

CodeArts Build

Service Overview

Issue 01
Date 2023-11-30



Copyright © Huawei Technologies Co., Ltd. 2023. All rights reserved.

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

Trademarks and Permissions



HUAWEI 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 "Vul. Response Process". For details about the policy, see the following website:<https://www.huawei.com/en/psirt/vul-response-process>
For enterprise customers who need to obtain vulnerability information, visit:<https://securitybulletin.huawei.com/enterprise/en/security-advisory>

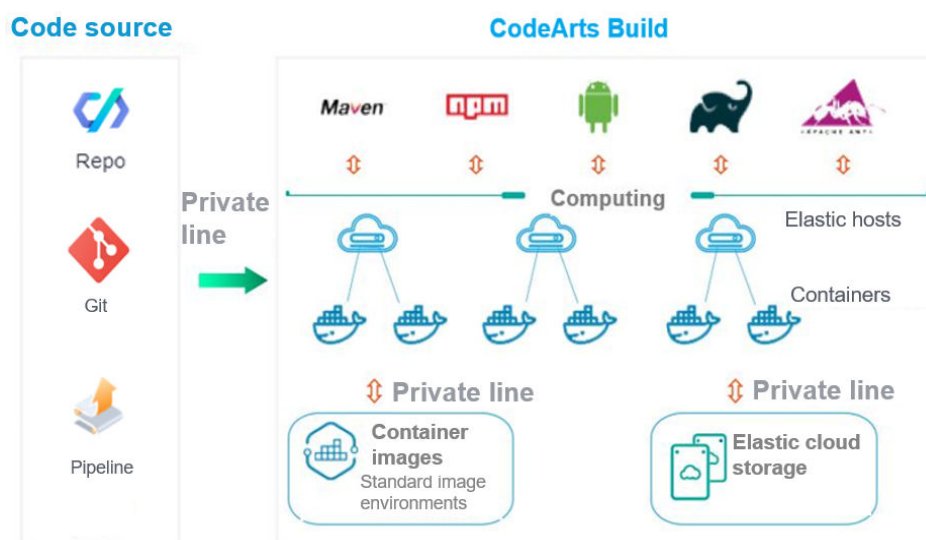
Contents

1 What Is CodeArts Build?	1
2 Advantages	2
3 Use Cases	3
4 Build Environment	5
5 Security	20
5.1 Authentication and Access Control.....	20
5.2 Data Protection Technologies.....	21
5.3 Cloud Trace Service (CTS).....	21
5.4 Service Resilience.....	21
6 Constraints and Restrictions	23

1 What Is CodeArts Build?

Build is the process of converting source code into a target file and packaging 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 shorter delivery period and higher delivery efficiency. With CodeArts Build, you can create, configure, and run build tasks with a few clicks. CodeArts Build also supports automation of 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: built-in images provided by CodeArts Build, supporting compilation and packaging with popular programming languages.
- Custom images: customized for your own applications and supporting multi-language compilation and packaging.

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 account-level cache
- Direct Connect for faster transmission

Scalable

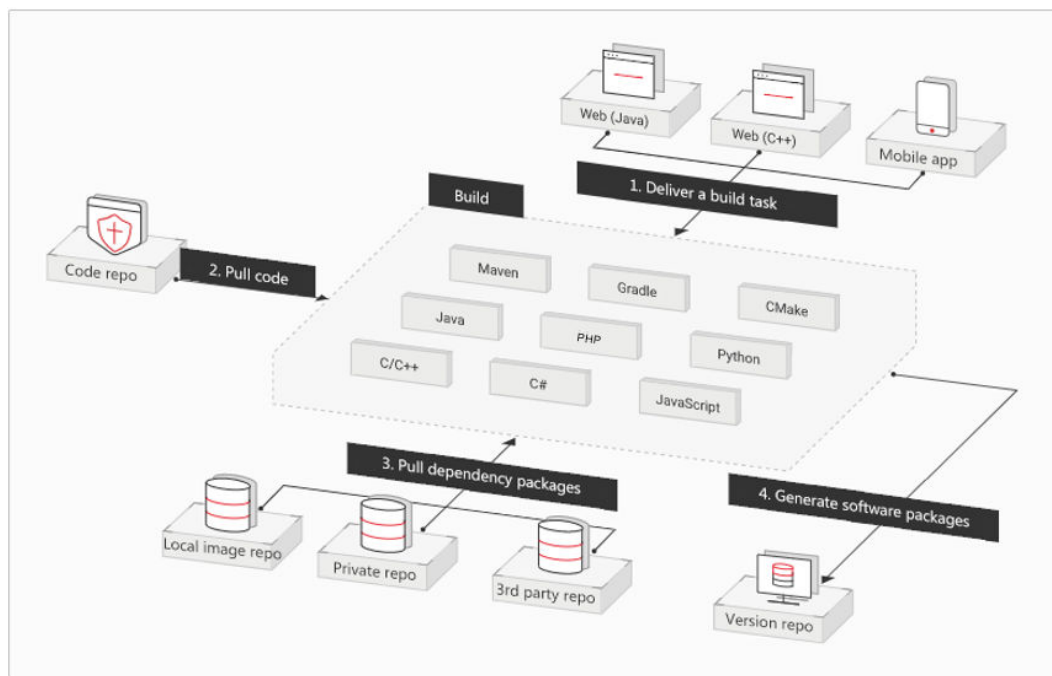
CodeArts Build can flexibly interconnect with different code hosting services and provides the following features:

- 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: There are various types of Internet services. They often require cross-platform, multi-language programming.
- 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 allocation of build resources accelerates the build speed. Using CodeArts Build, you can build C/C++ applications in Linux.

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 build Android mobile apps.

4 Build Environment

This section describes the basic environments (image and tool versions) for each build mode.

Build with Maven

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

Tool Version	Built-in Tool
maven3.8.5-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

Build 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	

Build Environment	Built-in Tool
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
JDK	OpenJDK 1.7–1.9 and OpenJDK 10–16
NDK	NDK 16–23

Build 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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

Build 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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

Tool Version	Built-in Image Tool
Node.js 16	<ul style="list-style-type: none"> • 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

Build 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
JDK	OpenJDK 1.7–1.9 and OpenJDK 10–17

Build 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.)

Tool Version	Description
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.)

Build 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	<ul style="list-style-type: none">• 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

Tool Version	Built-in Image Tool
Node.js 10.1.0	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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

Build 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	<ul style="list-style-type: none">• 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
Python 3.5	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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

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

Build 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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

Tool Version	Built-in Image Tool
Node.js 13	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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

Build Using 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

Build 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	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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

Tool Version	Built-in Image Tool
cmake2.8.12-gcc6.4.0	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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
cmake3.10.1-gcc5.5.0	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • 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

Build 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

Build 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

Build 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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

Tool Version	Built-in Image Tool
Node.js 16	<ul style="list-style-type: none">• 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

5 Security

5.1 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

5.2 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.	-

5.3 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](#).

5.4 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.

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 account
	Maximum parallel executions	5
	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 steps	50
	Maximum parallel executions per task	5
	Maximum duration per build	4 hours
	Retention period of historical build task execution records	1 month