Tenant Configurations
Virtual networks, gateways, advanced protection, platforms, applications, data, identity management, key management, and more
Platform security
Huawei Cloud Platform Services
Tenant Platform Services
Infrastructure security
IaaS
Compute
Storage
Database
Networking
Physical Infrastructure
Region
AZ
Edge
Huawei Cloud IAM
Tenant IAM
Device security
Terminal Device Security
Green: Huawei Cloud's responsibilities
Blue: Tenant's responsibilities

5.2 Authentication and Access Control

Identity Authentication

You can access CodeArts Build using its UI and APIs. Regardless of the access mode, your requests are sent through REST **APIs** provided by CodeArts Build.

CodeArts Build APIs can be accessed only after requests are authenticated. CodeArts Build supports two authentication modes:

- Token: Requests are authenticated using tokens. By default, token authentication is required to access the CodeArts Build console.
- AK/SK: Requests are encrypted using an AK/SK. This method is recommended because it provides higher security than token-based authentication. For operation details, see **AK/SK Signing and Authentication Guide**.
- For more authentication details and how to obtain tokens and signatures, see **Authentication**.

Access Control

CodeArts Build supports access control through IAM permissions.

Table 5-1 CodeArts Build access control

Method		Description	Reference
Permission management	IAM permissions	IAM permissions define which actions on your cloud resources are allowed and which actions are denied, to control access to your resources. By default, new IAM users do not have any permissions assigned. New users must be added to one or more groups, and permissions policies or roles must be attached to these groups.	**IAM Service Overview** **Permission Description**

5.3 Data Protection Technologies

CodeArts Build takes different methods and features to keep data secure and reliable.

Table 5-2 CodeArts Build data protection methods and features

Method	Description	Reference
Transmission encryption (HTTPS)	All CodeArts Build APIs use HTTPS for transmission.	**Making an API Request**
Personal data protection	CodeArts Build controls access to data and records logs by Cloud Trace Service (CTS) for operations performed on the data.	**Operations Recorded by CTS**
Privacy protection	CodeArts Build encrypts sensitive data such as database account information of users before storing it, supports encryption key rotation.	-
Data clearing	Sensitive data is deleted immediately after builds are complete.	-
Data backup	CodeArts Build supports user data backup.	-

5.4 Cloud Trace Service (CTS)

Cloud Trace Service (CTS) records operations on the cloud resources in your account. You can use the logs generated by CTS to perform security analysis, track resource changes, audit compliance, and locate faults.

After you enable CTS and configure a tracker, CTS can record management and data traces of CodeArts Build for auditing.

For details about how to enable and configure CTS, see **Enabling CTS**.

For details about CodeArts Build operations that can be traced, see **Operations Recorded by CTS**.

5.5 Service Resilience

Cross-AZ DR Deployment

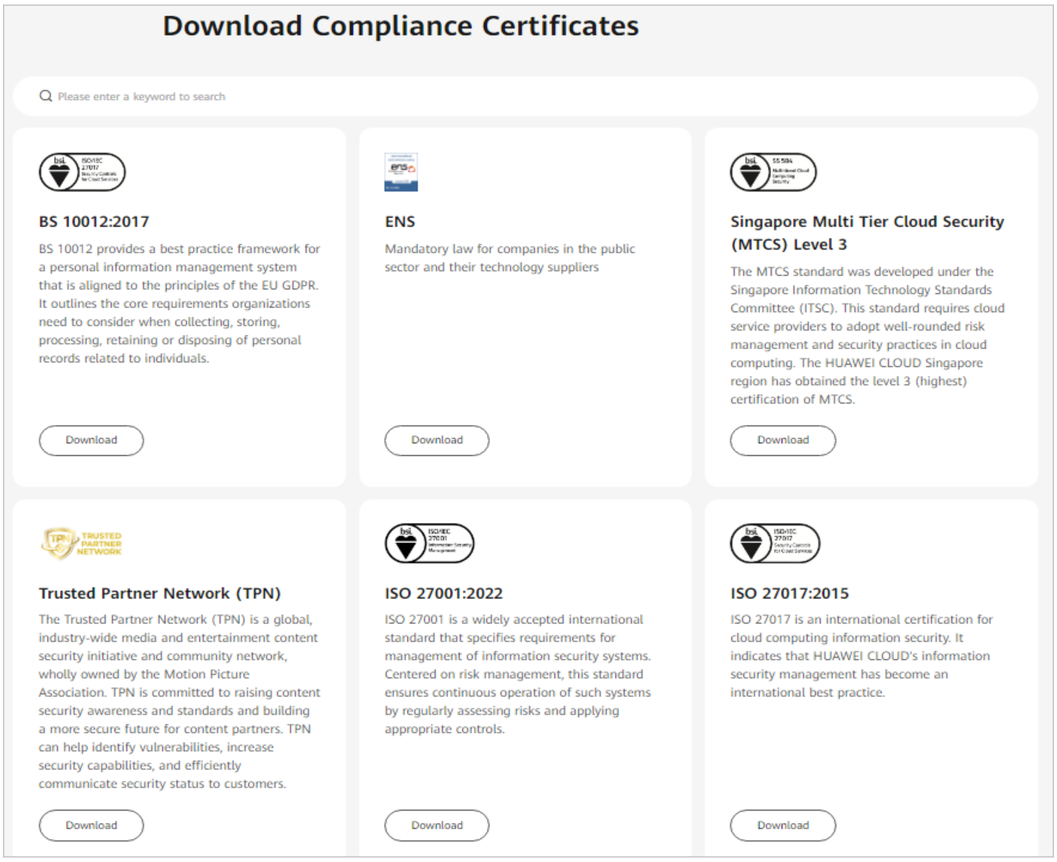
CodeArts Build uses cross-AZ deployment and inter-AZ data DR solutions. A homogeneous CodeArts Build DR cluster is deployed in another AZ (cross-AZ). If a natural disaster occurs in the geographical location of a production cluster or a fault occurs in the cluster, the production cluster cannot provide read and write services. In this case, the DR cluster can be switched to the production cluster to ensure that service processes can be quickly started and recovered for continuity and reliability.

5.6 Certificates

Compliance Certificates

Huawei Cloud services and platforms have obtained various security and compliance certifications from authoritative organizations, such as International Organization for Standardization (ISO). You can **download** them from the console.

Figure 5-2 Downloading compliance certificates



Download Compliance Certificates

Please enter a keyword to search

BS 10012:2017

BS 10012 provides a best practice framework for a personal information management system that is aligned to the principles of the EU GDPR. It outlines the core requirements organizations need to consider when collecting, storing, processing, retaining or disposing of personal records related to individuals.

Download

ENS

Mandatory law for companies in the public sector and their technology suppliers

Download

Singapore Multi Tier Cloud Security (MTCS) Level 3

The MTCS standard was developed under the Singapore Information Technology Standards Committee (ITSC). This standard requires cloud service providers to adopt well-rounded risk management and security practices in cloud computing. The HUAWEI CLOUD Singapore region has obtained the level 3 (highest) certification of MTCS.

Download

Trusted Partner Network (TPN)

The Trusted Partner Network (TPN) is a global, industry-wide media and entertainment content security initiative and community network, wholly owned by the Motion Picture Association. TPN is committed to raising content security awareness and standards and building a more secure future for content partners. TPN can help identify vulnerabilities, increase security capabilities, and efficiently communicate security status to customers.

Download

ISO 27001:2022

ISO 27001 is a widely accepted international standard that specifies requirements for management of information security systems. Centered on risk management, this standard ensures continuous operation of such systems by regularly assessing risks and applying appropriate controls.

Download

ISO 27017:2015

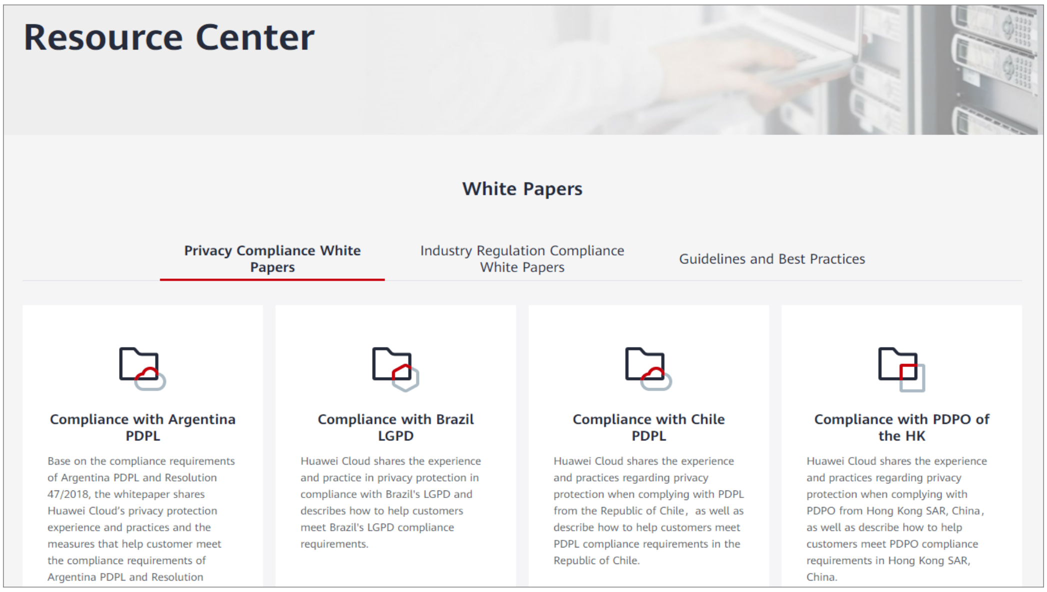
ISO 27017 is an international certification for cloud computing information security. It indicates that HUAWEI CLOUD's information security management has become an international best practice.

Download

Resource Center

Huawei Cloud also provides the following resources to help users meet compliance requirements. For details, see **Resource Center**.

Figure 5-3 Resource center



Resource Center

White Papers

Privacy Compliance White Papers | Industry Regulation Compliance White Papers | Guidelines and Best Practices

Compliance with Argentina PDPL

Base on the compliance requirements of Argentina PDPL and Resolution 47/2018, the whitepaper shares Huawei Cloud's privacy protection experience and practices and the measures that help customer meet the compliance requirements of Argentina PDPL and Resolution

Compliance with Brazil LGPD

Huawei Cloud shares the experience and practice in privacy protection in compliance with Brazil's LGPD and describes how to help customers meet Brazil's LGPD compliance requirements.

Compliance with Chile PDPL

Huawei Cloud shares the experience and practices regarding privacy protection when complying with PDPL from the Republic of Chile, as well as describe how to help customers meet PDPL compliance requirements in the Republic of Chile.

Compliance with PDPO of the HK

Huawei Cloud shares the experience and practices regarding privacy protection when complying with PDPO from Hong Kong SAR, China, as well as describe how to help customers meet PDPO compliance requirements in Hong Kong SAR, China.

6 Constraints and Restrictions

Table 6-1 describes the restrictions on CodeArts Build.

Table 6-1 CodeArts Build restrictions

Category	Item	Limit
Compilation and build	Maximum size of a single file to be uploaded in file management	100 KB
	Number of templates	5,000 per tenant
	Maximum parallel executions	2
	Maximum number of build tasks per tenant	10,000
	Maximum number of build task executions per year	Unlimited
Single build task	Maximum number of build actions	50
	Maximum parallel executions per task	2
	Maximum duration per build	4 hours
	Retention period of historical build task execution records	1 month