Deploy_User Guide_HWS&HWS_HK&G42-Copy

Issue 01

Date 2024-07-04

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.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: https://www.huawei.com

Email: support@huawei.com

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

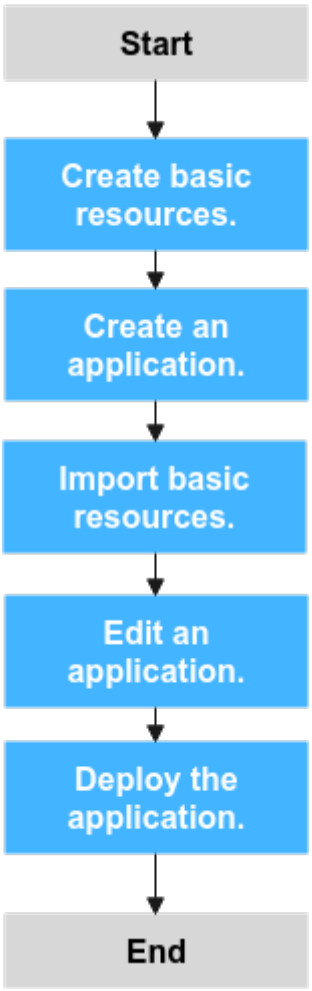
1 Overview

CodeArts Deploy is a visualized and automatic deployment service. It provides various deployment actions for you to customize deployment process, improving efficiency and reducing costs.

CodeArts Deploy has the following features:

- CodeArts Deploy supports deployment on hosts (Huawei Cloud ECSs, your own hosts, and third-party hosts) or containers (Huawei Cloud CCE clusters, on-premises clusters, and third-party clusters).
- Functions are encapsulated as plug-ins, which are easy to use. Common applications can be deployed out of the box.
- CodeArts Deploy provides system templates such as Tomcat, Spring Boot, and Django for you to deploy tasks quickly.
- You can drag and drop atomic actions to orchestrate and assemble applications, customize application templates, and create applications in one click.

You can use CodeArts Deploy to deploy a project. The following figure shows the workflow.



The operations involved are as follows:

- **Creating basic resources**
- **Creating an application**
- **Importing basic resources**
- **Editing an application**
- **Deploying an application**

2 Managing Basic Resources

2.1 Managing Host Clusters

2.1.1 Adding and Editing a Host Cluster

The **Basic Resources** page displays host clusters and Kubernetes clusters (available soon), to be deployed. You can create basic resources and import them to an application for deployment. A **target host** is the final deployment object. CodeArts Deploy deploys your resources such as artifacts to the target host in the environment. A **proxy server** is a computer used to provide access channels for other target hosts without public IP addresses. In CodeArts Deploy, proxy hosts are usually referred to as ECSs bound to public IP addresses unless otherwise specified.

Prerequisites

- You have the permission to edit projects. If you do not have the permission, contact the project administrator to grant the permission.
- A project is available. If no project is available, **create one**.
- A host is available. If no host is available, see **Applying for an ECS**.
- The proxy has been configured by referring to **Configuring a Proxy Host** and **Security Configuration**.
- The target host has been configured by referring to **Configuring a Target Host** and **Security Configuration**.

Creating a Cluster

Step 1 Go to the **Basic Resources** page.

In the target project, choose **Settings** > **General** > **Basic Resources**. The **Host Cluster** page is displayed.

Step 2 Create a host cluster.

Click **Create Host Cluster**, enter the following information, and click **Save**.

Parameter	Mandatory	Description
Cluster Name	Yes	Enter a user-defined host cluster name.
OS	Yes	Choose Linux or Windows as the operating system for the host.
Host Connection Mode	Yes	• Direct connection: Select a host bound with an EIP as the target host to connect to CodeArts. • Proxy: Select a host bound with an EIP as the proxy host to connect to CodeArts. If the target host cannot connect to the public network, select the proxy mode.
Execution Resource Pool	Yes	A resource pool is a collection of physical environments where commands are executed during software package deployment. You can use an **official resource pool** hosted by Huawei Cloud or host your own servers as a **self-hosted resource pool** on Huawei Cloud. For details about hosting your own servers, see **Self-hosted Resource Pool**.
Description	No	Description of the host cluster.

NOTE

To use a **Self-hosted resource pool**, perform the following operations:

- Configure a **Self-hosted resource pool** by referring to **Self-hosted Resource Pool**.
- On the **Basic Information** tab page of the target application, select **Self-hosted** for **Execution Host**.

Step 3 (Optional) If **Host Connection Mode** is set to **Proxy**, you need to add a proxy host and a target host.

1. Create a proxy host.

   Click **Add Host**, select **Adding IP** for **Add Hosts by**, enter the following information, and click **OK**. The proxy host is created.

   Table 2-1 Parameters of the proxy host (Linux)

Parameter	Mandatory	Description
Host Name	Yes	Enter a user-defined proxy host name.
IP	Yes	Enter a public IP address bound to the proxy host.

Parameter	Mandatory	Description
OS	Yes	The default value is the OS of the current host cluster and cannot be changed.
Authorization	Yes	Select a **password** or **key** for authentication as required. – If you select **Password**, the **Username** and **Password** are displayed. Take ECS as an example. You need to enter the ECS username and password. – If you select **Key**, the **Username** and **Key** are displayed. For details about how to generate and obtain a key, see **Obtaining the Linux key**.
SSH Port	Yes	Port **22** is recommended. You may customize the port number.

Table 2-2 Parameters of the proxy host (Windows)

Parameter	Mandatory	Description
Host Name	Yes	Enter a user-defined proxy host name.
IP	Yes	Enter a public IP address bound to the proxy host.
OS	Yes	The default value is the OS of the current host cluster and cannot be changed.
Authorization	Yes	Windows proxies support only password authentication. Take ECS as an example. You need to enter the ECS username and password.
Winrm Port	Yes	Port **5986** is recommended. You may customize the port number.

2. Add a target host.

   Click **Add Host**, select **Adding IP** for **Add Hosts by**, enter the following information, and click **OK**. The target host is created.

Table 2-3 Parameters of the target host (Linux)

Parameter	Mandatory	Description
Host Name	Yes	Enter a user-defined target host name.

Parameter	Mandatory	Description
Proxy Host	Yes	Select the target proxy host as the network proxy of the target host that cannot connect to the public network.
IP	Yes	Enter the private or public IP address of the target host.
OS	Yes	The default value is the OS of the current host cluster and cannot be changed.
Authorization	Yes	Select a **password** or **key** for authentication as required. – If you select **Password**, the **Username** and **Password** are displayed. Take ECS as an example. You need to enter the ECS username and password. – If you select **Key**, the **Username** and **Key** are displayed. For details about how to generate and obtain a key, see **Obtaining the Linux key**.
SSH Port	Yes	Port **22** is recommended. You may customize the port number.

Table 2-4 Parameters of the target host (Windows)

Parameter	Mandatory	Description
Host Name	Yes	Enter a user-defined target host name.
Proxy Host	Yes	Select the target proxy host as the network proxy of the target host that cannot connect to the public network.
IP	Yes	Enter the private or public IP address of the target host.
OS	Yes	The default value is the OS of the current host cluster and cannot be changed.
Authorization	Yes	Windows proxies support only password authentication. Take ECS as an example. You need to enter the ECS username and password.
Proxy Forwarding Port	Yes	Set this port to the listening port number in the **Windows proxy configuration**. Port **54** is recommended. User-defined ports are supported.

Step 4 (Optional) If **Host Connection Mode** is set to **Direct Connection**, you only need to add a target host.

Click **Add Host**, select **Adding IP** for **Add Hosts by**, enter the following information, and click **OK**. The target host is created.

Table 2-5 Parameters of the target host (Linux)

Parameter	Mandatory	Description
Host Name	Yes	Enter a user-defined target host name.
IP	Yes	Enter the public IP address bound to the target host.
OS	Yes	The default value is the OS of the current host cluster and cannot be changed.
Authorization	Yes	Select a **password** or **key** for authentication as required. • If you select **Password**, the **Username** and **Password** are displayed. Take ECS as an example. You need to enter the ECS username and password. • If you select **Key**, the **Username** and **Key** are displayed. For details about how to generate and obtain a key, see **Obtaining the Linux key**.
SSH Port	Yes	Port **22** is recommended. You may customize the port number.

Table 2-6 Parameters of the target host (Windows)

Parameter	Mandatory	Description
Host Name	Yes	Enter a user-defined target host name.
IP	Yes	Enter the public IP address bound to the target host.
OS	Yes	The default value is the OS of the current host cluster and cannot be changed.
Authorization	Yes	Windows proxies support only password authentication. Take ECS as an example. You need to enter the ECS username and password.
Winrm Port	Yes	Port **5986** is recommended. You may customize the port number.

Step 5 (Optional) You can also select **Importing ECS** for **Add Hosts by** and add your Huawei Cloud ECS as the target host or proxy host.

Step 6 Verify the host connectivity.

After the host is created, the system automatically verifies the connectivity. If the connectivity verification fails, click **Failed** and rectify the fault based on the failure cause displayed in the dialog box or click **View Solution**.

----End

⚠ CAUTION

You should meet the following requirements to import purchased ECS:

- ECS is running.
- ECS and host cluster have the same OS.
- Proxy host has a public IP (proxy mode) when using official resource pool.
- Host is not the imported target host.

In the proxy mode, you need to configure the proxy host before using the target host.

Editing a cluster

Step 1 Go to the host cluster page.

1. In the target project, choose **Settings** > **General** > **Basic Resources**. The Host Cluster page is displayed.
2. Click the target host cluster to enter its details page.

Step 2 Edit a host cluster.

- **Adding a host**: Click + in the **Operation** column of a cluster to add a host to the cluster.
- **Editing a cluster**: Click ✎ in the **Operation** column of a cluster to modify the cluster name, execution host, and description.
- **Deleting a cluster**: Click ••• in the **Operation** column of a cluster, click **Delete**, and click **OK**.

  NOTE

  If the target cluster contains resources, you need to clear all resources in the cluster and then delete the cluster.

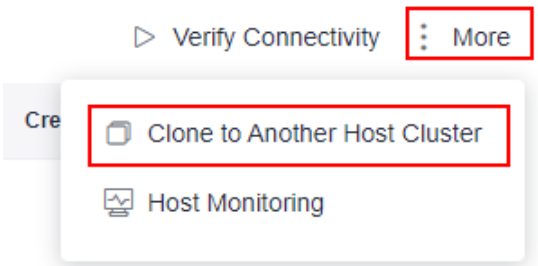
- **Managing Permissions**: Click ••• in the **Operation** column of a cluster, click **Manage Permissions**, and configure operation permissions for each role. Enable or disable permissions as required.

Default Permissions of a Host Cluster						
Role/ Permission	**View**	**Edit**	**Delete**	**Add Host**	**Clone Host**	**Manage Permissions**

Default Permissions of a Host Cluster						
Host cluster creator	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)
Project creator	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)
Project manager	Yes	Yes	Yes	Yes	Yes	Yes
Product manager	Yes	No	No	No	No	No
Test manager	Yes	No	No	No	Yes	No
O&M manager	Yes	No	No	No	Yes	No
System engineer	Yes	No	No	No	No	No
Committer	Yes	No	No	No	No	No
Developer	Yes	Yes	Yes	Yes	Yes	No
Tester	Yes	No	No	No	Yes	No
Participant	Yes	No	No	No	Yes	No
Viewer	Yes	No	No	No	Yes	No

Step 3 Edit hosts in the host cluster.

- **Verifying host connectivity in batches**: Select multiple hosts and click Verify Connectivity.
- **Cloning a host to another host cluster in batches**: Select multiple hosts and choose **More** > **Clone to Another Host Cluster**. Then select the target host.



- **Editing a host**: Click in the **Operation** column of a host to modify the configuration.
- **Enabling network connectivity verification**: Click in the **Operation** column of a host.
- **Deleting a host**: Click in the **Operation** column of a host, click **Delete**, and click **OK**.

  NOTE

  - If you want to delete a host from an application, select **Disassociate and Delete** to remove the host information from the environment. Otherwise, the host cannot be deleted.
  - A proxy host cannot be deleted directly. It is deleted only when its last target host is deleted from the environment.
- **Cloning a host to another host cluster**: Click in the **Operation** column of a host, click **Clone**, and select the target host cluster.

----End

Obtaining the Linux Key

Step 1 Check whether the key exists on the host.

Log in to the host and run the following command to switch to user **root**:

```
sudo su root
```

Run the following command to view the key file:

```
ls ~/.ssh
```

- If a message is displayed indicating that the directory does not exist or the **~/.ssh** directory does not contain the **id_rsa** file, go to **Generate a key**.
- If the **id_rsa** file exists in the **~/.ssh** directory, you can use the existing key file or go to **Generate a key** to generate a new key file.

Step 2 Generate a key.

Perform the following steps:

1. Generate a key.
   ```
   ssh-keygen -t rsa
   ```
2. When the following information is displayed, press **Enter**.
   ```
   Enter file in which to save the key (/root/.ssh/id_rsa):
   ```
3. When the following information is displayed, press **Enter**. By default, no password is set. (Setting the password will fail the key verification of CodeArts Deploy.)
   ```
   Enter passphrase (empty for no passphrase):
   ```
4. When the following information is displayed, press **Enter**.
   ```
   Enter same passphrase again:
   ```
5. If the following information is displayed, the key has been generated:

6. Run the following command: to view the key file generated in the **.ssh** directory. The **id_rsa** and **id_rsa.pub** files store the generated private key and public key, respectively.

```
ls ~/.ssh
```

Step 3 Check information about the key generated.

Run the following command:

```
cat ~/.ssh/id_rsa
```

- If the key prefix is **-----BEGIN RSA PRIVATE KEY-----**, the key is correct. Copy the key and save it to the local PC. Enter the key when adding a host or proxy.
- If the key prefix is **-----BEGIN OPENSSH PRIVATE KEY-----**, the key is incorrect. Run the following command to generate a new key:

```
ssh-keygen -m PEM -t rsa
```

Step 4 Authorize the key.

Run the following command to add the public key to the **authorized_keys** file of the host:

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

NOTE

- To perform operations on the **authorized_keys** file, you must have the permission to operate the **id_rsa** and **id_rsa.pub** files.
- When the key is used for connectivity verification, the username must be the name of the user who operates the **authorized_keys** file.
- Do not copy unnecessary spaces when manually copying the key. Otherwise, the connectivity verification may fail.

----End

2.1.2 Host Connection Mode

2.1.2.1 Overview

This section describes the mechanism and configuration method of the host connection mode. In the host deployment scenario, the execution host of CodeArts

Deploy communicates with the target host through SSH/WSMan to deploy applications. An **execution host** is also called a **resource pool** where deployment physically executed. In addition to an official resource pool, CodeArts Deploy allows you to connect your host to the platform to form a **self-hosted resource pool**. This service supports **direct host connection** and **proxy mode** to connect the resource pool to the target host. Before executing an application, ensure that the resource pool can communicate with target hosts. This process is called **host connectivity verification**.

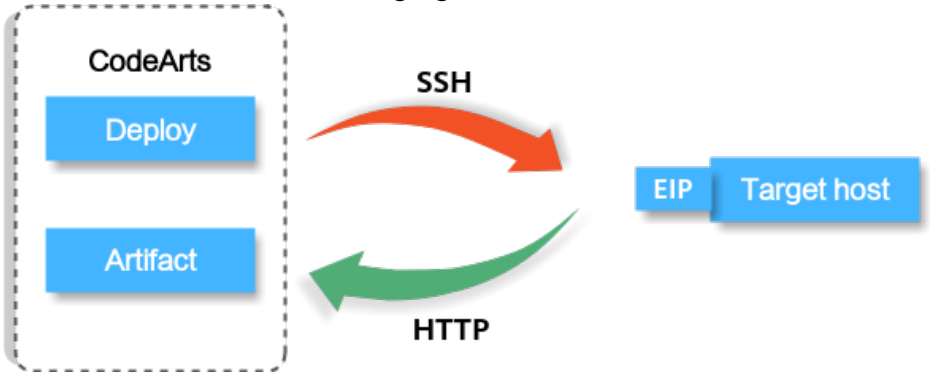
Direct Host Connection

Direct host connection is suitable when EIP resources are abundant or only a few EIPs are required for project demo verification.

NOTE

To ensure successful host connectivity verification, configure the **target host** and **enable the corresponding port**.

EIPs are bound to servers for connecting official resource pools with the target hosts, as shown in the following figure.



Proxy

Proxy connection is suitable when there are no sufficient EIP resources and deployment on ECSs without EIPs is required.

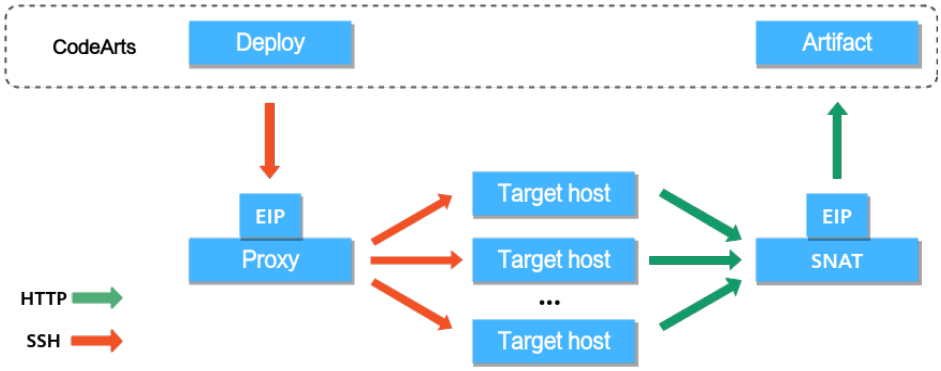
NOTE

To ensure successful host connectivity verification, configure the **target host** and **proxy host** and **enable the corresponding port**.

Linux proxy

Use an ECS bound with an EIP as a proxy. During the deployment, the executed commands will be delivered to the proxy and forwarded to each target host through SSH. The hosts will be deployed in batches.

When executing a deployment application, the target host accesses the NAT gateway to pull the software package, as shown in the following figure.



NOTE

- Red indicates the process of delivering deployment commands.
- Green indicates the process of pulling software packages.

Windows proxy

Use an ECS bound with an EIP as a proxy. During the deployment, the executed commands will be delivered to the proxy and forwarded to each target host through ports. The hosts will be deployed in batches.

2.1.2.2 Preparations

Before verifying host connectivity, perform the following operations as required:

- A host or proxy is available. For details about how to apply for a host or proxy host, see **Optional) Applying for an ECS**.
- You have obtained an EIP. For details about, see **(Optional) Applying for an EIP**.

(Optional) Applying for an ECS

Step 1 Go to the console, choose **Service List** > **Compute** > **Elastic Cloud Server**. The **Elastic Cloud Server** page is displayed.

Step 2 Click **Buy ECS** on the ECS console.

Step 3 On the ECS configuration page, set parameters as prompted.

Step 4 After setting the parameters, click **Submit** and the ECS is created.

----End

NOTICE

You can apply for an EIP during ECS creation or by referring to **(Optional) Applying for an EIP**.

Configure a security group for the created ECS by referring to **Configuring a Security Group**.

(Optional) Applying for an EIP

Step 1 Go to the console. In the upper left corner of the page, choose **Service List** > **Networking** > **Elastic IP**.

Step 2 Click **Buy EIP**.

Step 3 After setting the parameters, click **Next**.

----End

2.1.2.3 Security Configuration

To ensure successful host connectivity, configure the host as follows:

- If your host is a **newly applied cloud host**, configure the port by referring to **Configuring a Security Group**.

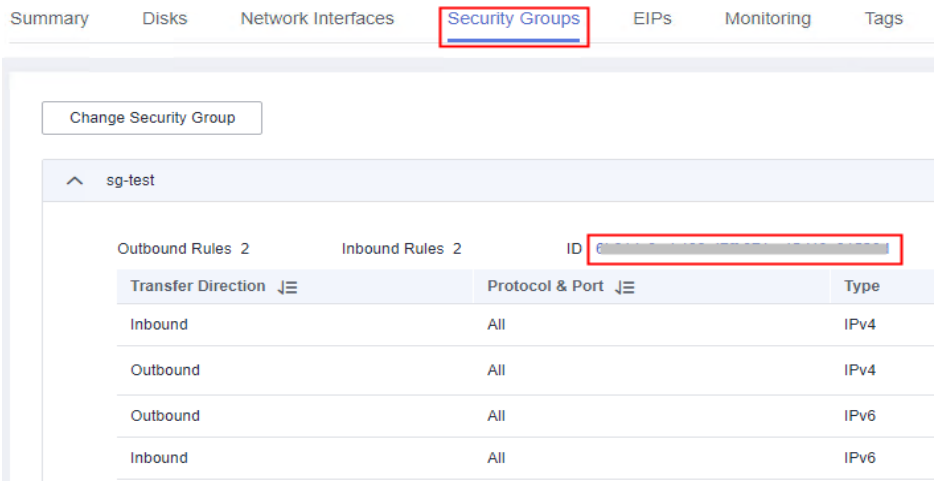
  If you have your own host, configure the port by referring to **Configuring the Firewall**.
- If you have high requirements on the security of the overall deployment process, you can **configure the IP address blacklist and whitelist** for the host.

Configuring a Security Group

Before verifying host connectivity, configure a security group and enable some ports. Otherwise, the connectivity verification may fail. (The following uses a Linux host as an example.)

Step 1 Go to the console. In the upper left corner of the page, choose **Service List** > **Compute** > **Elastic Cloud Server**. The Elastic Cloud Server page is displayed.

Step 2 Click the target ECS. On the ECS details page, click the **Security Groups** tab. Click the security group ID. On the page that is displayed, click the **Inbound Rules**.



Step 3 Click **Fast-Add Rule** and set the parameters as follows:

- For Linux hosts, enable port **22** in the inbound rule. For Windows hosts, enable ports **54**, **5985**, and **5986** in the inbound rule when adding the target host or proxy host. Set the remote end to **0.0.0.0/0** (open the preceding ports for all IP addresses).
- Remove the inbound restriction on the port of the application deployed on the host (for example, port **8080** of the Tomcat application or all ports of other applications must be enabled in the inbound direction). Otherwise, the application cannot be accessed.
- Remove the restriction on the outbound direction or at least make ports 80 and 443 accessible.

----End

Configuring the Firewall

Check the firewall configuration of the host and configure the firewall to allow the SSH protocol port to be accessed. Otherwise, the connectivity verification fails. The following part describes how to configure the firewall for different operating systems.

Linux Firewall Configurations

Table 2-7 Linux firewall configurations

OS Series	Configuration Method

CentOS/ EulerOS/ UnionTechOS	1. Check whether the SSH software package is installed on the local host.
	`rpm -qa | grep ssh`
	2. If no SSH software package is available, run the following command:
	`yum install openssh-server`
	3. Enable the SSH service.
	`service sshd start`
	4. Open the sshd configuration file.
	`vi /etc/ssh/sshd_config`
	5. Delete the comment tag before the listening port number.
	6. Restart the SSH service. Run the following command:
	`sudo service sshd start`
	7. Check whether port **22** is enabled.
	`netstat -ntpl | grep 22`

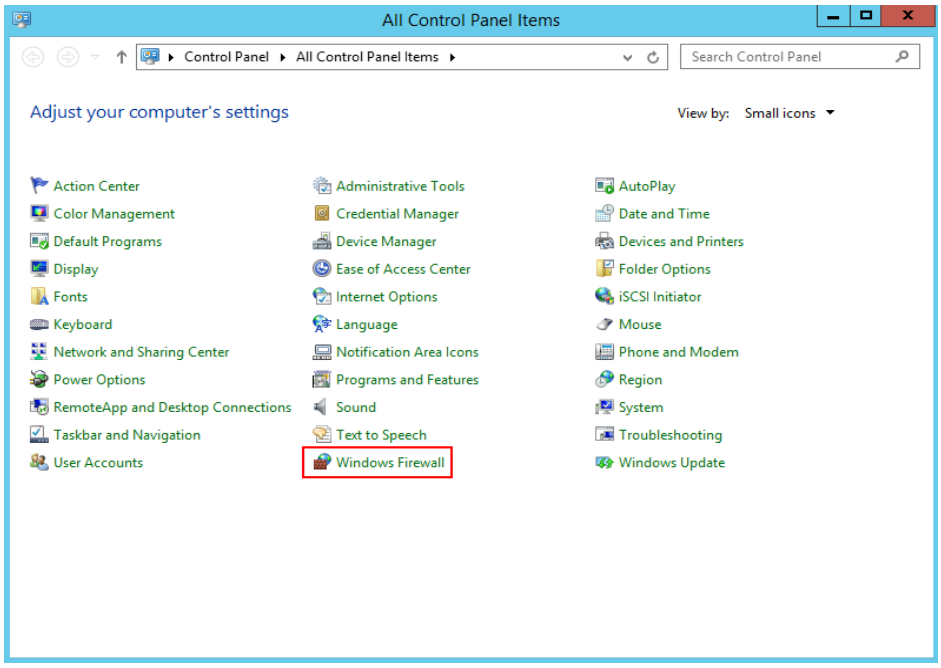
Debian	1. Log in to the system as the root user and install the ufw.
	`apt install ufw`
	2. Enable port **22**.
	`ufw allow 22/tcp`
	3. Check whether port **22** is enabled.
	`ufw status`

Ubuntu	1. Check the IP address of the local host.
	`ifconfig`
	2. Check whether the **22** port is occupied.
	`netstat -nltp|grep 22`
	3. If no port process exists, run the following commands in sequence:
	`sudo apt-get install openssh-server`
	`sudo apt-get install ufw`
	`sudo ufw enable`
	`sudo ufw allow 22`

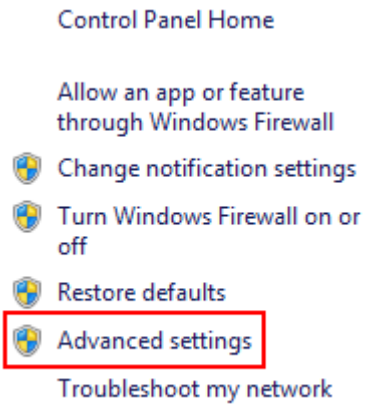
Windows Firewall Configurations

This section uses Windows Server 2012 as an example.

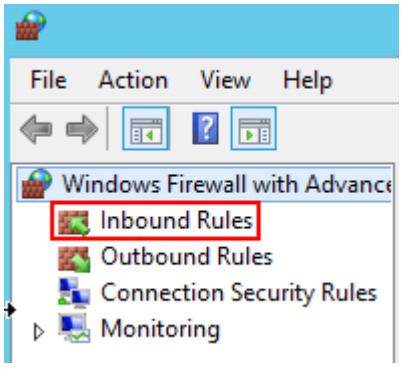
Step 1 Choose **Windows Firewall** on the control panel of the **Windows** host.



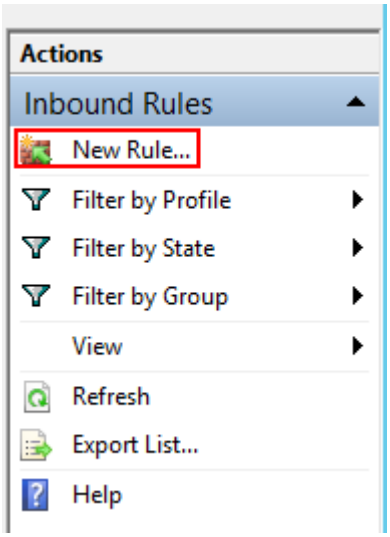
Step 2 Click **Advanced settings**.



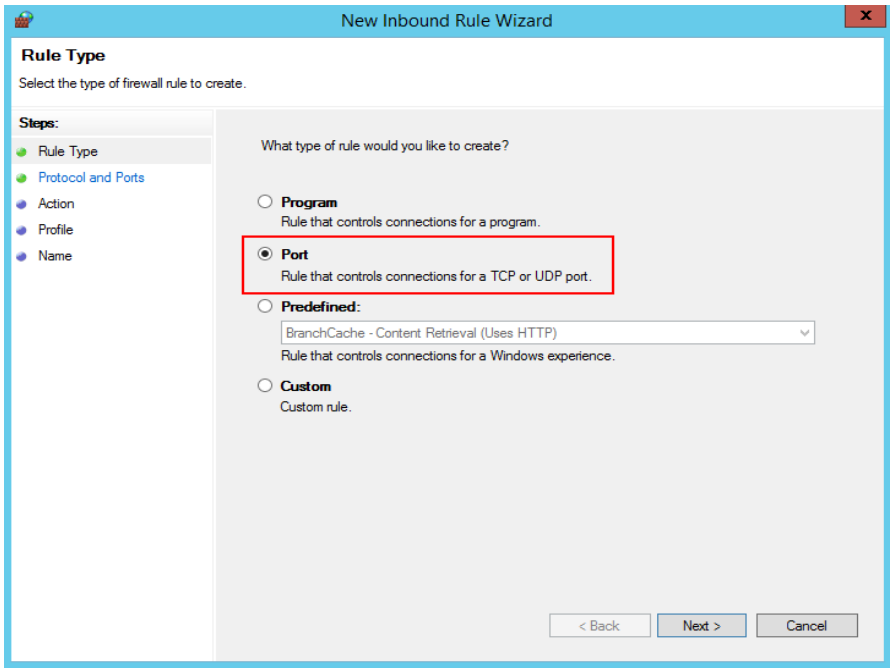
Step 3 Click **Inbound Rules**.



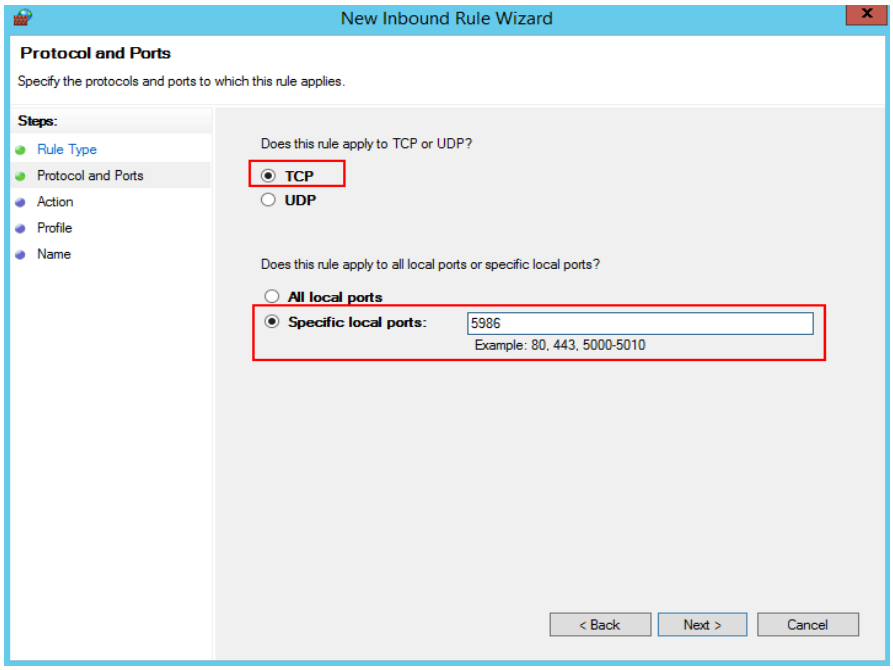
Step 4 Click **New Rule**.



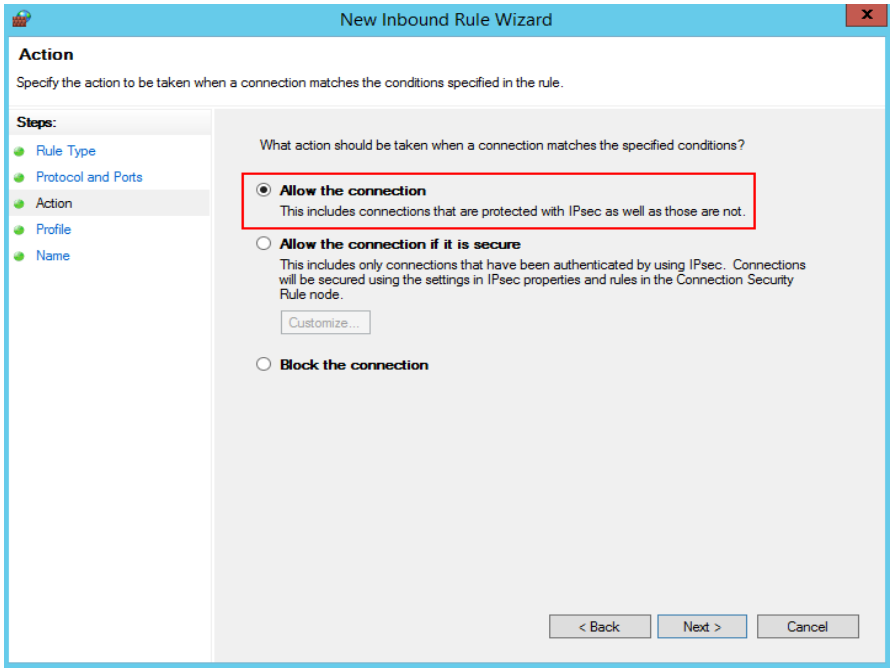
Step 5 Set **Rule Type** to **Port** and click **Next**.



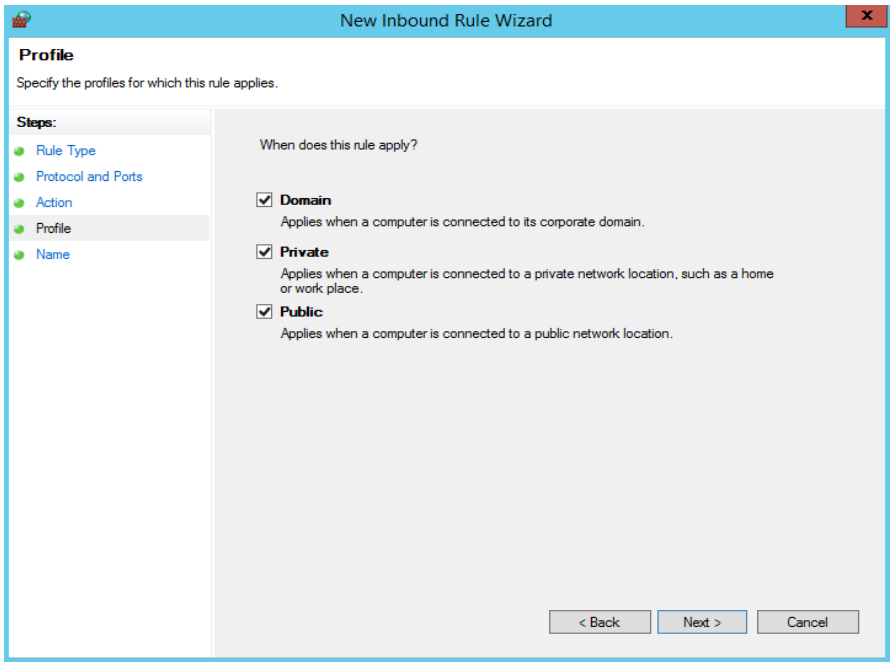
Step 6 Select **TCP** and **Specific local ports**, enter port **5986**, and click **Next**.



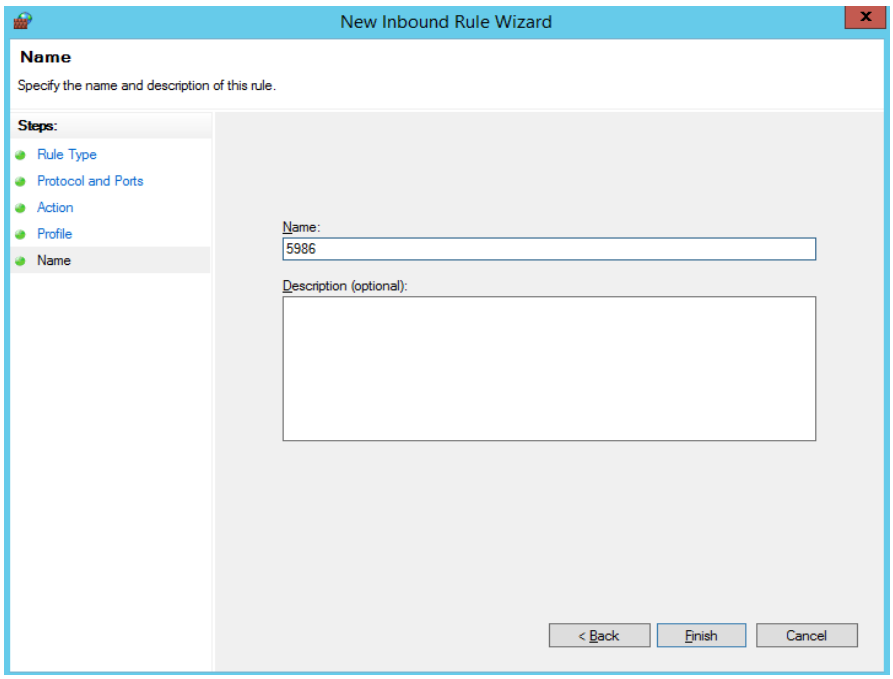
Step 7 Select **Allow the connection**, and click **Next**.



Step 8 Select all the options for **Profile** and click **Next**.



Step 9 Enter a rule name and click **Finish**.



----End

Configuring the IP Address Blacklist and Whitelist

You can add IP addresses to the firewall whitelist and check whether these IP addresses are blocked by the firewall for security purposes in the scenario where the official resource pool is used.

Commands related to the IP address whitelist

- Check whether there are blocked IP addresses:


```
vi /var/log/secure
```

- Check whether the preceding IP addresses are blocked by the host. If yes, run the following command to remove the restriction:
  ```
  vi /etc/hosts.deny
  ```
- Add required IP addresses to the whitelist. Example:
  ```
  vi /etc/hosts.allow
  sshd:***.***.***.***:allow        #Use the IP address to be added to the whitelist.
  ```

2.1.2.4 Configuring a Target Host

Before adding a host to a created host cluster, you need to perform the following operations on the host:

NOTE

This configuration has been completed for Huawei Cloud ECSs (Linux) by default. You do not need to configure it again. However, you need to configure it for Windows ECSs.

Configuring a Linux Host

To ensure that the connectivity verification of the Linux host is successful, ensure that **Python** is installed on the Linux host and the **SELinux mechanism** is enabled. The target host must meet the following requirements:

- **Install Python**

  Install **Python version 2.6** or later. If Python earlier than 2.6 has been installed, run the following commands to install the following modules on the host:

  – Ubuntu
  ```
  sudo apt install python-minimal python-simplejson
  ```
  – CentOS or EulerOS
  ```
  sudo yum install python-minimal python-simplejson
  ln -s /usr/bin/python2 /usr/bin/python
  ```

  NOTE

  Before using Advanced Packaging Tool (APT) or Yellowdog Updater, Modified (yum), ensure that an available source has been configured.

- **Enable SELinux and install libselinux-python**

  a. Run the following command to check the SELinux status:
  ```
  /usr/sbin/sestatus
  ```

  NOTE

  Mode corresponding to the value of SELinux:

  SELinux=disabled: disabled.

  SELinux=enforcing: forcible mode, indicating that all behavior that violates the security policy are prohibited.

  SELinux=permissive: indicates that all behavior that violates security policies are not prohibited but are recorded in logs.

  b. If SELinux is set to disabled, SELinux is disabled on the host. In this case, perform the following steps to change the SELinux status.

  i. Run the following command to edit the config file of the SELinux:
  ```
  vi /etc/selinux/config
  ```


ii. Modify the SELinux parameters based on the site requirements.

SELinux=enforcing: forcible mode, indicating that all behavior that violates the security policy are prohibited.

SELinux=permissive: indicates that all behavior that violates security policies are not prohibited but are recorded in logs.

iii. After the modification, press **Esc** to exit. Run the following command to save the file and exit.

```
wq
```

iv. Create the hidden file **.autorelabel** in the root directory, run the following command, and restart the Linux host.

```
touch /.autorelabel
```

c. Run the following commands to install **libselinux-python**:

- Ubuntu

```
sudo apt install libselinux-python
```

- CentOS or EulerOS

```
sudo yum install libselinux-python
```

Configuring the Host Running Windows

To ensure that the Window host connectivity verification succeeds, perform the following operations on target hosts. The following uses a Windows Server 2012 as an example. The configuration modes include **automatic script configuration** and **manual configuration**.

NOTE

For details about how to configure a host running Windows 10, or Windows Server 2016 or Windows Server 2019 as the target host, see the configuration method of Windows Server 2012. For details about how to obtain the script, see **Windows2016ConfigureRemotingForAnsible.zip**.

For details about how to configure a target host running Windows 7, see the configuration method of Windows 2012. For details about how to obtain the script, see **Windows2012ConfigureRemotingForAnsible**.

- **Automatic Script Configuration**

To use an automatic configuration script to add a host running Windows Server 2012 as an authorized host, perform the following steps:

Step 1 Perform security configuration by referring to **Security Configuration**, to ensure successful connectivity verification.

Step 2 Obtain the automatic configuration script.

1. Download **Windows2012ConfigureRemotingForAnsible.zip**.
2. Decompress **Windows2012ConfigureRemotingForAnsible.zip** to obtain script **Windows2012ConfigureRemotingForAnsible.ps1**.

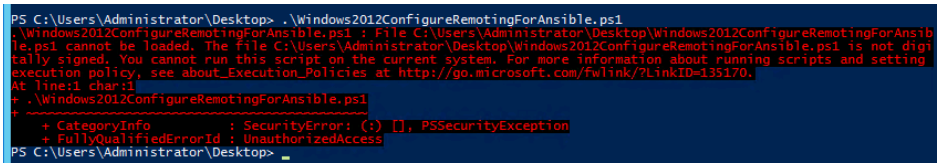
Step 3 Configure the host.

Log in to the host, open PowerShell, access the directory where script **Windows2012ConfigureRemotingForAnsible.ps1** is stored, and run the following command:

```
.\Windows2012ConfigureRemotingForAnsible.ps1
```


The output is as follows:

The system may display a message indicating that **the file cannot be loaded and a digital signature is required**.



This error occurs because the script cannot be executed in default mode of PowerShell. If this happens, run the following command in PowerShell to change the execution policy to **unrestricted**:

```
set-executionpolicy unrestricted
```

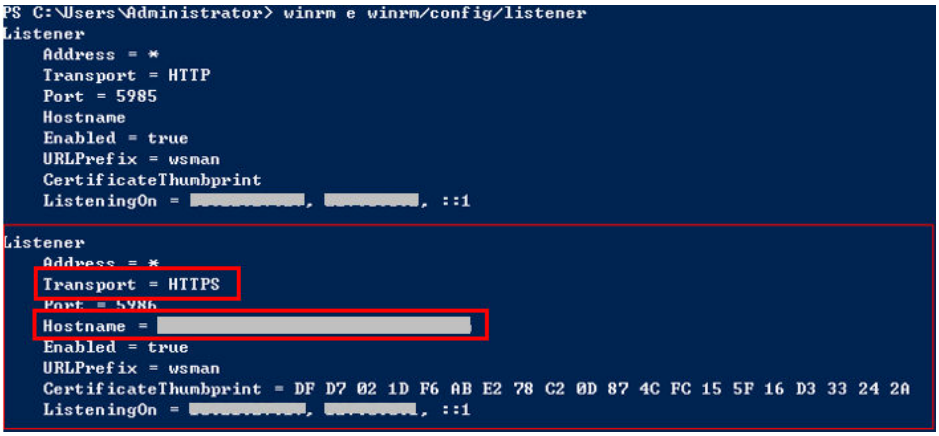
Enter **Y** to confirm the change.

Step 4 View the configuration.

Run the following command in PowerShell:

```
winrm e winrm/config/listener
```

If the output contains **HTTPS** and **Hostname** is not left blank, the listening is successful. The Windows Server 2012 deployment environment is automatically configured.



NOTE

If **Hostname** is left blank in the command output, the host does not have IIS or signature certificate information. In this case, run the following script:

```
# Configure WinRM.
winrm enumerate winrm/config/listener
winrm quickconfig
winrm set winrm/config/service/auth '@{Basic="true"}'
winrm set winrm/config/service/auth '@{CredSSP="true"}'
winrm set winrm/config/service '@{AllowUnencrypted="true"}'
# Install IIS.
Import-Module servermanager
Add-windowsfeature Web-Server,Web-WebServer,Web-Common-Http,Web-Static-Content,Web-Default-Doc,Web-Dir-Browsing,Web-Http-Errors,Web-App-Dev,Web-ASP,Web-ISAPI-Ext,Web-Health,Web-Http-Logging,Web-Log-Libraries,Web-Request-Monitor,Web-Security,Web-Filtering,Web-Stat-Compression,Web-Mgmt-Tools
# Create a self-signed certificate.
New-SelfSignedCertificate -CertStoreLocation Cert:\LocalMachine\My\ -DnsName 'windows-deploy-connect'
# View the self-signed certificate.
ls Cert:\LocalMachine\My
# Add a secure connection using the created self-signed certificate.
$windows_test_key=(Get-ChildItem -Path Cert:\LocalMachine\My | Where-Object {$_.Subject -match "windows-deploy-connect"}).Thumbprint
cmd /c "winrm set winrm/config/Listener?Address=*+Transport=HTTPS @{Enabled=`"true`";Port=`"5986`";Hostname=`"windows-deploy-connect`";CertificateThumbprint=`"$windows_test_key`"}"
```

----End

- **Manual Configuration**

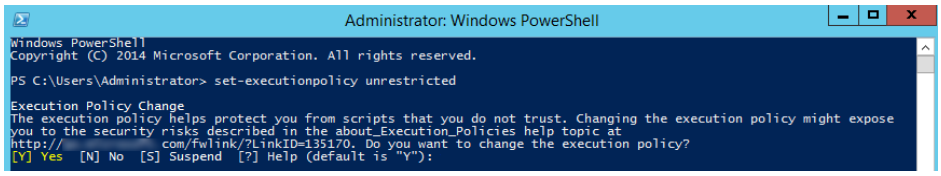
To manually add a host running Windows Server 2012 as an authorized host, perform the following steps:

Step 1 Change the PowerShell execution policy to **unrestricted**.

Open PowerShell as an administrator and run the following command:

```
set-executionpolicy unrestricted
```

The output is as follows:



Enter **Y** to confirm the change.

Step 2 Configure WinRM.

1. Run the following commands in PowerShell:

   ```
   winrm enumerate winrm/config/listener
   winrm quickconfig
   winrm set winrm/config/service/auth '@{Basic="true"}'
   winrm set winrm/config/service/auth '@{CredSSP="true"}'
   winrm set winrm/config/service '@{AllowUnencrypted="true"}'
   ```

2. Run the following command to check whether the configuration is successful:

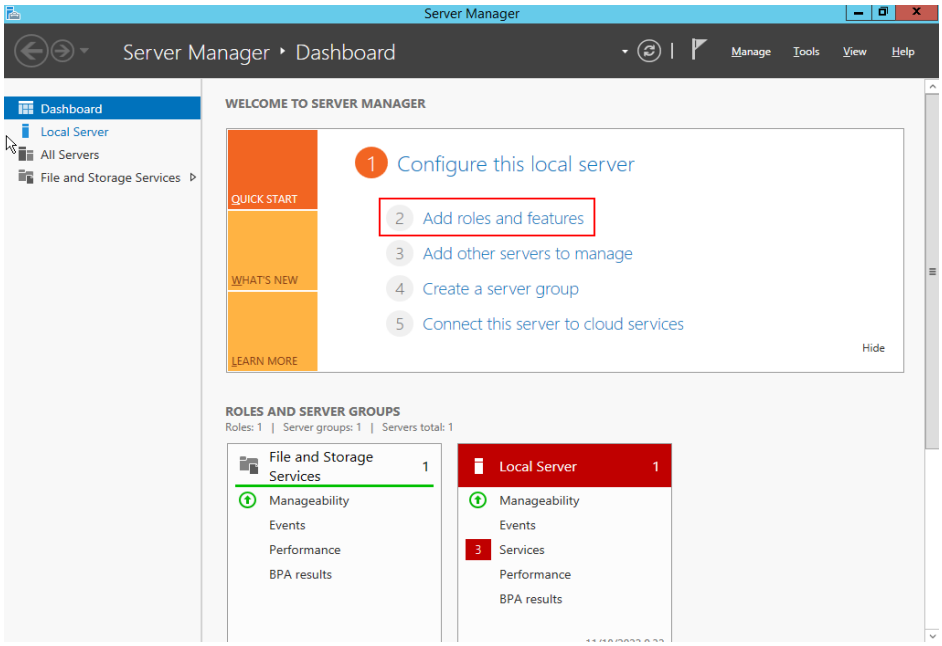
   ```
   winrm get winrm/config/service/auth
   ```

   If the values of **Basic**, **Kerberos**, and **CredSSP** are all **true**, the configuration is successful.

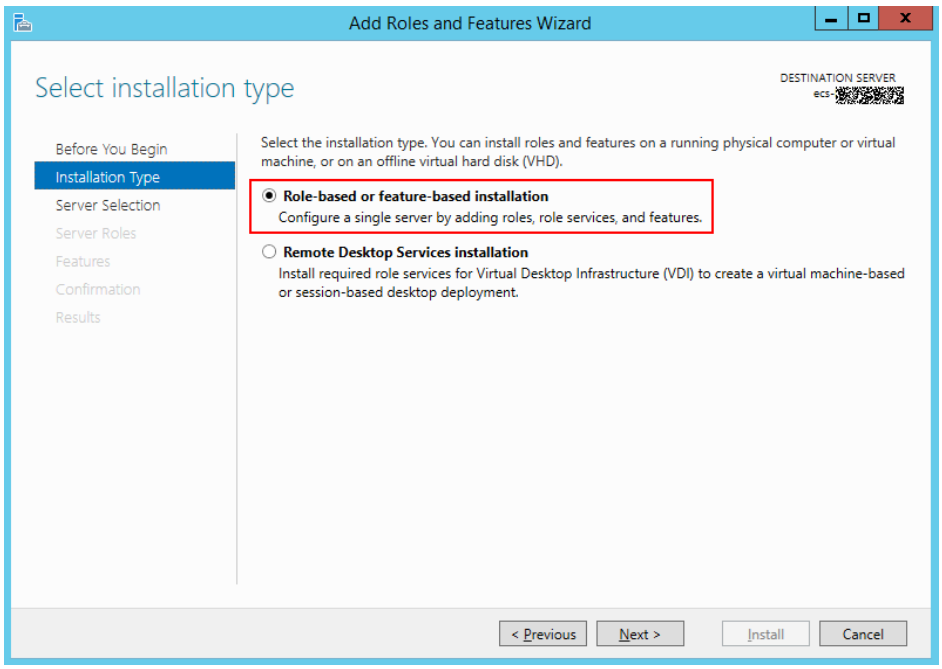

```
PS C:\Users\Administrator> winrm get winrm/config/service/auth
Auth
    Basic = true
    Kerberos = true
    Negotiate = true
    Certificate = false
    CredSSP = true
    CbtHardeningLevel = Relaxed
```

Step 3 Install the certificate.

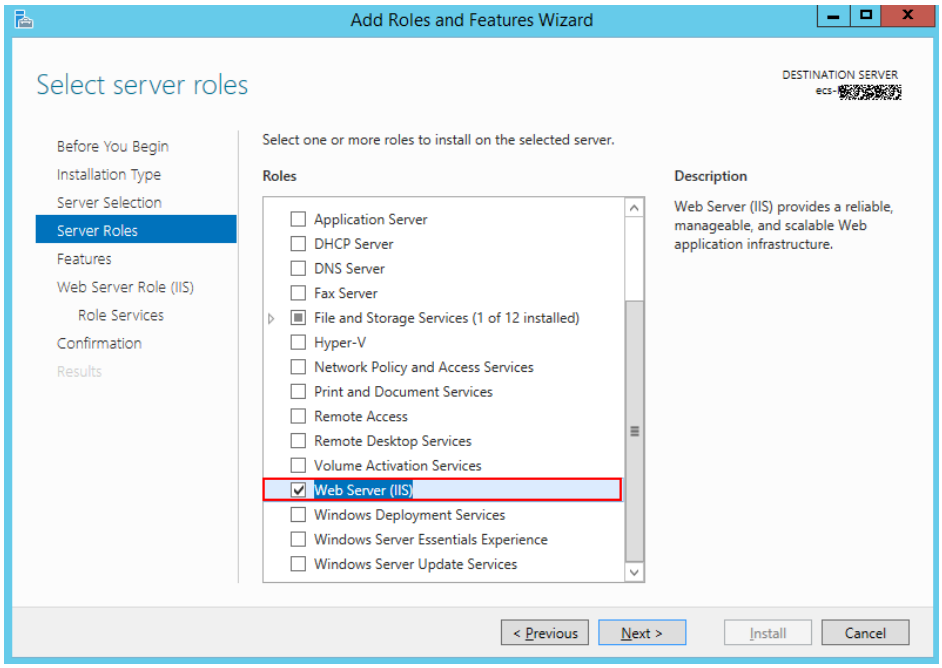
1. Open **Server Manager**, and start IIS.
2. Click **Add roles and features > Next**, as shown in the following figure.



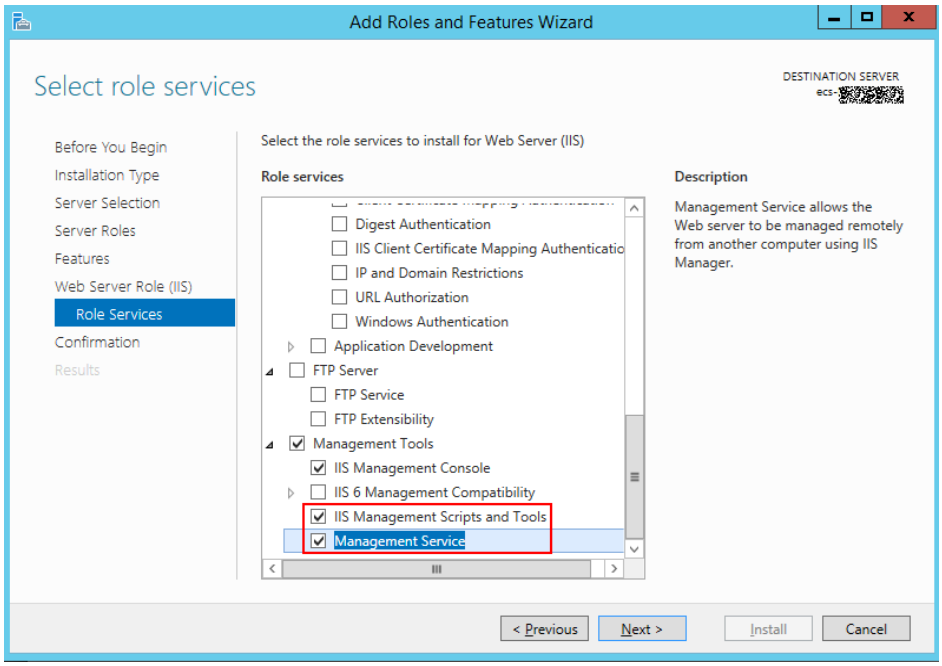
3. On the **Installation Type** page, select the first option and click **Next**.



4. Go to the **Server Roles** page and select **Web Server (IIS)**.

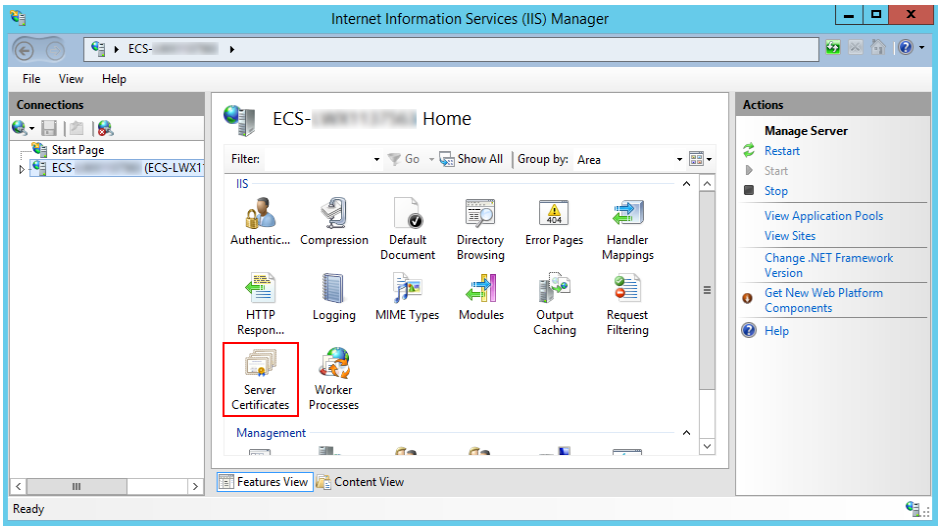


5. Go to the **Features** page, select **.NET Framework 4.5**, and click **Next**.
6. Go to the **Role Services** page, select **IIS Management Scripts and Tools** and **Management Service**, and click **Next** to complete the installation.

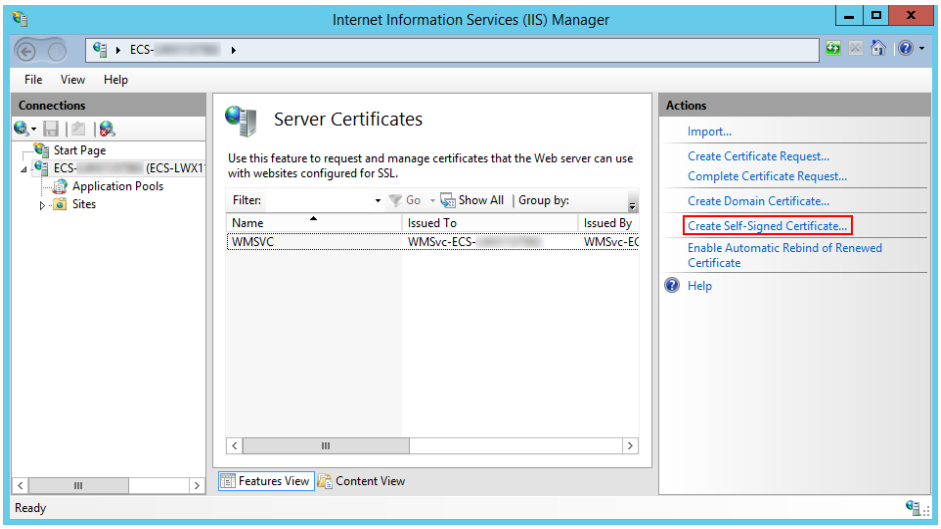


Step 4 Add a certificate.

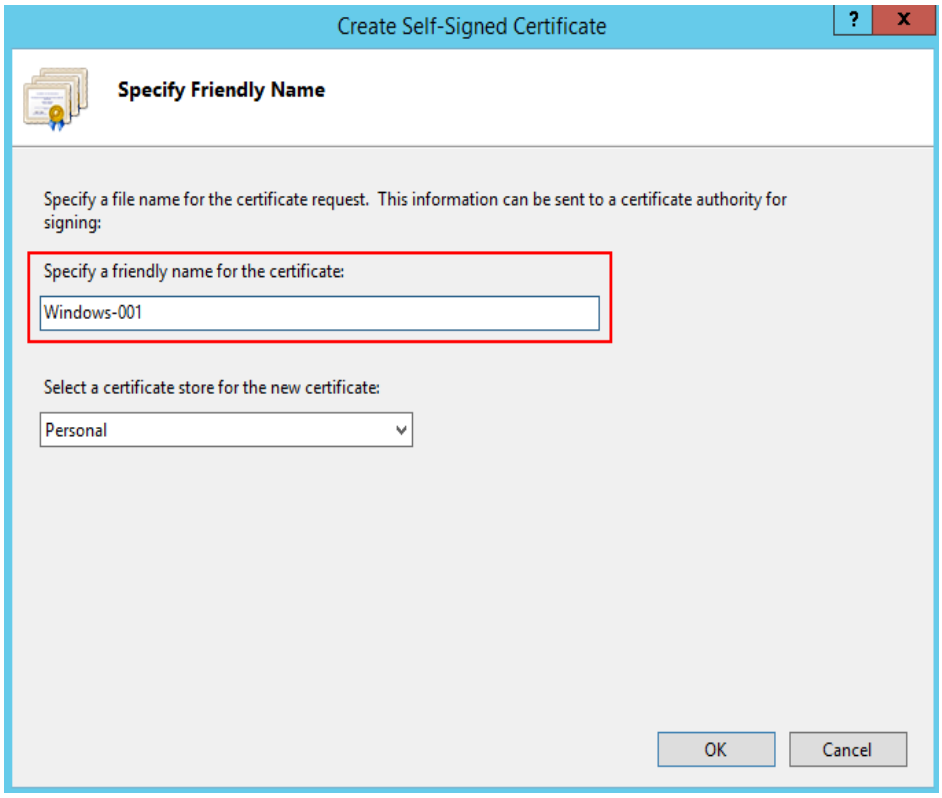
1. Press **Windows+R** to open the **Run** dialog box, enter the **inetmgr** command for opening the IIS management window, and click **OK**.
2. Open IIS Manager, and double-click **Server Certificates**.



3. On the **Server Certificates** page, click **Create Self-Signed Certificate**.



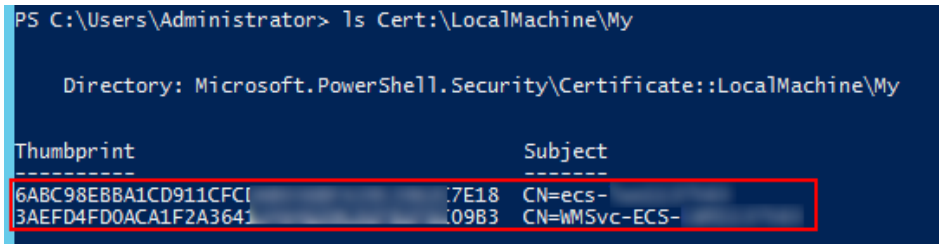
4. In the **Specify Friendly Name** dialog box, enter the certificate name and click **OK**.



5. Run the following command to view the certificate in PowerShell:

```
ls Cert:\LocalMachine\My
```

If the following two columns of data are displayed, the certificate is added.



6. Use the certificate to listen to the HTTPS port and configure a secure connection.

The commands are in the following format:

```
winrm create winrm/config/Listener?Address=*+Transport=HTTPS @{Port="User-defined port; default: 5986";Hostname="Certificate domain name";CertificateThumbprint="Certificate key value"}
```

NOTE

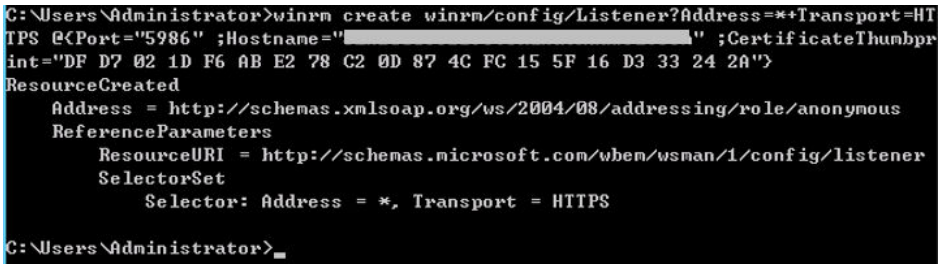
- **Hostname** is the value in the **Subject** column in the preceding step.
- **CertificateThumbprint** is the value in the **Thumbprint** column in the preceding step. Every two characters are separated by a space.

Enter the following commands in the command prompt, as shown in the following figure.

```
winrm create winrm/config/Listener?Address=*+Transport=HTTPS
@{Port="5986";Hostname="XXXXXXXXXXXXXXXX";CertificateThumbprint="DF D7 02 1D F6 AB E2 78 C2 0D 87 4C FC 15 5F 16 D3 33 24 2A"}
```

NOTE

Run the command in the command prompt and separate every two characters in the value of **Thumbprint** with a space. Otherwise, the connectivity verification may fail. If the characters are not separated by spaces, delete the signatures and add them again.



NOTE

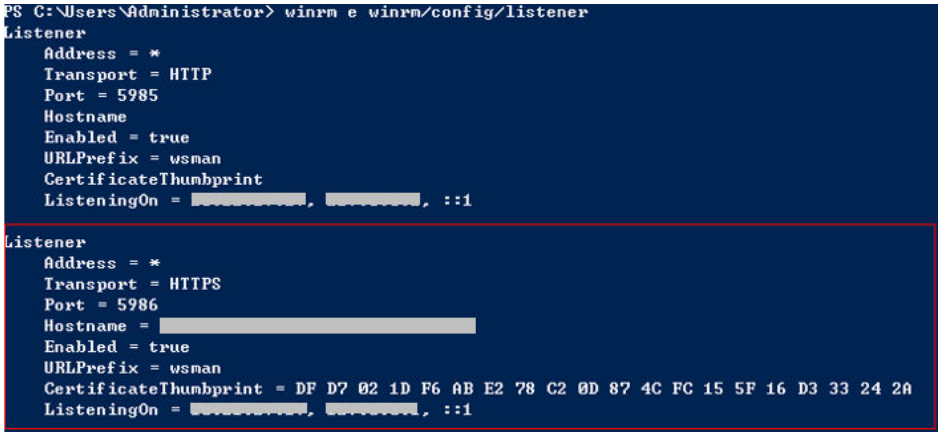
If the system displays a message indicating that the service cannot create the resource because it already exists, run the following command to delete the resource and perform this step again:

```
winrm delete winrm/config/Listener?Address=*+Transport=HTTPS
```

7. Run the following command to check whether the listening is successful in PowerShell:

```
winrm e winrm/config/listener
```

If the output contains **HTTPS**, the listening is successful.



Step 5 Ensure that security configuration has been performed by referring to **Security Configuration** to ensure successful connectivity verification.

----End

2.1.2.5 Configuring a Proxy Host

Before adding a proxy to a created host cluster, you need to perform the following operations on the proxy:

Configuring a Linux Proxy

Required Resources

You have configured the following resources in a Virtual Private Cloud (VPC):

Resource Type	Supported Resource Specifications	Quantity	Description
EIP	Bandwidth ≥ 5 Mbit/s	2	1. When creating a proxy, you need to add an ECS bound to an elastic IP address as the proxy. 2. When creating an SNAT gateway, you need to bind an EIP to it.

Procedure

Step 1 Enable the SSH forwarding function of the proxy.

- If the proxy mode is used, run the following command to check whether **AllowTcpForwarding** has been enabled for SSH on the proxy:
  ```
  grep AllowTcpForwarding "/etc/ssh/sshd_config"
  ```
- If the value is **no**, set it to **yes** and run the following command to restart the sshd service:
  ```
  service sshd restart
  ```

Step 2 Configure SNAT for the proxy.

1. Click ☰ in the upper left corner and choose **Networking > NAT Gateway** to access the console.
2. On the Network Console, click **Buy Public NAT Gateway**.
3. Enter related information and click **Next**.
4. After the NAT gateway is created, return to the NAT gateway list and click the target NAT gateway.
5. On the NAT gateway details page, click the **SNAT Rules** tab, click **Add SNAT Rule**, configure required parameters, and click **OK**.
6. Check whether the SNAT rule is added.

Step 3 Check routing policies.

1. Go to the console. In the upper left corner of the page, click ☰ and choose **Networking** > **Virtual Private Cloud** to access the network console.
2. Choose **Virtual Private Cloud** > **Route Tables** and click the target route table.

   Check the route information.

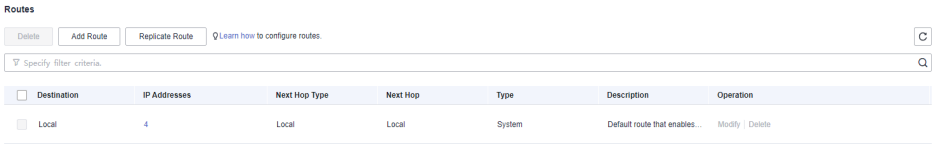


Table 2-8 Description of route information

Route Information	Description
Destination	Destination CIDR block. The default value is **0.0.0.0/0**. Select the IP address for the access environment based on project requirements.
IP Addresses	Click to view detailed information about the IP addresses.
Next Hop Type	Set it to **NAT gateway**.
Next Hop	Set it to the public NAT gateway that you have added the SNAT rule to.
Type	**System**: A system route is automatically added by the system and cannot be modified or deleted. **Custom**: A user-defined route is added by a user to direct traffic to a desired destination, and can be modified and deleted. The destination of a custom route cannot overlap with that of a system route.
Description	Description of the route.
Operation	You can modify and delete routes.

----End

Configuring a Windows Proxy

Required Resources

- A Windows host is available.
- The network connection between the proxy and hosts is normal.

Procedure

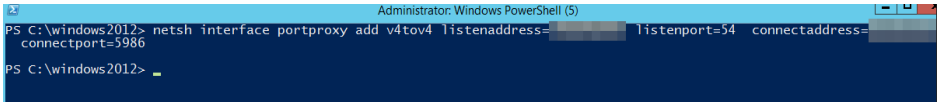
Step 1 Log in to the proxy, open PowerShell, and run the **netsh** command. Replace the parameters based on the parameter descriptions.

```
netsh interface portproxy add v4tov4 listenaddress=${proxy_ip}  listenport=${proxy_port}  connectaddress=${host_ip}  connectport=${host_port}
```

Table 2-9 Parameters

Parameter	Description
${proxy_ip}	Private IP address of the proxy.
${proxy_port}	Listening port of the proxy, for example, **54**.
${host_ip}	Private IP address of the host.

Parameter	Description
${host_port}	Port of the host. Generally, the port is **5986**.



Step 2 To **configure the security group and firewall**, enable the proxy listening port, that is, **${proxy_port}** in the preceding command.

----End

2.1.2.6 Self-hosted Resource Pool

This section introduces how to add your own hosts to the self-hosted resource pool.

Procedure

Step 1 Create an ECS.

1. Go to the console, choose **Service List** > **Compute** > **Elastic Cloud Server**. The **Elastic Cloud Server** page is displayed.
2. Click **Buy ECS** on the ECS console.
3. On the ECS configuration page, set parameters as prompted.
4. Click **Submit**

NOTE

You can apply for an EIP during ECS creation or by referring to **(Optional) Applying for an EIP**.

Configure a security group for the created ECS by referring to **Configuring a Security Group**.

Stop idle ECSs to avoid waste and unnecessary billing.

Step 2 Obtain the AK/SK.

1. On the console, click the username in the upper right corner and select **My Credentials** from the drop-down list.
2. Choose **Access Key** > **Create Access Key**.
3. Click **OK** to save the AK/SK.

Step 3 Create an agent pool.

1. On the CodeArts homepage, click the account name in the upper right corner and click **Account Settings**.
2. Choose **Agent Pool** > **Create Pool**, enter a **Pool Name**, set **Pool Type** to **DOCKER**, and click **Save**.

Step 4 Create an agent.

1. Click the created pool and click **Create Agent**. Enable **Install a JDK automatically**, **Install Git automatically**, and **Install Docker automatically**.
2. Enter the AK/SK obtained in **2**, and select **I have read and agree to the** *Privacy Statement* **and** *CodeArts Service Statement* **and understand that related configurations and authentication information will be used by CodeArts to perform operations with this service.**.
3. Click **Generate Command** to automatically generate the **Octopus Agent** command for installing the agent, then click **Copy Command**.

Step 5 Run the **Octopus Agent** command.

1. Log in to the ECS created in **1** and run the copied command in **/root** directory. The following information is displayed:
   ```
   End Install Octopus Agent,Agent output logs have been printed to [ /opt/octopus-agent/logs/octopus-agent.log ]
   ```
2. Check the status of the installed agent on the **Agents** page. If the status is **Idle**, the installation is successful.

   NOTE

   If the agent status is **Offline**, delete the agent and repeat steps 3 to 5.

----End

3 Managing Applications

3.1 Creating an Application

3.1.1 Prerequisites

CodeArts Deploy supports deployment on hosts, containers, microservice applications, and functions. This section describes how to create an application.

- You have the permission to create applications. For details, see **Application Permission Matrix**.
- A project is available. If no project is available, **create one**.

3.1.2 Creating an Application Using a Template

CodeArts Deploy provides deployment process templates (**system templates**) and allows you to save custom deployment processes as custom templates, enabling you to quickly create applications. This section describes how to create an application using a template.

Using a System Template

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Choose **CICD** > **Deploy**.

Step 3 Click **Create Application**. On the **Set Basic Information** page that is displayed, modify the basic information such as **Name**, **Project**, **Execution Host**, and **Description**, as required. For details, see **Editing the Basic Information**.

Step 4 After editing the basic application information, click **Next** and go to the **Select Template** page.

Step 5 Select a recommended template based on the service type. Click **OK**. The **Deployment Actions** page is displayed. The left pane is the action orchestration area, and the right pane is the action details area.

Step 6 (Optional) Click above or below an added action. All actions that can be added are displayed on the right of the page. You can add an action before or after the current action.

NOTE

- You can drag, add, and delete actions in the action orchestration area.
- You can click **Save as Custom Template** to save the current application as a custom template, which will be displayed under **Orchestration Template Management** > **Custom Templates**.

Step 7 (Optional) After adding an action, configure the action information. For details, see **Deployment Actions**.

Step 8 (Optional) After the action information is configured, switch to the **Basic Information** tab page and edit basic information as required. For details, see **Editing the Basic Information**.

Step 9 (Optional) After the preceding operations are complete, switch to the **Parameters** tab page, and create custom parameters as required. For details, see **Editing Parameters**.

Step 10 (Optional) After the preceding operations are complete, switch to the **Environment Management** tab page, and create and manage environments as required. For details, see **Editing the Environment**.

Step 11 (Optional) After the preceding operations are complete, switch to the **Permissions** tab page, and configure role permissions as required. For details, see **Editing Permissions**.

Step 12 (Optional) After the preceding operations are complete, switch to the **Notifications** tab page to notify users of application events through emails. For details, see **Editing Notifications**.

Step 13 After configuring all information, click **Save**.

----End

3.1.3 Customizing an Application

CodeArts Deploy allows you to customize applications. This section describes how to customize an application.

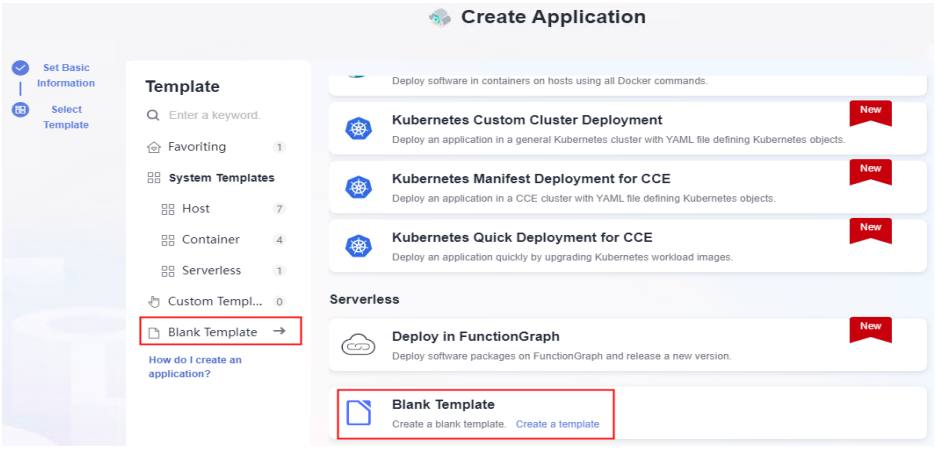
Procedure

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

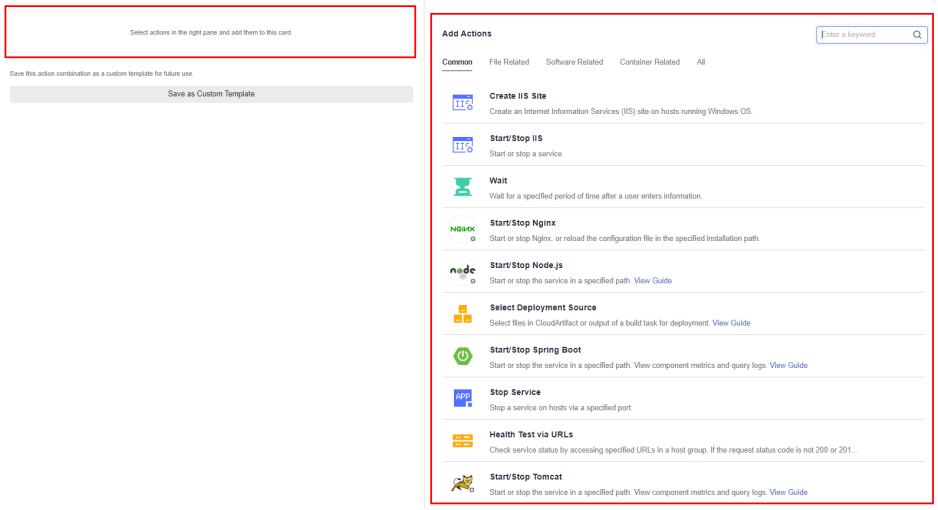
Step 2 Choose **CICD** > **Deploy**.

Step 3 Click **Create Application**. On the **Set Basic Information** page that is displayed, modify the basic information such as **Name**, **Project**, **Execution Host**, and **Description**, as required. For details, see **Editing the Basic Information**.

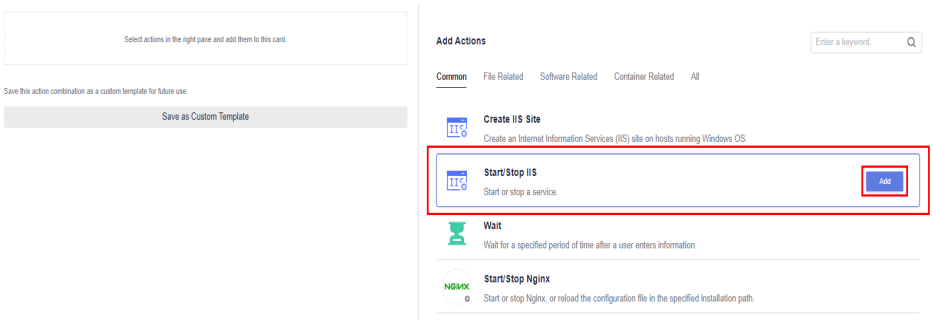
Step 4 After editing the basic application information, click **Next**. On the deployment template selection page that is displayed, select **Blank Template** and click **OK**.



The **Deployment Actions** page is displayed. The left pane is the action orchestration area, and the right pane is the list of actions.



Step 5 On the right list, click **Add** of the target action to add the action to the orchestration area.



Step 6 (Optional) Click  above or below an added action. All actions that can be added are displayed on the right of the page. You can add an action before or after the current action.

NOTE

- You can drag, add, and delete actions in the action orchestration area.
- You can click **Save as Custom Template** to save the current application as a custom template, which will be displayed under **Orchestration Template Management** > **Custom Templates**.

Step 7 (Optional) After adding an action, configure the action information. For details, see **Deployment Actions**.

Step 8 (Optional) After the action information is configured, switch to the **Basic Information** tab page and edit basic information as required. For details, see **Editing the Basic Information**.

Step 9 (Optional) After the preceding operations are complete, switch to the **Parameters** tab page, and create custom parameters as required. For details, see **Editing Parameters**.

Step 10 (Optional) After the preceding operations are complete, switch to the **Environment Management** tab page, and create and manage environments as required. For details, see **Editing the Environment**.

Step 11 (Optional) After the preceding operations are complete, switch to the **Permissions** tab page, and configure role permissions as required. For details, see **Editing Permissions**.

Step 12 (Optional) After the preceding operations are complete, switch to the **Notifications** tab page to notify users of application events through emails. For details, see **Editing Notifications**.

Step 13 After configuring all information, click **Save**.

----End

3.2 Editing an Application

3.2.1 Overview

CodeArts Deploy allows you to edit the **basic information**, **deployment actions**, **parameters**, **environment**, **permissions**, and **notifications** of an application. This section describes how to edit an application.

Prerequisites

- An application is available. If no application is available, create one by following the instructions provided in **Creating an Application**.
- You have the permission to deploy applications. For details, see **Application Permission Matrix**.

3.2.2 Editing the Basic Information

The **Basic Information** tab page contains the application name, project, and description. This section describes how to edit basic information.

Procedure

Step 1 Select the target application, click ***, and click **Edit**. The **Deployment Actions** page is displayed.

Step 2 Click **Basic Information**. You can edit **Name**, **Execution Host**, and **Description** as required.

Table 3-1 Parameters

Parameter	Description
Name	Mandatory. Name of an application.
Project	Retain the default value. Project to which an application belongs. If your account does not have a project, click **Create Project** and click **Scrum** to create one.
Description	Optional. Description of an application.
Execution Host	Optional. A resource pool is a collection of physical environments where commands are executed during software package deployment. You can use an **official resource pool** hosted by Huawei Cloud or host your own servers as a **self-hosted resource pool** on Huawei Cloud. For details about hosting your own servers, see **Self-hosted Resource Pool**.
Deploy from Pipeline	Optional. Toggling on the switch indicates that this application can run only in a pipeline. It cannot run independently.

NOTE

Note: If the tenant account has enabled **Intranet Secure Access** (only for whitelisted users), the self-hosted resource pool is selected by default and cannot be changed.

Step 3 After modifying all information, click **Save**.

----End

3.2.3 Editing a Deployment Action

On the **Deployment Actions** tab page, you can delete, edit, add, and drag deployment actions. This section describes how to configure deployment actions.

Procedure

Step 1 Select the target application, click ***, and click **Edit**. The **Deployment Actions** page is displayed.

- Add an action.

Click above or below an added action. All actions that can be added are displayed in the right pane. You can add an action before or after the current action.

- Modify the action details.

  Click the target action and modify the action details in the right pane.

- Copy, delete, or disable an action.

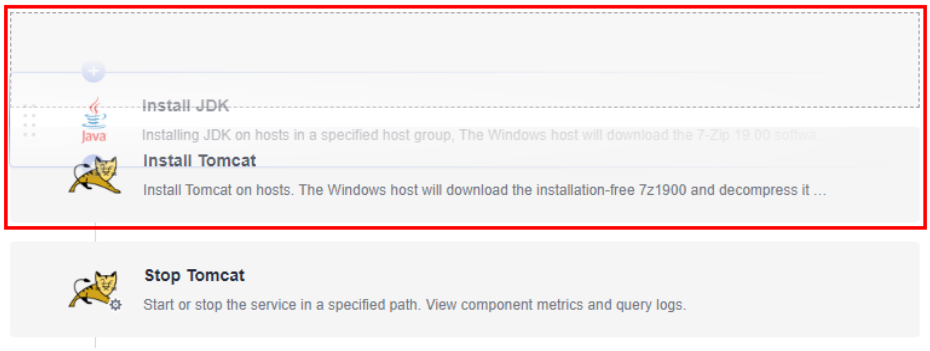
  Click and click **Clone** to clone the action to the current application.

  Click and click **Delete** to delete the action from the current application.

  Click and click **Disable** to disable the action in the current application.

- Adjust the execution sequence of actions.

  You can drag actions as required.



Step 2 After modifying all information, click **Save**.

----End

3.2.4 Editing Parameters

On the **Parameters** tab page, you can add, delete, and modify parameters. This section describes how to configure parameters.

Procedure

Step 1 Select the target application, click , and click **Edit**. The **Deployment Actions** page is displayed.

Step 2 Click **Parameters**.

- Edit a parameter.

  You can edit **Name**, **Type**, **Default Value**, **Private Parameter**, **Runtime Settings**, and **Description** of the existing parameters. You can also click to drag parameters to change their order.

- Add a parameter.

  Click **Create Parameter** to add a parameter. You can also click in the operation column to add a new parameter and edit it under the current parameter.

- Delete a parameter.

Click to delete an existing parameter.

NOTE

For details about the parameters, see **Managing Parameters**.

Step 3 After modifying all information, click **Save**.

----End

3.2.5 Editing the Environment

In the host deployment scenario, you can add, delete, and modify the environment of the application on the **Environment Management** tab page. This section describes how to configure the environment of the application.

Procedure

Step 1 Select the target application, click , and click **Edit**. The **Deployment Actions** page is displayed.

Step 2 Click the **Environment Management** tab.

Step 3 Create an environment.

1. Click **Create Environment**, set the following parameters, and click **Save**.

Parameter	Mandatory	Description
Environment	Yes	User-defined environment name.
Resource Type	Yes	You can choose **Host** or **Kubernetes** (available soon) based on the environment requirements.
OS	Yes	Choose **Linux** or **Windows** as the operating system for the host.
Description	No	Enter a description of the environment.

2. Click **Import Host**. The system automatically filters all clusters that meet the requirements of the current environment. Select a target host cluster and import a host to the environment in either of the following ways:

   Import separately: Click in the **Operation** column of a host to import the host to the environment.

   Import in batches: Select multiple hosts and click **Import**.

   NOTE

   If you import a target host bound to a proxy host, the proxy host will be imported to the environment automatically.

3. (Optional) After the host is imported, click on the page to verify the connectivity.

Step 4 Modify an environment.

1. Edit the environment.
2. Edit the host cluster.
   - **Editing an environment**: Click in the Operation column of an environment to modify the environment name and description.
   - **Deleting an environment**: Click in the Operation column of an environment, and click **Yes**.
   - **Managing permissions**: Click in the Operation column of an environment to configure operation permissions for each role. Enable or disable permissions as required.

Default Permissions of a Host Cluster					
Role/ Permission	**View**	**Edit**	**Delete**	**Deploy**	**Manage Permissions**
Environment creator	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)
Project creator	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)
Project manager	Yes	Yes	Yes	Yes	Yes
Product manager	Yes	No	No	No	No
Test manager	Yes	No	No	No	No
O&M manager	Yes	Yes	Yes	Yes	Yes
System engineer	Yes	Yes	Yes	Yes	No
Committer	Yes	Yes	Yes	Yes	No
Developer	Yes	Yes	Yes	Yes	No
Tester	Yes	No	No	No	No
Participant	Yes	No	No	No	No
Viewer	Yes	No	No	No	No

3. Edit the host in the environment.
   - **Verifying host connectivity in batches**: Select multiple hosts and click Verify Connectivity.
   - **Enabling network connectivity verification**: Click in the **Operation** column of a host.
   - **Deleting a host**: Click in the **Operation** column of a host, click **Delete**, and click **OK**.

     NOTE

     A proxy host cannot be deleted directly. A proxy host is deleted, when its last target host is deleted from the environment.

Step 5 After modifying all information, click **Save**.

----End

3.2.6 Editing Permissions

On the **Permissions** tab page, you can manage and control application permissions. This section describes how to configure application permissions.

Procedure

Step 1 Select the target application, click , and click **Edit**. The **Deployment Actions** page is displayed.

Step 2 Click the **Permissions** tab.

Configure operation permissions for each role as required.

- indicates that the permission is enabled. You can click it again to disable the permission.
- indicates that the permission is disabled. You can click it again to enable the permission.
- indicates that the permission is enabled and cannot be changed.

NOTE

- An application or project creator's permissions cannot be changed.
- If you do not have the **Edit** permission, the editing page cannot be displayed.
- If you have the **Edit** permission but do not have the **Assign Permissions**, the **Edit**, **Delete**, **View**, **Run**, and **Clone** buttons are unavailable.
- Project administrators (project creators and PMs) can customize roles and edit the permissions of custom roles.

Step 3 After modifying all information, click **Save**.

----End

Application Permission Matrix

The following table lists the permission matrix of applications.

Default Application Permissions									
Role/ Operation	**Default Permissions**								
	Create	**View**	**Edit**	**Delete**	**Run**	**Clone**	**Disable**	**Create Environment**	**Assign Permissions**
App creator	-	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)
Project creator	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)	Yes (cannot be changed)
Project manager	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Product manager	Yes	Yes	No	No	No	No	No	No	No
Test manager	No	Yes	No	No	No	No	No	No	No
O&M manager	No	Yes	No	No	No	No	No	Yes	No
System engineer	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Committer	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No
Developer	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No
Tester	No	Yes	No	No	No	No	No	No	No

Default Application Permissions									
Participant	No	Yes	No	No	No	No	No	No	No
Viewer	No	Yes	No	No	No	No	No	No	No

NOTICE

- Roles with the **Assign Permissions** permission can modify the permission matrix, but permissions of the **Project creator** and **App creator** roles cannot be modified.
- Only the **Project creator**, **Project manager**, and **Developer** roles have the permission to create applications.

3.2.7 Editing Notifications

On the **Notifications**Notifications tab page, you can set notification rules to send events to the application creator, executor, and members who have followed the application via emails. This section describes how to configure notifications.

Procedure

Step 1 Select the target application, click , and click **Edit**. The **Deployment Actions** page is displayed.

Step 2 Click **Notifications**.

Step 3 To set event notifications, click (indicates that the notification is enabled) or (indicates that the notification is disabled) as required.

- **Email**: pushes application activities by email to the application creator, executor, and members who have followed the application.

Step 4 After modifying all information, click **Save**.

----End

3.3 Deploying an Application

You can deploy applications using the following methods:

- **Re-deployment**: Deploy an existing application again. This method applies to the scenario where the original application configurations are used to complete deployment.
- **Rollback**: Select a historical deployment record to roll back the application. Deployment records of the last 31 days can be retained. This method applies to the scenario where the historical application configurations are used to complete deployment.

Prerequisites

- An application is available. If no application is available, create one by following the instructions provided in **Creating an Application**.
- You have the permission to deploy applications. For details, see **Application Permission Matrix**.
- A deployment record is available for the target application in the rollback scenario.

New deployment

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

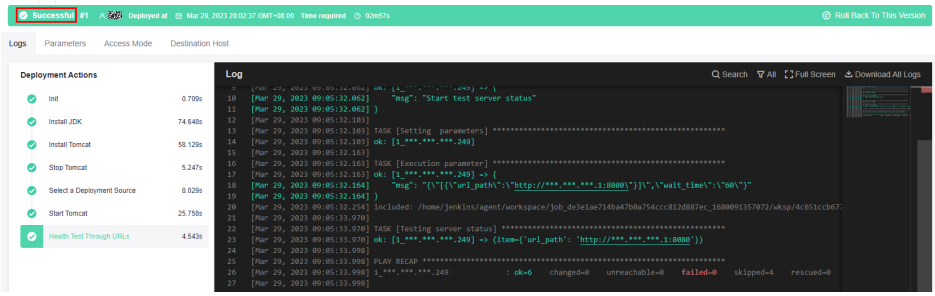
Step 2 Choose **CICD** > **Deploy**.

Step 3 Select the target application in the application list and click  .

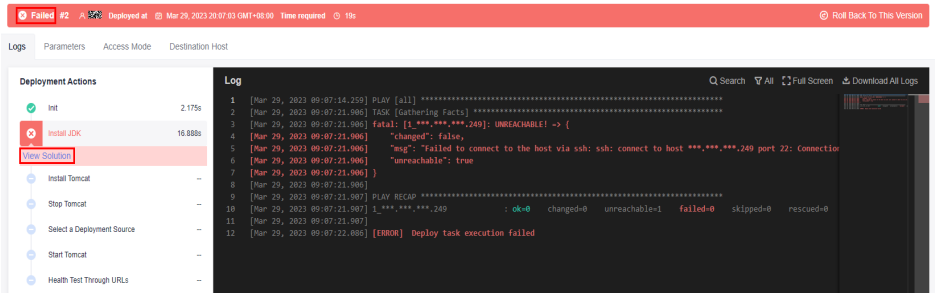
Step 4 If runtime parameters are configured, the parameter setting dialog box is displayed. Enter the parameter values and click **OK** to deploy the application. For details, see **Managing Parameters**.

Step 5 After the deployment is complete, click the application name and click the target deployment record. The application status bar changes to green and the message **Successful** is displayed, indicating that the application is successfully deployed.

The following figure shows that the deployment is successful.



If the application status bar turns red and displays **Failed**, the application fails to be deployed. In this case, click **View Solution**.



NOTE

Each time an application is deployed, a version record is added. The record with the largest ID is the latest deployment record. Deployment records of the last 31 days can be retained.

----End

Rollback Deployment

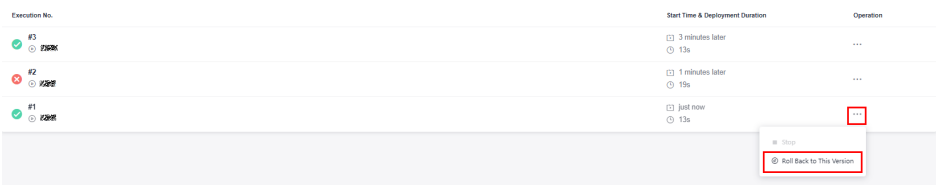
Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Choose **CICD** > **Deploy**.

Step 3 Click the target application name in the application list. The application deployment records are displayed.

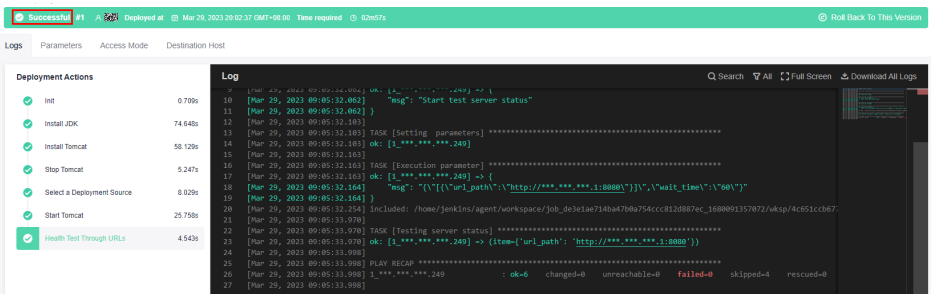
Step 4 Select the target version and click **Roll Back to This Version**. In the displayed dialog box, click **OK**.

The following uses version 2 as an example.

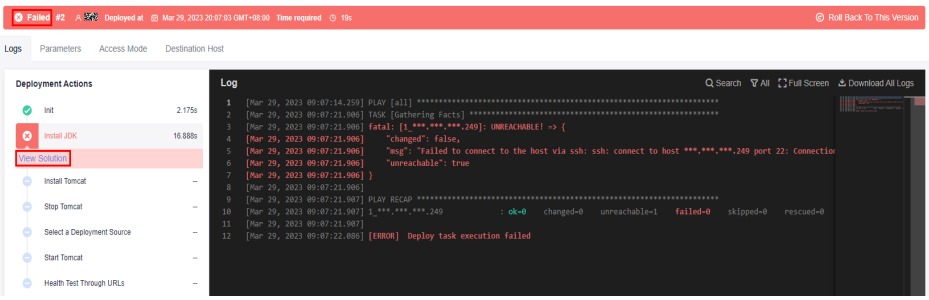


Step 5 After the deployment is complete, click the deployment record. The application status bar changes to green and the message **Successful** is displayed, indicating that the application is successfully deployed.

The following figure shows that the deployment is successful.



If the application status bar turns red and displays **Failed**, the application fails to be deployed. In this case, click **View Solution**.



NOTE

Each time an application is deployed, a version record is added. The record with the largest ID is the latest deployment record. Deployment records of the last 31 days can be retained.

----End

3.4 Viewing an Application

This section describes how to view the information about a created application, including the deployment records and configuration details of the application.

- Deployment records: You can view the deployment records of an application from the last 31 days.
- Configuration details: You can view the latest configurations of an application.
- Operation records: You can view the operation records of an application in the last year.

Prerequisites

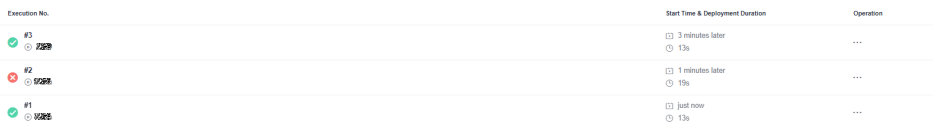
- An application is available. If no application is available, create one by following the instructions provided in **Creating an Application**.
- You have the permission to view applications. For details, see **Application Permission Matrix**.
- The target application has a deployment record.

Procedure

Each time an application is deployed, a version record is added. The record with the largest ID is the latest deployment record. You can check information about deployed applications, such as **Logs**, **Parameters**, **Access Mode**, and **Environment**.

Step 1 Click the target application name in the application list. The application deployment records are displayed.

Step 2 View the historical versions of the application, which are sorted by deployment time from newest to oldest.



You can click an application version in the preceding figure to view its **Logs**, **Parameters**, **Access Mode**, and **Environment**.

Table 3-2 Parameters

Parameter	Description
Logs	Deployment log information. **NOTE** • By default, all logs of the application are displayed. Click a deployment action to view its log. • You can click **Full Screen** in the upper right corner of the log window to maximize the log window, and click **Exit Full Screen** to exit the full screen. • You can click **Download All Logs** to download all logs to a local directory.
Parameters	Parameters in the **Runtime Parameter** dialog box when the application is deployed.
Access Mode	This parameter is available only when **Deployment Actions** of an application contain **Health Test via URLs**. You can add this action to test the service status by accessing a URL on a specified host.
Environment	Environment where the application is deployed in the host deployment scenario.

Step 3 Go back to the deployment records page and switch tabs to view the latest **Basic Information**, **Deployment Actions**, **Parameters**, **Environment Management**, **Permissions**, and **NotificationsNotifications**.

Step 4 Click the ⋮ icon next to **Deploy** and click **History** in the drop-down list to view the operation records in the last year. You can also return to the application list, select a target application, click the ⋯ icon, and click **History** to access the page.

Parameter	Description
Operator	Nickname of the operator.
Type	Operation type: **Create** and **Edit**
Last Modified	Time when the operation is performed.
Object	Object on which the operator performs operations. Options: **Application**, **Deployment Actions**, **Parameters**, **Permissions**, and **Notifications**.

----End

3.5 Managing Groups

Users can manage applications of the same features by sorting applications to user-defined groups based on functions or environments. For example,

applications can be classified into categories such as a production environment, a development environment, and a test environment based on environments.

Prerequisites

By default, only the project creator, project admin, project manager, system engineer, committer and developer have the group management permissions. The project creator and project manager can assign the group management permission to other roles.

Procedure

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Choose **CICD** > **Deploy**.

Step 3 Move the cursor to **All Groups**. The icon is displayed. Click to expand the deployment group panel.

Step 4 Click . The **Manage Groups** dialog box is displayed.

Step 5 Move the cursor to the row where **All Groups** is located and click .

Step 6 Enter the group name. Click to create the group or click to cancel.

After the group is created, you can perform the following operations:

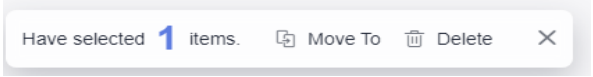
- Click in the row where the group is located to create a subgroup. You can create a maximum of three levels of subgroups.
- Click in the row where the group is located to change the group name.
- Click in the row where the group is located to move or delete the group.

NOTE

After the first group is created, an **Ungrouped** group is automatically generated. New applications and ungrouped applications are automatically added to the **Ungrouped** group. If no group is selected when creating an application, the newly created and ungrouped applications are automatically added to **Ungrouped**.

Step 7 After all groups are created, click **Close** to return to the application list page. You can move applications to the corresponding groups as required.

1. Select the applications to be grouped. The following dialog box is displayed at the bottom of the page.



2. Click **Move To**. The **Move Group** dialog box is displayed. You can move the application to the corresponding group.

----End

3.6 Following/Cloning/Deleting Applications

This section describes how to follow, clone, delete, and disable an application.

Prerequisites

- An application is available. If no application is available, create one by following the instructions provided in **Creating an Application**.
- You have the permission to view, clone, delete, and disable applications. For details, see **Application Permission Matrix**.

Following an application

If there are many applications in the application list, you can follow an application to pin it on the top of the application list. If you follow multiple applications, the applications are displayed on the top from newest to oldest based on the time when they are followed.

On the **Applications** page, click ☆ next to the target application to follow it.

Cloning an application

You can clone an application without affecting the original application.

On the **Applications** page, click ··· next to the target application and click **Clone**. The **Deployment Actions** page is displayed. Click **Basic Information** to change the application name and click **Save**.

Deleting an application

You can delete an application that is no longer needed.

On the **Applications** page, click ··· next to the target application and click **Delete**. In the displayed dialog box, enter the application name and click **Yes**.

⚠ CAUTION

Note that the application cannot be restored after being deleted.

3.7 Deployment Actions

3.7.1 Health Test via URLs

This action aims to access a URL on a target host to test the service status. The following table shows the configuration information.

Table 3-3 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Retries	If a service does not start up when the health test reaches the maximum retry times, the service fails this test.
Interval (s)	Interval between two retries.
Test Path	Health test URL. Multiple URLs can be added.
Control Options	• Enable this action. • Continue the task even if this action fails.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.2 Selecting a Deployment Source

Select the software package path; or download the software package corresponding to the build record from Artifact to the target environment.

Table 3-4 Configuration parameters when Artifact is the source

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Source	**Artifact** and **Build task** are available.
Environment	Target environment.
Software package	Select a software package to be deployed from the Artifact.
Download Path	Path for downloading the software package to the target host.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

Table 3-5 Configuration parameters when software package is the source

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.

Parameter	Description
Source	**Artifact** and **Build task** are available.
Environment	Target environment.
Build Task	Target build task. If there is no build task, **create one**.
Artifact Filtering Mode	**Build version** and **Build branch** are available.
Build No.	Sequence number of the target build task.
Download Path	Path for downloading the software package to the target host.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.3 Stopping a Service

This action aims to stop a service with a specified port. The following table shows the configuration information.

Table 3-6 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Service Port Number	Port of the service to stop.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.4 Copying a File

This step supports file copying between directories within hosts and file copying across environments. The configuration is as follows:

Table 3-7 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Copy Mode	• To copy files from one directory to another directory on the same host, select **Within host**. • To copy files from one host to another host, select **Across hosts**.
Environment	The environment where applications will be copied.
Target environment	If **Copy Mode** is set to **Across hosts**, this parameter indicates the target environment. **CAUTION** If an environment contains multiple hosts, copy files from all hosts in the target environment.
Files	Specify the source path and destination path of the file to copy. Both paths must be absolute paths.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.5 Deleting a File

This action aims to delete a file or folder from a host in a specified environment. The following table shows the configuration.

Table 3-8 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
File Path	Path of the file or folder to delete. **NOTE** If a file path is specified, the file is deleted. If a folder path is specified, the folder and all files in the folder are deleted.

Parameter	Description
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.6 Decompressing a File

This action aims to decompress a file from one path on a host to another path on the host. The following table shows the information configuration.

Table 3-9 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Decompress File	Path of the file to be decompressed or stored.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.7 Modifying a Configuration File

This action aims to modify the specified content in a file by identifying specific identifiers. The following figure shows the configuration page.

Table 3-10 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.

Parameter	Description
Absolute Path	Absolute path of the configuration file to modify. • Modify a single file, for example, **/usr/local/server.config**. • Modify multiple files, for example, **/usr/local/server.config;/usr/local/a.txt**. • The wildcard character (*) can be used, for example, **/usr/local/*.config**. However, you cannot separate multiple wildcards using semicolons (;), for example, **/usr/local/*.config;/usr/local/*.txt**.
Prefix and Suffix	Parameter reference flag. If no prefix or suffix is matched, the configuration file remains unchanged and no error is reported in logs.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

Configuration Example

To change the service port, perform the following steps:

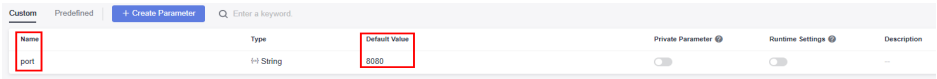
Step 1 Open the configuration file and view the content.

Figure 3-1 Viewing the configuration file

```
ServerPort=${port}
UserName=#{name}#
```

Step 2 Change the prefix label and suffix label. For example, change the prefix to **${** and the suffix to **}**.

Step 3 On the **Parameters** tab, set **Name** and **Default Value**.



Step 4 Save the configuration and deploy the application.

Step 5 After the deployment is complete, open the configuration file again.

The value of **${port}** is changed to **8080**.

Figure 3-2 Viewing the configuration file

```
ServerPort=8080
UserName=#{name}#
```

----End

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.8 Starting or Stopping Tomcat

Start or stop a service in a specified path. In addition, you can monitor metrics and query logs of components after deployment. The following table shows the information configuration.

Table 3-11 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Operation	**Start** and **Stop** are available.
Absolute Path	Absolute path of the Tomcat service.
HTTP Port	HTTP port of the Tomcat service.
AJP Port	AJP port of the Tomcat service.
Service Shutdown Port	Shutdown port listened by the Tomcat service.
Waiting Time	The time required for starting the service. If you select **Start** for **Operation**, the system checks the process during the startup to determine whether the service is started successfully. You can adjust the time based on the actual time required for starting the service. If the time is improper, the check result is invalid.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.9 Starting or Stopping Spring Boot

This action aims to start or stop the Spring Boot service in a specified path in an environment. The following table shows the configuration.

Table 3-12 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Operation	**Start** and **Stop** are available.
Absolute Path	Absolute path of the Spring Boot service.
System Variables	• Optional. • Java running parameters. JVM variables are used. The commonly used parameter is **-D**. • The **-XX** and **-X** parameters are used to set the memory and GC parameters, respectively. The parameter settings may vary according to JVMs. • The **-D** and **-X** parameters are followed by Java. When starting the service, you can set the memory required for service running. **NOTE** The common parameter format is **-Dfile.encoding=UTF-8 -Xms256m -Xmx512m**.
Command Parameters	• Optional. • Spring Boot running parameters, that is, application parameters. • If you choose to start the service, you can use the -- parameter to set the listening port of the Spring Boot service. **NOTE** The common parameter format is **--server.port=9000 --spring.profiles.active=prod**.
Waiting Time	Time for waiting for the service to start. If you choose to start the service, the system checks the process during the startup to determine whether the service is started successfully. You can adjust the time based on the actual time required for starting the service. If the time is improper, the detection fails.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.10 Starting or Stopping Nginx

This action aims to start or stop the Nginx service in a specified path in an environment. The following table shows the configuration.

Table 3-13 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Operation	**Start Nginx**, **Reload configuration file**, **Stop Nginx immediately**, and **Quit Nginx gracefully** are available.
Nginx Installation Path	Enter the installation path of the Nginx service in the target environment.
Modify configuration file before execution	Enable or disable this function based on whether to modify the Nginx configuration file on the target host.
Nginx Configuration File Path	Path of the Nginx configuration file on the target host.
Configuration File Backup Path	Target path for backing up the original Nginx configuration file on the target host.
Configuration File Content	Content of the new configuration file.
Action Control	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.11 Starting or Stopping IIS

This action aims to start or stop IIS by specifying a name. The following table shows the configuration.

Table 3-14 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Operation	**Start** and **Stop** are available.
Service Name	Enter the name of the target service.
Control Options	• Enable this action. • Continue the task even if this action fails.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.12 Starting or Stopping Node.js

This action aims to start or stop the Node.js service in a specified path in a host. The following figure shows the configuration.

Table 3-15 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Operation	**Start** and **Stop** are available.
Absolute Path	Absolute path of the Node.js service.
Command Parameters	Optional. • Node.js running parameters, that is, application parameters. • If you choose **Start** for **Operation**, you can configure parameters, such as the listening port of Node.js to start Node.js.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.13 Starting or Stopping the Go service

This action aims to start or stop a service in a specified path.

Table 3-16 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Operation	**Start** and **Stop** are available.
Absolute Path	Installation path of Go.
Waiting Time	Time for waiting for the service to start. If you choose to start the service, the system checks the process during the startup to determine whether the service is started successfully. You can adjust the time based on the actual time required for starting the service. If the time is improper, the detection fails.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

3.7.14 Running Shell Commands

This action aims to run shell scripts on a host in a specified environment. The following table shows the configuration.

Table 3-17 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Shell Commands	Bash scripts to run.

Parameter	Description
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

Example: Using Shell Commands to View Service Logs

After application deployment is complete, you can run the shell commands to view the service startup or execution logs.

Preparations

1. Ensure that you are an authorized user of a host. Only authorized users have the permissions required to view service startup or execution logs.
2. Determine the full path of the service startup log.

   NOTE

   The following describes how to install the Tomcat service:

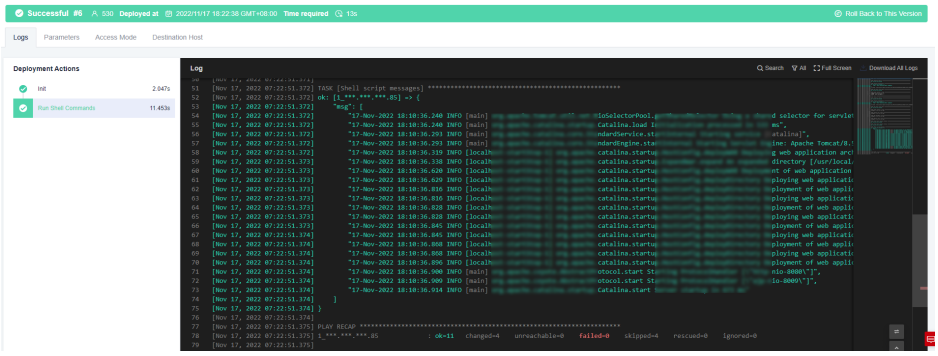
   Full path of the service startup log: **/usr/local/tomcat/apache-tomcat-8.5.38/logs/catalina.out**

Procedure

Step 1 Run the **tail** command to query the service startup or execution logs.

Step 2 Run the following command to query the last 20 lines of the log. The following figure shows the command output.

```
tail -n 20 /usr/local/tomcat/apache-tomcat-8.5.38/logs/catalina.out
```



----End

NOTE

Do not run the **cat** command when running the shell command to view files. If the log file is too large, it may take some time to load data. Do not use the **tail –f** command.

If the shell command to be executed contains more than 10,240 characters, you are advised to **Run Shell Script** extension.

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.15 Running Shell Scripts

This action aims to run shell scripts on a host in a specified environment. The following table shows the configuration.

Table 3-18 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Running Mode	**Default** and **Background** are available. **NOTE** • Default: The result is printed, but the related service or process cannot be started. • Background: The service or process can be started, but the result will not be printed.
Shell Script Path	Path of the shell script on the target host.
Running Parameters	Before executing the script, set parameters. During script execution, the entered parameter values are loaded and used.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

How Do I Use Script Execution Parameters?

Step 1 Use **$1**, **$2**, and similar formats in the shell script to reference parameter. For example, the content of the **hello.sh** script is shown in the following figure.

Figure 3-3 Script content

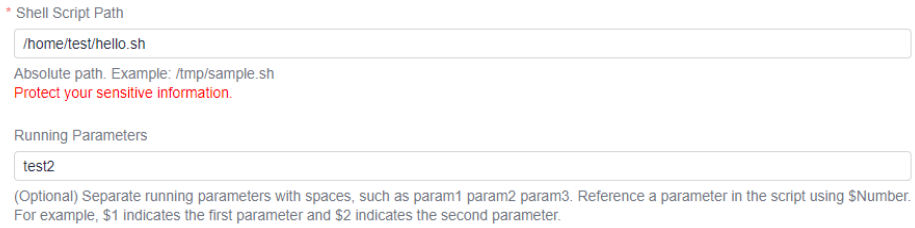
```
[root@SZX1000478390 test]# pwd
/home/test
[root@SZX1000478390 test]# ls
hello.sh
[root@SZX1000478390 test]# cat hello.sh


#!/bin/bash
#test execution parameters

cd /home/test
mkdir $1
```

Step 2 Separate running parameters with spaces, as shown in the following figure.

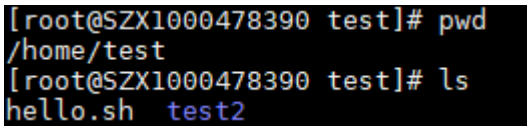
Figure 3-4 Entering running parameters



Step 3 View the result.

In the script, **$1** is replaced with **test2**, and the **test2** directory is created.

Figure 3-5 Viewing the result



----End

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.16 Running PowerShell Commands

This action aims to run the PowerShell commands in a specified path on a Windows host. The following table shows the configuration page.

Table 3-19 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
PowerShell Commands	Specifies the command to be executed.
Control Options	• Enable this action. • Continue the task even if this action fails.

NOTE

If the PowerShell command to be executed contains more than 10240 characters, you are advised to **run the PowerShell script** extension.

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.17 Running PowerShell Scripts

This action aims to run the PowerShell scripts in a specified path on a Windows host. The following figure shows the configuration page.

Table 3-20 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Running Mode	**Default** and **Background** are available. **NOTE** • Default: The result is printed, but the related service or process cannot be started. • Background: The service or process can be started, but the result will not be printed.
Script Path	Path of the script on the target host.
Running Parameters	Before executing the script, set parameters. During script execution, the entered parameter values are loaded and used.
Control Options	• Enable this action. • Continue the task even if this action fails.

How Do I Use PowerShell Script Execution Parameters?

Step 1 Use **param($a,$b)** at the beginning of the script to declare variables **a** and **b**.

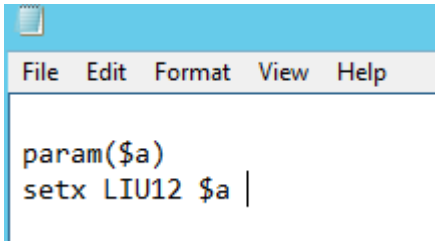
Step 2 Use variables **$a** and **$b** in the script.

Step 3 When running the script, enter the values of variables **a** and **b** in the script running parameters and separate them with spaces.

Example:

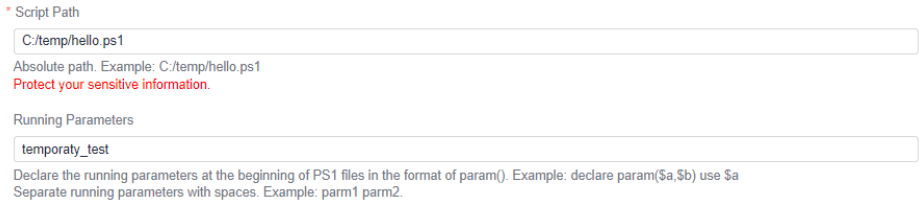
The following figure shows the content of the **hello.ps1** script. Set an environment variable to import a temporary value.

Figure 3-6 Script content



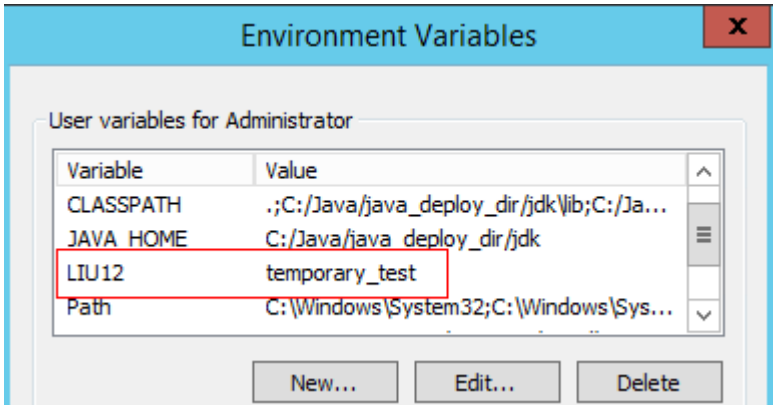
Configure parameters.

Figure 3-7 Settings



Step 4 View the result.

Figure 3-8 Result



----End

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.18 Running Docker Commands

This action aims to run Docker commands on a host to build, push, pull, and run images. The following describes how to configure each command.

login and logout

Step 1 Search for and add action **Run Docker Command**.

Step 2 Select **login** or **logout** for **Command**.

NOTE

Only self-hosted and SWR repositories are supported. You are advised to use the action **Run Shell Commands** to log in to or log out of a public repository.

When running the **login** command, retain the default value **No** for **Restart Docker**.

When you log in to a private repository, Docker provides a valid credential of the private repository in the ***.docker/config.json*** file. By default, the credential is encoded using Base64. You are advised to use **docker-credential-pass** and **gpg** to enhance Docker security.

Step 3 Select the image repository to be logged in to or logged out of. If no image repository is available, click **Create**.

The **Create Service Endpoint: Docker repository** dialog box is displayed, as shown in the following figure.

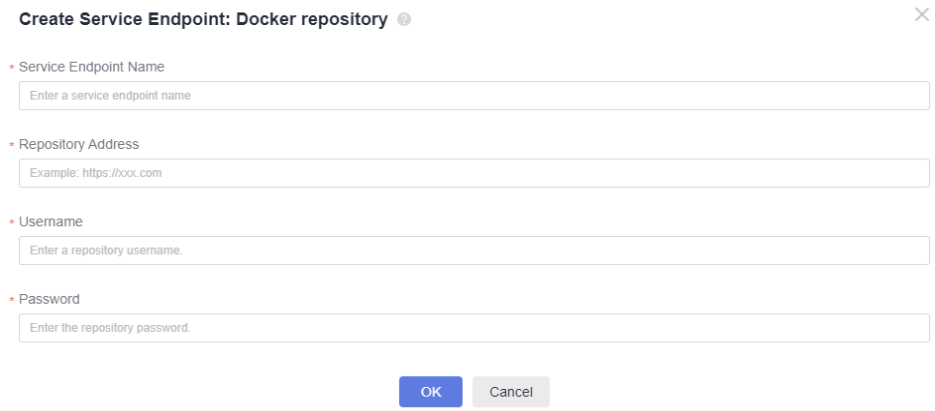
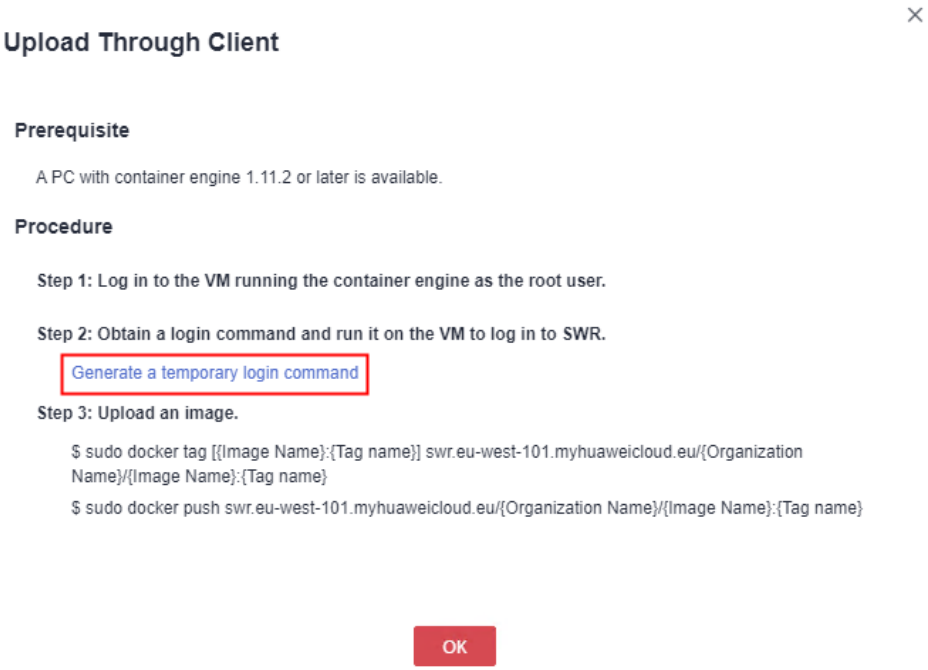


Table 3-21 Parameters

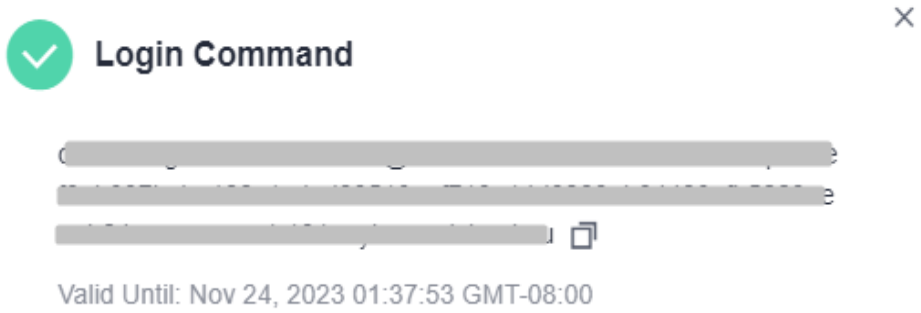
Parameter	Description
Service Endpoint Name	Name of the service endpoint to the image repository. The name facilitates service endpoint selection and management.
Repository Address	Address of the image repository. You can use a self-hosted or SWR repository. **NOTE** The image repository address cannot contain the organization name or image name. The repository address is in the **https://*XXXX*.com** or **http://*XXXX*.com** format.
Username	Username for logging in to the image repository.
Password	Password for logging in to the image repository.

Step 4 Log in to SWR.

1. Log in to the console. In the upper left corner of the page, choose **Service List** > **Application** > **SoftWare Repository for Container**. On the SWR console, choose **My Images** > **Upload Through Client**.
2. In the displayed dialog box, click **Generate a temporary login command**.



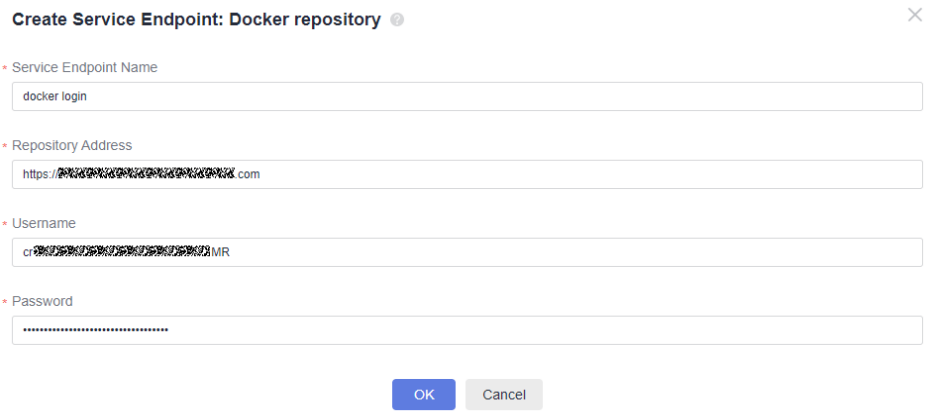
3. This topic uses the temporary command as an example. After you click **Generate a temporary login command**, the following dialog box is displayed.



NOTICE

- **-u** is followed by the username.
- **-p** is followed by the password.
- **swr.*XXXXX*.com** is the repository address.

4. On the deployment page, add the service endpoint.



NOTE

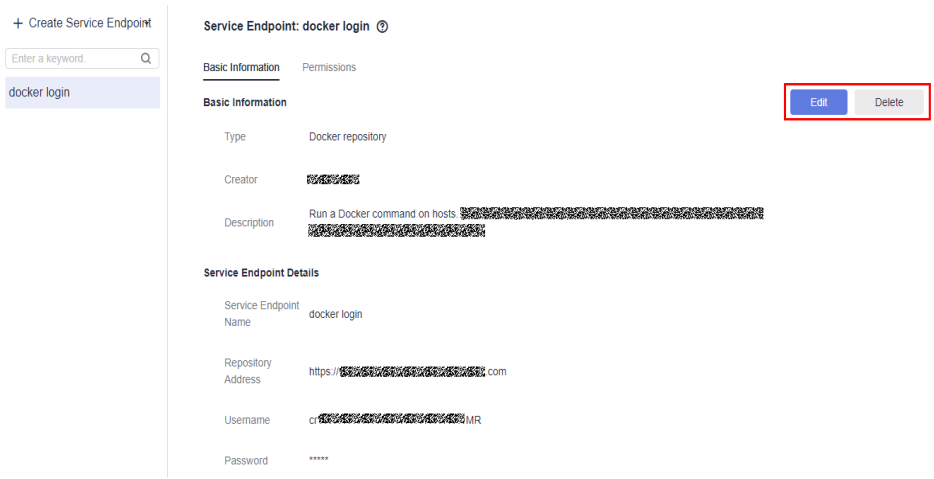
The repository URL must be in the **https://*XXXX*.com** or **http://*XXXX*.com** format.

The image repository address obtained by running the temporary command must be prefixed with **https://** or **http:**.

5. Click **OK**.

   Then you can run the **login** command to log in to the image repository on CodeArts Deploy.

Step 5 To modify the information about an image repository, click **Manage** next to **Image Repository** in the action **Run Docker Commands**. On the displayed page, edit or delete the repository.



----End

build

Preparations

To create a Docker image, upload the created **Dockerfile** to the target host in action **Select Deployment Source**.

Configuration Method

Step 1 Search for and add action **Run Docker Command**.

Step 2 Select the **build** command, enter the path where the Dockerfile is stored on the target host, and enter the execution parameters of the **build** command.

----End

tag

Step 1 Search for and add action **Run Docker Command**.

Step 2 Select the **tag** command, enter the image to be tagged, and set the execution parameters (optional) of the **tag** command.

NOTE

To add tags to multiple groups of images, separate the tags using newline characters.

----End

run

Step 1 Search for and add action **Run Docker Command**.

Step 2 Select the **run** command and enter the execution parameters of the **docker run** command.

NOTE

When running the **run** command, you cannot create or start a container in interactive mode. Instead, you must add the **-d** execution parameter so that the command can run in the background.

----End

Others

1. Search for and add action **Run Docker Command**.
2. Select a command (**push**, **pull**, **start**, **stop**, **restart**, **rm**, or **rmi**) and enter the execution parameters of the command.

   The command output similar to the following is displayed:

```
push: docker.test-registry.com/branch/Ubuntu:v1
pull: docker.test-registry.com/branch/Ubuntu:v1
rm: -f db01 db02
rmi: -f docker.test-registry.com/branch/Ubuntu:v1
start/stop/restart: container ID or name
```

   NOTE

   If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.19 Wait

This action aims to control the time between two adjacent actions.

Table 3-22 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Waiting Time (s)	Waiting time between two adjacent actions.
Control Options	• Enable this action. • Continue the task even if this action fails.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.20 Deploying on Kubernetes

This step uses kubectl commands to control your Kubernetes cluster and perform deployment operations. Currently, deployment on Kubernetes consists of the following three phases. Select a proper solution based on project requirements.

- **Deploying an Application in Kubernetes (CCE Cluster) Using Manifest**
- **Deploying an Application in Kubernetes (CCE Cluster) Quickly**
- **Deploying an Application with a Custom Kubernetes Cluster**

3.7.21 Deploying an Application in Kubernetes (CCE Cluster) Using Manifest

This section introduces how to deploy an application in a Huawei Cloud CCE cluster with manifest file defining Kubernetes objects.

Prerequisites

A **CCE cluster** is available.

Procedure

Table 3-23 Parameters

Parameter	Description
Action Name	This parameter is mandatory. After the action name is added, it will be displayed in the **Deployment Actions** orchestration area. **NOTICE** The action name can contain 1 to 128 characters, including letters, digits, hyphens (-), underscores (_), commas (,), semicolons (;), colons (:), slashes (/), parentheses, and spaces. However, it cannot start or end with a space.

Parameter	Description
File Source	This parameter is mandatory. **Artifact**, **Repo**, or **obs** can be selected. Here, **Artifact** has been selected.
Manifest File	This parameter is mandatory. Select the manifest files to be deployed. These files must be suffixed with .yaml, .yml, or .json. Click ... . On the file selection page that is displayed, select the manifest file to be deployed in Artifact. By default, the project cannot be changed. You can search for the manifest file by keyword or upload the local manifest file to the repository, click  , refresh the repository file, and select the manifest file. Click  .
Tenant	This parameter is mandatory. There are two options: • **Current**: The software package is deployed in the CCE cluster of the current tenant for release. Select **Current**. The current tenant must have the CCE cluster operation permission. If the current tenant does not have the CCE cluster operation permission, select **Authorized User** for deployment. • **Other**: The software package is deployed and published in the CCE cluster of another tenant in IAM authorization mode. If you select **Other**, you must select an authorized tenant to deploy the CCE cluster. **NOTE** You are advised to configure the AK/SK of a member account that has the **CCE cluster operation permission** and not advised to configure the AK/SK of a tenant account.
Authorized User	This parameter is optional. If you do not have the permission to execute an API, this parameter enables you to obtain the temporary AK/SK of the parent user to execute the CCE API through IAM.
Region	Select the region for deployment.
Cluster Name	Select the Kubernetes cluster applied on CCE.
Namespace	Select the namespace of the Kubernetes cluster on CCE.
Control Options	Continue the task even if this action fails.
Overtime	Maximum execution duration of an action, in minutes. If the task duration exceeds the specified time before you stop the application, the action will be timed out. Value range: 1–30

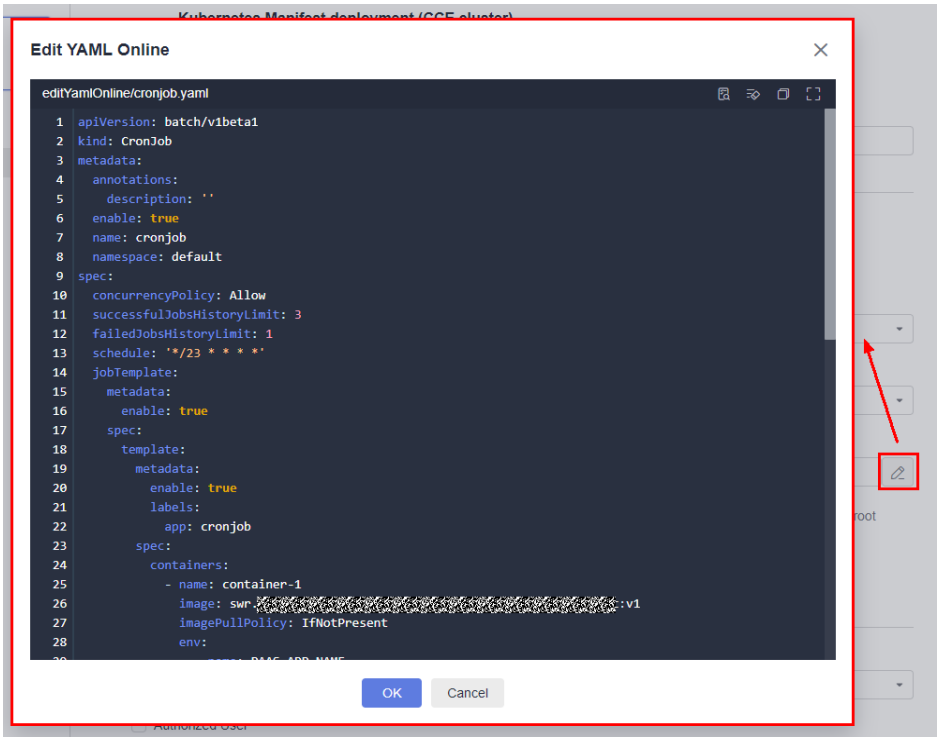
Online Manifest File Editing

When **File Source** is set to **Repo**, you can edit the manifest file online. Click the icon next to **Manifest File** to go to the online editing page.

NOTE

To modify the manifest file in Repo, you must have the corresponding member permissions. For details, see section **"Repository Member Permission"** of CodeArts Repo.

- Click the icon to perform basic syntax verification on the content of the manifest file to better optimize your code.
- Click the icon to optimize the manifest file format.
- Click the icon to copy all the content of the manifest file.
- Click the icon to display the content of the manifest file in full screen to better browse the code.



3.7.22 Deploying an Application in Kubernetes (CCE Cluster) Quickly

This section introduces how to deploy an application quickly by upgrading Kubernetes workload images.

Prerequisites

A **CCE cluster** is available.

Procedure

Table 3-24 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Deployment Tenant	• **Current tenant**: The software package is deployed in the CCE cluster of the current tenant for release. Select **Current tenant**. The current tenant must have the CCE cluster operation permission. If the current tenant does not have the CCE cluster operation permission, select **Authorized User** for deployment. • **Other**: The software package is deployed and published in the CCE cluster of another tenant in IAM authorization mode. If you select **Other**, you must select an authorized tenant to deploy the CCE cluster. **NOTE**
Escalate privilege with IAM	If you do not have the permission to execute an API, this parameter enables you to obtain the temporary AK/SK of the parent user to execute the CCE API through IAM.
Region	Select the region for deployment.
Cluster Name	Select the Kubernetes cluster applied on CCE.
Namespace	Select the namespace of the Kubernetes cluster on CCE.
Workload	Select the target Deployment.
Pods	Enter the number of instances to be deployed. **NOTE** If blank, the current number of pods in the CCE cluster will be adopted.
Container	Select the name of the CCE container to be deployed.
Image	Select the image to be deployed.
Image Tag	Select the tag of the image to be deployed.

Parameter	Description
Container Specifications	You can configure the specifications of the target container in the target workload. • **CPU Quota** – **Request**: Minimum number of CPU cores required by a container. Resources are scheduled for the container based on this value. The container can be scheduled to a node only when the total available CPU on the node is greater than or equal to the requested quota. – **Limit**: Maximum number of CPU cores required by a container. If the CPU usage is greater than the limit, the CPU resources used by the container may be limited. • **Memory Quota** – **Request**: Minimum memory size required by a container. Resources are scheduled for the container based on this value. The container can be scheduled to a node only when the total available memory on the node is greater than or equal to the requested quota. – **Limit**: Maximum memory size available for a container. When the memory usage exceeds the configured memory limit, the instance may be restarted, affecting normal use of deployment.
Environment Variables	You can configure environment variables of the target container in the target workload. You can synchronize real-time environment variables from CCE to this page to replace current variables.
Control Options	Continue the task even if this action fails.
Overtime	Maximum execution duration of an action, in minutes. If the task duration exceeds the specified time before you stop the application, the action will be timed out. Value range: 1–30

NOTE

You can use ${XXX} to reference parameters **Container**, **Image**, **Image Tag**, **Instances**, and (including **APM Environment** and **APM Sub-Businesses**). For operation details, see **Parameter Management**.

3.7.23 Deploying an Application with a Custom Kubernetes Cluster

Deploy an application in a Kubernetes cluster with a manifest file defining Kubernetes objects.

Prerequisites

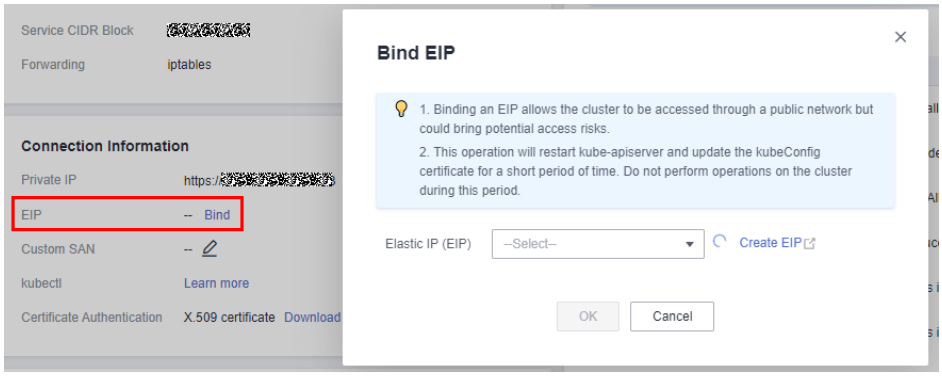
A custom cluster is available.

Procedure

Step 1 Obtain the kubeconfig file.

- **Your Kubernetes cluster is used as an example.**
- **Example of a CCE cluster**

  a. Go to the console. In the upper left corner of the page, choose **Service List** > > **Cloud Container Engine**. Click the target cluster and click **Bind** next to **EIP** to bind the public IP address.



  NOTE

  The CodeArts Deploy official resource pool and your Kubernetes cluster are not in the same VPC. Therefore, you can access the Kubernetes cluster only through an EIP.

  b. Click **Configure** next to **kubectl** in the **Connection Information** area. On the displayed page, click **Download** under **Download the kubeconfig file** to download the configuration file.

  NOTE

  After the download is complete, a **kubeconfig.json** file is available.

Step 2 Create a Kubernetes endpoint.

1. Log in to CodeArts Deploy.
2. Click **Create Application**, enter basic information, click **Next**, select **Blank Template**, and click **OK**. The **Deployment Actions** tab page is displayed.
3. Click **All**, search for **Deploy a Custom Kubernetes Cluster**, and click **Add**.
4. Create an endpoint for accessing the Kubernetes cluster.

   Click **Create** to create a Kubernetes access point.

   After entering the information, click **Verify and OK** to check whether the endpoint is configured successfully.

Table 3-25 Parameters

Parameter	Description
Service Endpoint Name	Name of the service endpoint.
Kubernetes URL	Set this parameter to the public API Server address in the custom cluster.
Kubeconfig	Copy all content in the **kubeconfig.json** file.

5. Configure other required parameters as prompted to complete deployment over the public network with Kubernetes.

Table 3-26 Parameters

Parameter	Description
Region	Select the region for deployment.
Kubernetes Service Endpoint	Select the target Kubernetes access point. You can create and manage Kubernetes access points.
kubectl Command	Select the target kubectl command.
Use manifest file	If this option is selected, you need to select the target manifest file for deployment. The file name must be suffixed with .yaml, .yml, or .json.
kubectl Parameters	kubectl command parameters to be executed.
Control Options	Continue the task even if this action fails.

----End

3.7.24 Kubernetes Nginx-Ingress Grayscale Deployment (CCE Cluster)

Grayscale deployment of CCE Kubernetes clusters based on the ingress-nginx component

Prerequisites

A **CCE cluster** is available.

NOTE

Currently, cluster versions from 1.19 to 1.25 are supported.

Procedure

Table 3-27 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Tenant	• **Current**: The software package is deployed in the CCE cluster of the current tenant for release. Select **Current**. The current tenant must have the CCE cluster operation permission. If the current tenant does not have the CCE cluster operation permission, select **Authorized User** for deployment. • **Other**: The software package is deployed and published in the CCE cluster of another tenant in IAM authorization mode. If you select **Other**, you must select an authorized tenant to deploy the CCE cluster. **NOTE** You are advised to configure the AK/SK of a member account that has the **CCE cluster operation permission** and not advised to configure the AK/SK of a tenant account.
Authorized User	If you do not have the permission to execute an API, this parameter enables you to obtain the temporary AK/SK of the parent user to execute the CCE API through IAM.
Region	Select the region for deployment.
Cluster Name	Select the Kubernetes cluster applied on CCE.
Namespace	Select the namespace of the Kubernetes cluster on CCE.
Workload	Select the target Deployment.
Service	Name of the service bound to the target workload.
Ingress	Select the name of the route bound to the target service.
Container	Select the name of the CCE container to be deployed.
Image	Select the image to be deployed.
Image Tag	Select the tag of the image to be deployed.

Parameter	Description
Enabling grayscale configuration	**Grayscale release policy**: • **Header** **Header-Key**: You can enter the key of a custom header. **Header-Value**: You can enter a custom header value. The value can be a character string or a regular expression. The regular expression format is ^....$. Grayscale traffic weight (%): Traffic can be customized. • **Cookie** Cookie: Custom cookie content can be entered. Grayscale traffic weight (%): Traffic can be customized. **NOTE** The values of **Header** and **Cookie** can contain a maximum of 500 characters.

3.7.25 Deploying with Helm3

Helm is a Kubernetes package management tool, which is similar to the package manager in Linux, such as yum and APT. Helm can easily deploy packaged YAML files on Kubernetes. Helm 3 is the commonly used and stable version of Helm.

CodeArts Deploy allows you to use Helm to deploy and upgrade Kubernetes clusters.

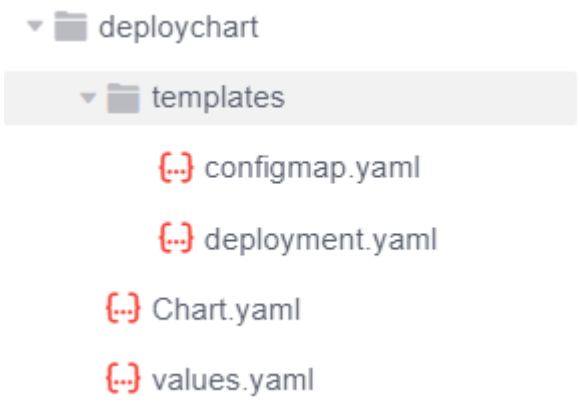
Table 3-28 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Cluster Type	The default value is **Custom**.
Kubernetes Service Endpoint	You can select a CCE cluster or your own Kubernetes cluster. • Huawei Cloud CCE clusters 1. Create a CCE cluster. 2. Create a namespace. 3. Select the CCE cluster to be deployed. • Your own Kubernetes cluster **Configure the kubeconfig file** and select the cluster to be deployed.
Helm Command	**install**, **upgrade**, and **uninstall** are available. If you select **upgrade** for **Helm Command** and the chart name does not exist, the system automatically run **install**.
Namespace	Enter a namespace.

Parameter	Description
Chart Name	The custom chart name. You can perform the upgrade operation on the same chart name.
Chart Package Source	Select the source of the chart package to be installed. **Artifact** and **Repo** are available. If you select **Repo**, you need to specify the code repository and branch.
Chart Package	Enter a directory or GZIP package with a chart file structure.
Values File	Select the **values** file from Artifact. For example, if you specify **Myvalues.yaml**, **-f Myvalues.yaml** will be added to Helm command parameters.
Values	Set values in the CLI. If you specify **key1=val1,key2=val2** (separate values with commas), **-set key1=val1,key2=val2** will be added to Helm command parameters.
Helm Command Parameters	Add other content to Helm command parameters. For details, see **Helm Install**, **Helm Upgrade**, and **Helm Uninstall**.
Control Options	Continue the task even if this action fails.

Environment Preparation for Helm3 Deployment Example

This section uses the chart directory as an example to describe how to prepare the environment for the following three examples. Use the following template to deploy a CCE cluster and create the following directories in the code repository of CodeArts Repo.



Segment in **configmap.yaml**

```
metadata:
  name: {{ .Values.configmapname }}
```

Segment in **deployment.yaml**


```
spec:
  template:
    spec:
      containers:
        - image: '{{ .Values.imagename }}:{{ .Values.imagetag }}'
```

Segment in **values.yaml**

```
configmapname: valuesfromfile
imagename: httpd
imagetag: latest
```

NOTE

{{**.Values.***xxxx*}} corresponds to the variable defined in the **values.yaml** file in the chart.

The following three examples are based on this section.

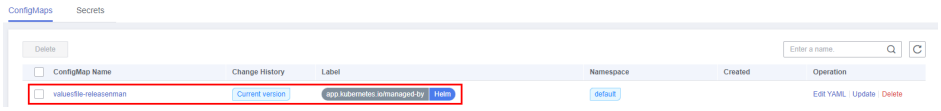
1. **Example 1: Using the Chart Package or Chart File Structure Directory for Deployment**

   If the default **values** file exists in the chart, you do not need to specify the **values** file in Artifact. You can directly deploy the default **values** file.

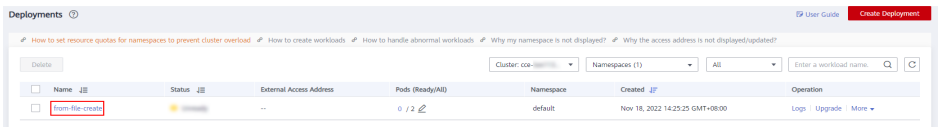
   The deployment result is as follows:

   ```
   TASK [excute logs] ********************************************************************
   ok: [localhost] => {
       "msg": [
           "Release \"myapp\" does not exist. Installing it now.",
           "NAME: myapp",
           "LAST DEPLOYED: Fri Sep 24 06:57:09 2021",
           "NAMESPACE: default",
           "STATUS: deployed",
           "REVISION: 1",
           "TEST SUITE: None"
       ]
   }
   ```

   The corresponding ConfigMap generated on the CCE console is as follows:

   The corresponding Deployment generated on the CCE console is as follows:

2. **Example 2: Deploying Helm3 by Specifying the Values File in CodeArts Artifact**

   This example demonstrates how to deploy Helm3 by specifying the **Values** file in Artifact.

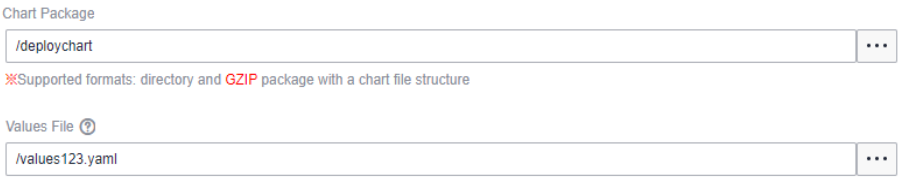
   NOTE

   The values defined in the external **values** file will overwrite the values defined in the **Values** file in the chart.

   Segment of an external **Values** file. In this example, the file is named **values123.yaml**.

   ```
   configmapname: valuesfile-releasenman
   imagename: nginx
   imagetag: stable
   ```

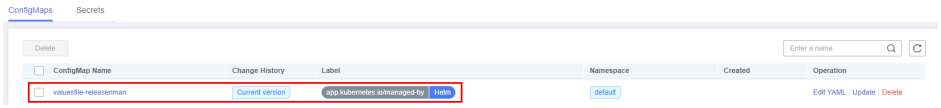

As shown in the following figure, for **Values File**, select the **Values** file in Artifact.



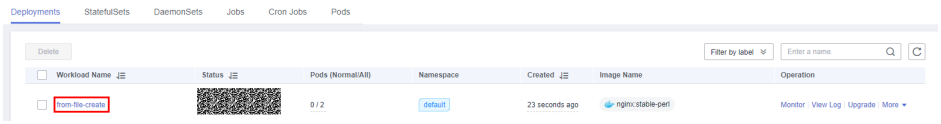
The deployment result is as follows:

```
TASK [excute logs] ************************************************************
ok: [localhost] => {
    "msg": [
        "Release \"myapp\" has been upgraded. Happy Helming!",
        "NAME: myapp",
        "LAST DEPLOYED: Fri Sep 24 07:21:15 2021",
        "NAMESPACE: default",
        "STATUS: deployed",
        "REVISION: 3",
        "TEST SUITE: None"
    ]
}
```

The corresponding ConfigMap generated on the CCE console is as follows:



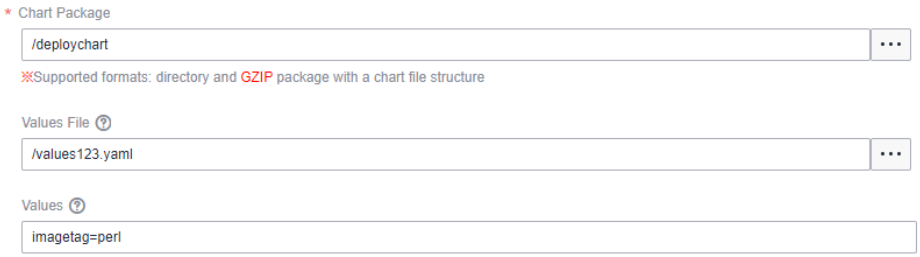
The corresponding Deployment generated on the CCE console is as follows:



3. Example 3: Deploying Helm3 by Configuring Values

If **Values** is set, it has the highest priority and overwrites the values set in the **values** file of Chart and the values set in the external **values** file.

The following figure describes how to configure the image version:



Segment of the **Values File** in **Chart Package**:

```
imagetag: latest
```

Segment of **Values File** in Artifact:

```
imagetag: stable
```

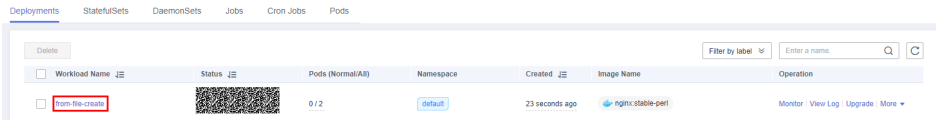
When setting **Values**, enter the following information:

```
imagetag=perl
```


The deployment result is as follows:

```
TASK [excute logs] ********************************************************
ok: [localhost] => {
    "msg": [
        "Release \"myapp\" has been upgraded. Happy Helming!",
        "NAME: myapp",
        "LAST DEPLOYED: Fri Sep 24 07:41:39 2021",
        "NAMESPACE: default",
        "STATUS: deployed",
        "REVISION: 10",
        "TEST SUITE: None"
    ]
}
```

The corresponding Deployment generated on the CCE console is as follows:



3.7.26 Ansible

This action aims to execute the uploaded playbook on the host. Here is the configuration page.

Table 3-29 Parameters

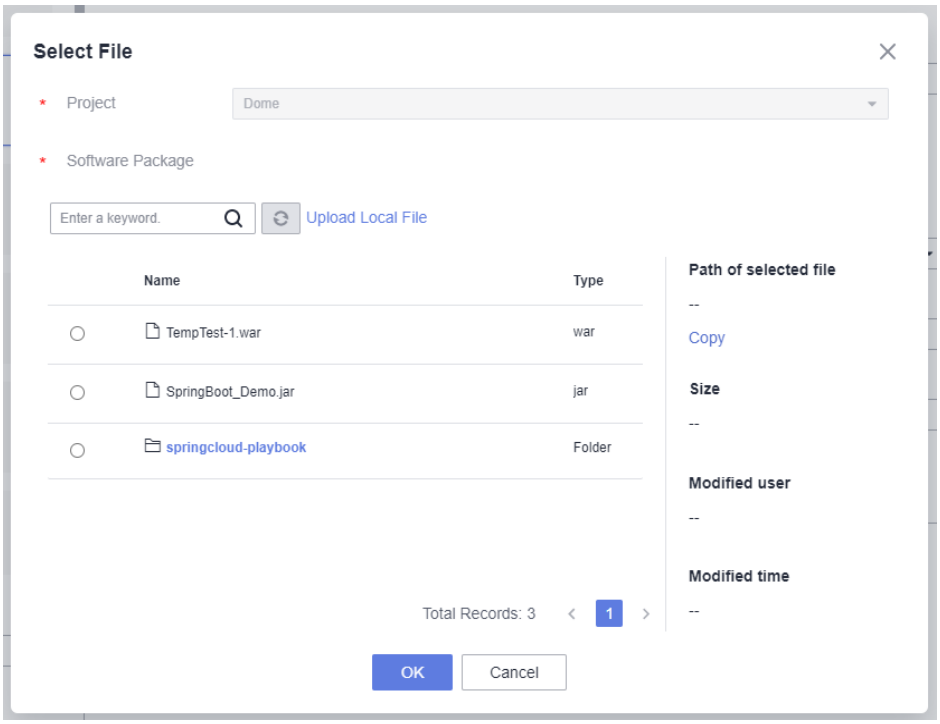
Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Playbook Source	**Artifact** and **Repo** are available.
Playbook File	You can select an existing playbook file from Artifact or a playbook file uploaded from a local host. **NOTE** Local software packages or files uploaded to Artifact can be reused.
Entry File Path	The entry file path of playbook.
Control Options	• Enable this action. • Continue the task even if this action fails.

The following describes how to use playbook based on the playbook source:

CodeArts Artifact

Step 1 On the tab displaying application action details, select **Artifact** for **Playbook Source**, and click on the right of **Playbook File**.

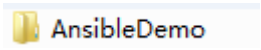
Step 2 In the displayed **Select File** dialog box, select the corresponding playbook file compression package.



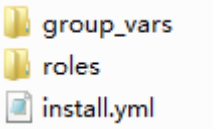
Step 3 Enter the entry file path of playbook.

Step 4 The entry file path is the directory generated after the playbook package is decompressed.

- If the directory after decompression is similar to that shown in the following figure, the entry file path is **AnsibleDemo/install.yml**.

- If the directory after decompression is similar to that shown in the following figure, the entry file path is **install.yml**.

----End

CodeArts Repo

Step 1 On the tab displaying application action details, select **Repo** for **Playbook Source**.

Step 2 Select the code repository address (that is, the SSH URL of the code repository for storing playbook) from the **Repo** drop-down list.

Step 3 Select a code repository before selecting a branch.

Step 4 Select the entry file path.

NOTE

The entry file path is generated after the playbook package is decompressed.

Step 5 Configure parameters.

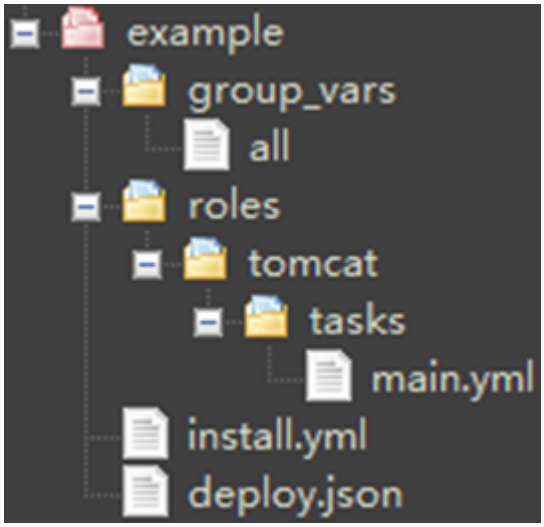
Switch to the **Parameter** tab and click **Create Now** to add parameters for the Ansible application action or replace parameters in the **all** file.

After an application task is executed, the corresponding build task is downloaded to a specified path.

The following is an example of the architecture and content of the ALL file:

```
tomcat_url: http://mirror.olnevhost.net/pub/apache/tomcat/tomcat-7/v7.0.78/bin/apache-tomcat-7.0.78.tar.gz
war_url: http://test.com/xxx.war
```

Figure 3-9 File architecture



NOTE

- If an added parameter exists in the playbook **all** file, the parameter with the same name in the **all** file will be replaced. Otherwise, the parameter will be used as a new parameter.
- The parameter name cannot contain the following characters: decimal points (.), hyphens (-), and colons (:).

----End

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.27 Creating IIS Site

What Is IIS?

Internet Information Services (IIS) is a basic Internet service provided by Microsoft based on Microsoft Windows.

It is a service and a component of the Windows 2000 Server series. Different from common applications, IIS is a part of the operating system like a driver. It is started when the system is started.

Creating an IIS Site on a Windows Host

Table 3-30 Parameters

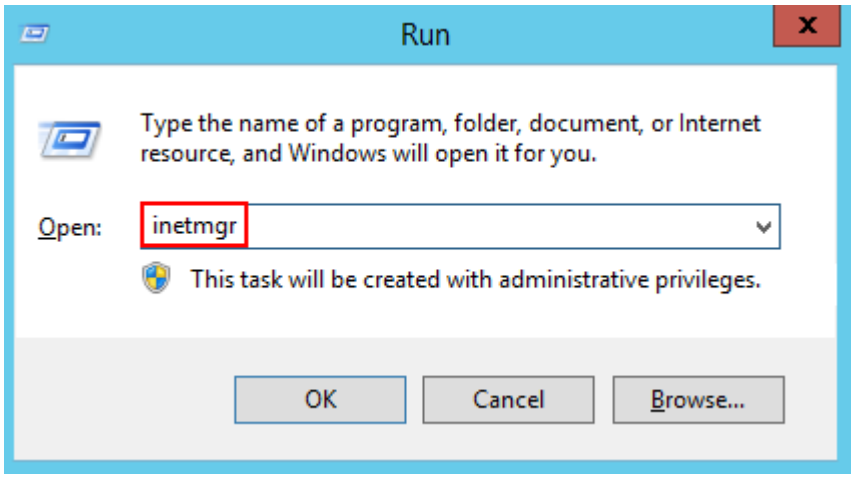
Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Application Pool	Application pool of IIS.
.Net CLR Version	Version of .Net CLR.
Website Name	Name of the website.
Port	The (listening) port that is bound.
Physical Path	Physical path of the application.
Log Path	Log path of the IIS running site.
Control Options	• Enable this action. • Continue the task even if this action fails.

Procedure

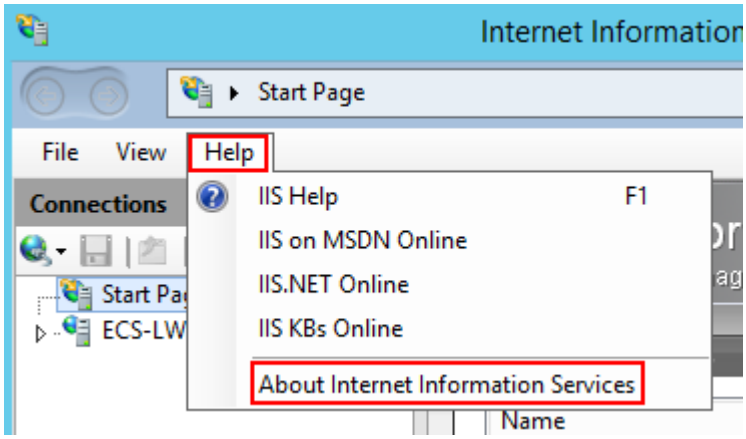
Step 1 Select an environment for the application.

Step 2 Check whether the IIS version of the Windows host where the application is to be performed is later than **7.0**. The procedure is as follows:

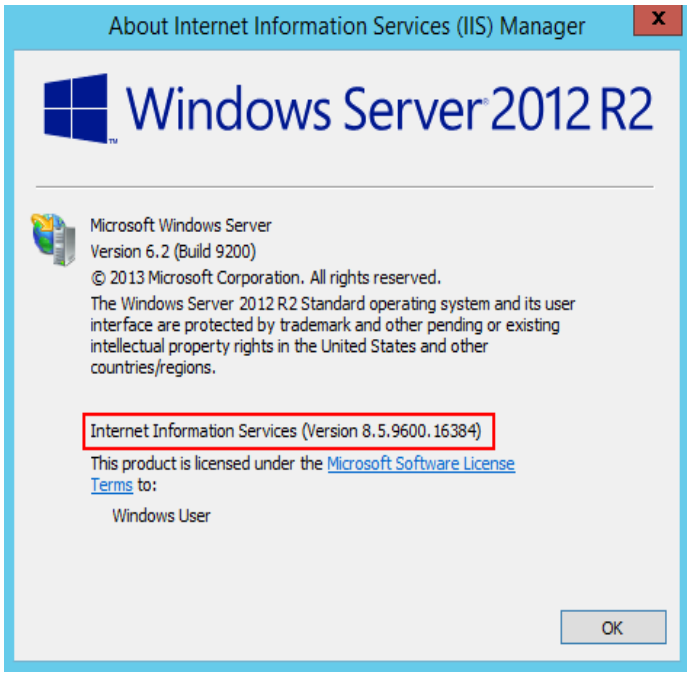
a. Press **Windows+R** to open the **Run** dialog box, enter the **inetmgr** command for opening the IIS management window, and click **OK**.



b. Start Internet Information Services (IIS) Manager, and then choose **Help** > **About Internet Information Services**.



c. View the IIS version in the displayed dialog box.



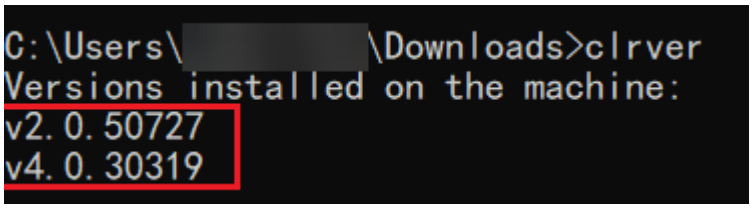
NOTE

If the IIS version of the Windows host where the application is to be performed is earlier than or is **7.0**, upgrade the IIS.

Step 3 Enter the physical path and log path of the application.

Step 4 Enter the .Net CLR Version of the target Windows host.

- Find **clever.exe** in **C:\Program Files\Microsoft SDKs\Windows\v7.0A\bin** or **C:\Program Files\Microsoft SDKs\Windows\v8.0A\bin\NETFX 4.0 Tools**.
- Run the program in the **cmd** window to obtain the version supported by the .NET CLR.



NOTE

If the clever.exe program cannot be found, download and install it.

Step 5 Enter the application pool name and website name.

Step 6 Specify a port (the port bound) and deploy the application.

----End

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.28 Installing IIS

This action aims to install Internet Information Services (IIS) on environments. The following table shows the configuration.

Table 3-31 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Control Options	• Enable this action. • Continue the task even if this action fails.

NOTE

Installing IIS service does not support Windows 7 and Windows 10.

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.29 Installing JDK

This action aims to install JDK on a host. The following figure shows the configuration page.

Table 3-32 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
JDK Version	JDK version.
Installation Path	Installation path of JDK.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

When purchasing a Huawei Cloud ECS, you are advised to select CentOS, Ubuntu, or Huawei Cloud EulerOS based on Arm. EulerOS based on Arm does not have the yum source of openjdk-11.

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.30 Installing Tomcat

This action aims to install Tomcat on a host. The following figure shows the configuration page.

Table 3-33 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Tomcat Version	Tomcat version.
Installation Path	Installation path of Tomcat.
HTTP Port	Default port: **8080**
AJP Port	Default port: **8009**
Service Shutdown Port	Default port: **8005**

Parameter	Description
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.31 Installing Nginx

This action aims to install Nginx on a host. The following figure shows the configuration page.

Table 3-34 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Nginx Version	Nginx version.
Installation Path	Installation path of Nginx.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.32 Installing Go

This action aims to install Go on a host. The following figure shows the configuration page.

Table 3-35 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Go Version	GO version.
Installation Path	Installation path of Go.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.33 Installing PHP

This action aims to install PHP on a host. The following figure shows the configuration page.

Table 3-36 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
PHP Version	PHP version.
Installation Path	Installation path of PHP.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.34 Installing Node.js

This action aims to install Node.js on a host. The following figure shows the configuration page.

Table 3-37 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Node.js Version	Node.js version.
Installation Path	Installation path of Node.js.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.35 Installing Python

This action aims to install Python on a host. The following table shows the configuration page.

Table 3-38 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Python Version	Python version.
Installation Path	Installation path of Python.

Parameter	Description
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

NOTE

If you encounter any problem during deployment, see **Solutions to Common Problems**.

3.7.36 Installing/Uninstalling Docker

Install or uninstall the Docker environment on hosts.

Table 3-39 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Environment	Target environment.
Operation	Select **Install Docker** or **Uninstall Docker**. **NOTE** The Docker service supports only users with the sudo permission. This installation will overwrite the previous Docker version.
Docker Version	Target version of Docker to be installed.
Control Options	• Enable this action. • Continue the task even if this action fails. • Execute this action with the sudo permission.

3.7.37 Istio Gray Release

Istio provides you with microservice-based traffic governance capabilities. Istio allows you to develop a set of traffic distribution policies based on standards and deliver the policies to application pods in a non-intrusive manner, implementing smooth and stable grayscale release.

Prerequisites

1. A CCE cluster is available. If no CCE cluster is available, **create one**.
2. A workload of the current version exists and a Service has been created. If no workload exists, create a **workload**.

3. The ASM service has been enabled, and **Configuration Diagnosis Result** of the corresponding service on the **Service Management** tab page is in the normal state. If the ASM service is not enabled, enable the ASM service.
4. An Istio workload has been created and associated with the Service of the current version.

Procedure

Table 3-40 Parameters of custom release mode

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Cluster Name	Select a target cluster.
Namespace	Enter a namespace.
Release Mode	**Custom** and **Fast** are supported.
File Source	• **Artifact** **YML File**: Select the target YML file. • **Repo** **Repo**: Select the target code repository. **Branch**: Select the target branch. **YML File Path**: path of the target YML file.
Control Options	Continue the task even if this action fails.

Table 3-41 Parameters of fast release mode

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Cluster Name	Select a target cluster.
Namespace	Enter a namespace.
Release Mode	**Custom** and **Fast** are supported.

Parameter	Description
Traffic Takeover	**Gray release** **VirtualService Name**: Select the target VirtualService. Log in to the ASM console, choose **Mesh Configuration** > **Istio Resource Management** and filter the target namespace and istio resources. **DestinationRule Name**: Select the target destination rule. Log in to the ASM console, choose **Mesh Configuration** > **Istio Resource Management** and filter the target namespace and istio resources. **Current Version**: Use the keyword "version" in the label to distinguish the official version from the gray version. The version number must be the same as the subsets object name in DestinationRule and is used as an identifier for gray traffic distribution. **Gray Version Number**: Use the keyword "version" in the label to distinguish the official version from the gray version. The version number must be the same as the subsets object name in DestinationRule and is used as an identifier for gray traffic distribution. **Gray release policy**: • **Based on traffic ratio** **Gray Version Traffic (%)**: Traffic can be customized. • **Based on request content-Cookie** **Cookie Content**: Cookie content can be customized. • **Based on request content-Header** **Custom Header**: Headers can be added and customized.
	Official release **VirtualService Name**: Select the target VirtualService. Log in to the ASM console, choose **Mesh Configuration** > **Istio Resource Management** and filter the target namespace and istio resources. **DestinationRule Name**: Select the target destination rule. Log in to the ASM console, choose **Mesh Configuration** > **Istio Resource Management** and filter the target namespace and istio resources. **Official Version**: Enter the version that officially takes over traffic.
Control Options	Continue the task even if this action fails.

3.7.38 Deploying to FunctionGraph

In this step, you can deploy software packages in the Artifact, Repo, and OBS to FunctionGraph to release a new version.

Prerequisites

You have **FunctionGraph operation permissions**.

Procedure

Table 3-42 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Deployment Tenant	• **Current tenant**: The software package is deployed in the FunctionGraph extension of the current tenant for release. Select **Current tenant**. The current tenant must have the FunctionGraph operation permission. If the current tenant does not have the FunctionGraph operation permission, select **IAM authorization** for deployment. • **Other tenant**: The software package is deployed and published in the FunctionGraph of another tenant in IAM authorization mode. You must select an authorized tenant for FunctionGraph deployment. **NOTE** You are advised to configure the AK/SK of a member account with **FunctionGraph operation permissions** and not advised to configure the AK/SK of a tenant account.
IAM authorization	If the current user does not have the FunctionGraph operation permissions, you can use IAM to authorize the user.
Function	For details about the function created in FunctionGraph, see **Quickly Creating a FunctionGraph Function**.
Deployment source	The deployment source can be Artifact, Repo, or OBS. • **Artifact**: You can select a software package from Artifact. The software package can be in ZIP or JAR format and must meet FunctionGraph requirements. For details, see **how to develop a function**. • **Repo**: You can manage code repos after choosing **Code** > **Repo**. • **OBS**: You can directly enter the address of the software package uploaded to OBS.
Release New Version	New versions of FunctionGraph can be released. A function can have a maximum of 10 version numbers and each version number must be unique.
Control Options	• Enable this action. • Continue the task even if this action fails.

NOTICE

If **Deployment source** is set to **Artifact** or **Repo**, the maximum size of a code package is 50 MB. If the size of a code package exceeds 50 MB, you are advised to deploy the code package using OBS.

3.7.39 FunctionGraph Grayscale Release

This action supports version switching and grayscale release based on the function alias mechanism of FunctionGraph.

Prerequisites

You have **FunctionGraph operation permissions**.

Procedure

Table 3-43 Parameters

Parameter	Description
Action Name	Name of an action displayed in the deployment actions area.
Deployment Tenant	• **Current tenant**: The software package is deployed in the FunctionGraph extension of the current tenant for release. Select **Current tenant**. The current tenant must have the FunctionGraph operation permission. If the current tenant does not have the FunctionGraph operation permission, select **IAM authorization** for deployment. • **Other tenant**: The software package is deployed and published in the FunctionGraph of another tenant in IAM authorization mode. If you select **Other tenant**, you must select an authorized tenant for FunctionGraph deployment. **NOTE** You are advised to configure the AK/SK of a member account with **FunctionGraph operation permissions** and not advised to configure the AK/SK of a tenant account.
IAM authorization	If the current user does not have the FunctionGraph operation permissions, you can use IAM to authorize the user.
Function Name	For details about the function created in FunctionGraph, see **Quickly Creating a FunctionGraph Function**.
Function Alias	Alias of a function. A function alias can be bound to two versions (including one grayscale release version). In addition, you can configure traffic distribution weights for two versions under the same alias. Only one alias can be created for each version.
Version	You can use an alias to call a function of the corresponding version. Only one alias can be created for each version.

Parameter	Description
Turn on grayscale version	The grayscale release version can distribute the traffic of the main version. You can control how much traffic to distribute with a weight. For more information, see **Managing Versions**.
Control Options	• Enable this action. • Continue the task even if this action fails.

4 Managing Orchestration Templates

4.1 Background

CodeArts Deploy provides developers with common deployment process templates, that is, **system templates**. You can use system templates to quickly create applications. If system templates cannot meet your requirements, you can add, modify, or delete deployment actions based on system templates. For details, see **Editing a Deployment Action** or create **custom templates**.

NOTE

You can add a system template to favorites. After the template is added to favorites, the template is moved to the top of the system template list. If multiple templates are added to favorites, latest templates are displayed on top.

4.2 System Templates

4.2.1 Deploying to FunctionGraph

This action deploys a software package to FunctionGraph and releases a new version.

The related deployment actions are as follows:

NOTE

For details, see **Deploying to FunctionGraph**.

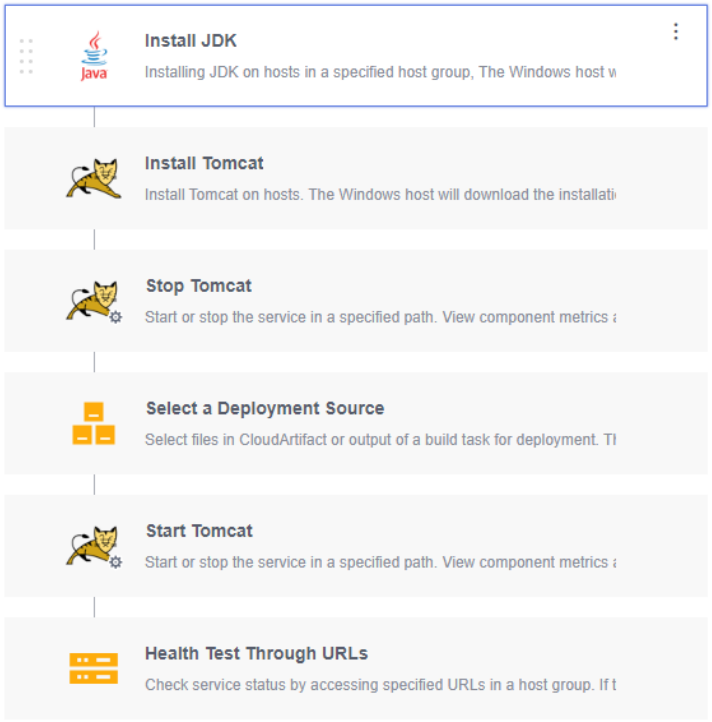
4.2.2 Deploying a Tomcat Application

Deploy a WAR package to the Tomcat service on a host and start the service.

NOTE

Ensure that the **Tomcat** has been installed on the target host. If the **Tomcat** has been installed, remove the **Install Tomcat** action from the template.

The related deployment action is as follows.



Step 1 **Install JDK**.

Step 2 **Install Tomcat**.

Step 3 **Stop Tomcat**.

Step 4 **Select a deployment source**.

Step 5 **Start Tomcat**.

Step 6 **Perform health test through URLs**.

----End

Table 4-1 describes the parameters in the template.

Table 4-1 Template parameters

Parameter	Description
host_group	Target environment where the application is deployed.
package_url	Software package download link. To obtain it, go to the **Artifact** > **Release Repos** page.
service_port	Port number of a Tomcat application. The default value is **8080**.

Parameter	Description
tomcat_home_path	Path of the Tomcat client. The default value is **/usr/local/tomcat/apache-tomcat-8.5.38**.

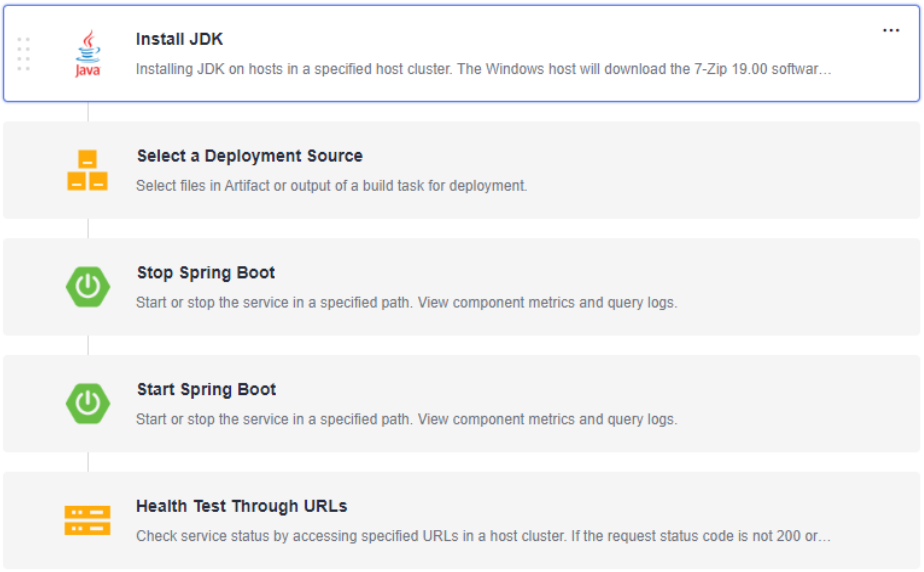
4.2.3 Deploying a Spring Boot Application

Deploy a Spring Boot Java background application on the host and start the service.

NOTE

Ensure that the **JDK** has been installed on the target host. If the **JDK** has been installed, remove the **Install JDK** action from the template.

The related deployment action is as follows.



Step 1 **Install JDK**.

Step 2 **Select a deployment source**.

Step 3 **Stop Spring Boot**.

Step 4 **Start Spring Boot**.

Step 5 **Perform health test through URLs**.

----End

Table 4-2 describes the parameters to be set in the template.

Table 4-2 Template parameters

Parameter	Description
host_group	Target environment where the application is deployed.

Parameter	Description
package_url	Software package download link. To obtain it, go to the **Artifact** > **Release Repos** page.
service_port	Port number of a Spring Boot application. The default value is **8080**.
package_name	Name of the Spring Boot application release package. **NOTE** The name does not contain the file name extension.

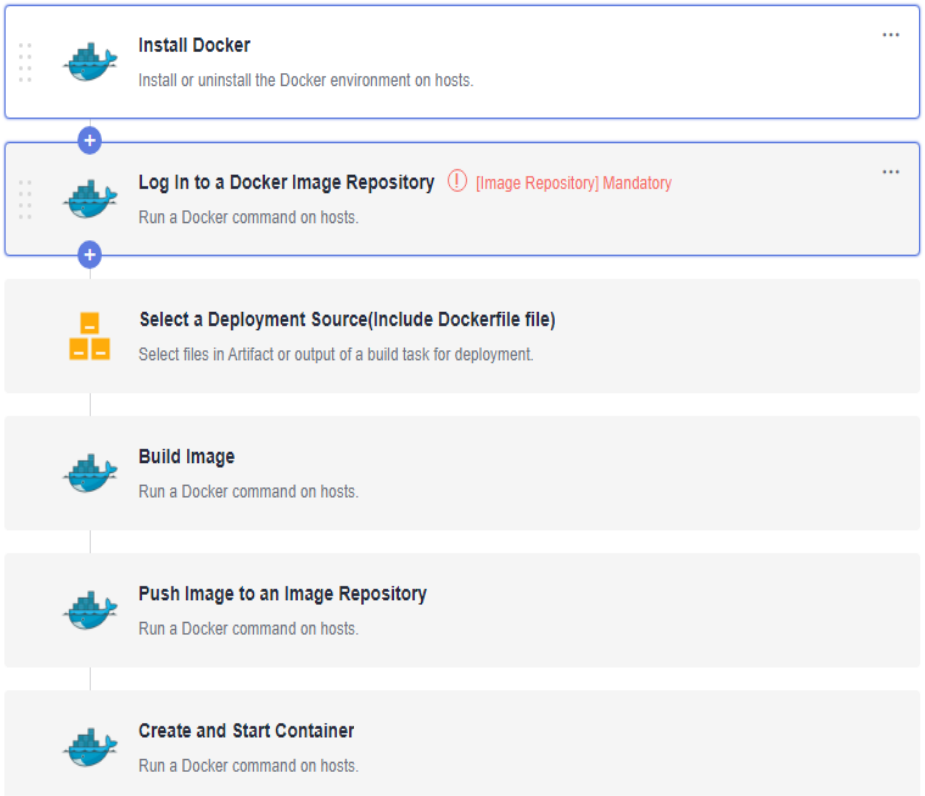
4.2.4 Deploying a Docker Application (Linux)

Install Docker on the host, log in to the remote repository, download files such as Dockerfiles, and perform operations such as build, push, and run.

NOTE

Docker has been installed on the target host. Remove the **Install Docker** action from the template.

The related deployment action is as follows:



Step 1 **Install Docker**.

Step 2 **Log in to the Docker image repository**.

Step 3 **Select a deployment source**.

Step 4 **Build an image**.

Step 5 **Tag the image**.

Step 6 **Upload the image to the image repository**.

Step 7 **Create and run a container**.

----End

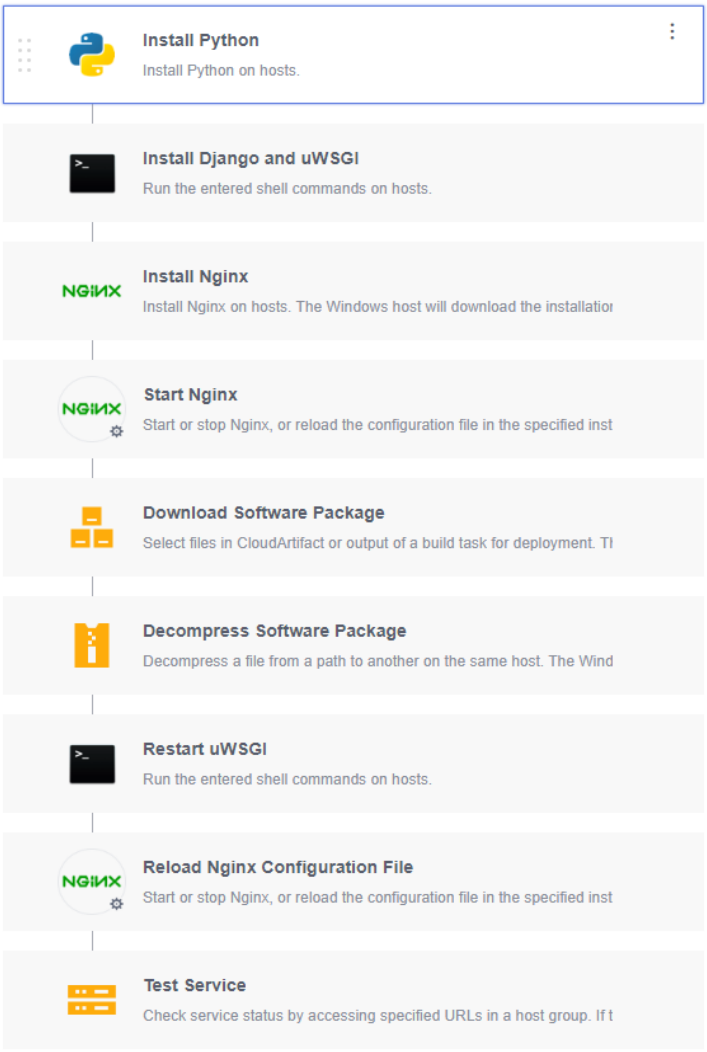
4.2.5 Deploying a Django Application

Deploy a Django Python background application on the host and start the service.

NOTE

- Ensure that the **Nginx** has been installed on the target host. If the **Nginx** has been installed, remove the **Install Nginx** action from the template.
- Ensure that the **uWSGI** has been installed on the target host. If the **uWSGI** has been installed, remove the **Install uWSGI** action from the template.
- You have configured the pip and yum sources. yum and pip are used to install software. The corresponding source addresses are configured to ensure normal installation.
- You have created a Django project and uploaded the project files to Artifact. You can use CodeArts Build to compress the Django project files and upload the package to Artifact. Then, download and decompress the package during the deployment.

The related deployment action is as follows.



Step 1 **Install Python**.

Step 2 **Install Django and uWSGI**.

Step 3 **Install Nginx**.

Step 4 **Start Nginx**.

Step 5 **Download the software package**.

Step 6 **Decompress the software package**.

Step 7 **Restart the uWSGI**

Step 8 **Reload the Nginx configuration file**.

Step 9 **Test services**.

----End

Table 4-3 describes the parameters to be set in the template.

Table 4-3 Template parameters

Parameter	Description
host_group	Target environment where the application is deployed.
service_port	Port number of a Django application. The default value is **8080**.
uwsgi_pid_file_path	Path of the uWSGI process ID file.
uwsgi_lni_file_path	Path of the uWSGI configuration file.
package_path	Path for downloading the Django application release package to the target host.
package_name	Name of the Django application release package downloaded to the target host.
package_url	Software package download link. To obtain it, go to the **Artifact** > **Release Repos** page.

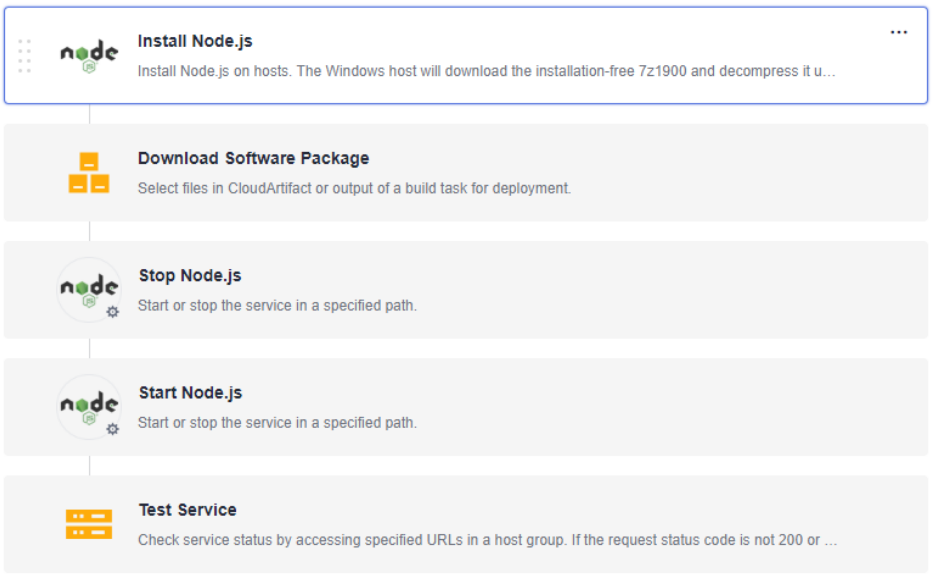
4.2.6 Deploying a Node.js Application

Deploy a Node.js web service on the host and start the service.

NOTE

Node.js has been installed on the target host. Remove the **Install Node.js** action from the template.

The related deployment action is as follows.



Step 1 Install Node.js.

Step 2 **Download the software package**.

Step 3 **Stop Node.js**.

Step 4 **Start Node.js**.

Step 5 **Test services**.

----End

Table 4-4 describes the parameters to be set in the template.

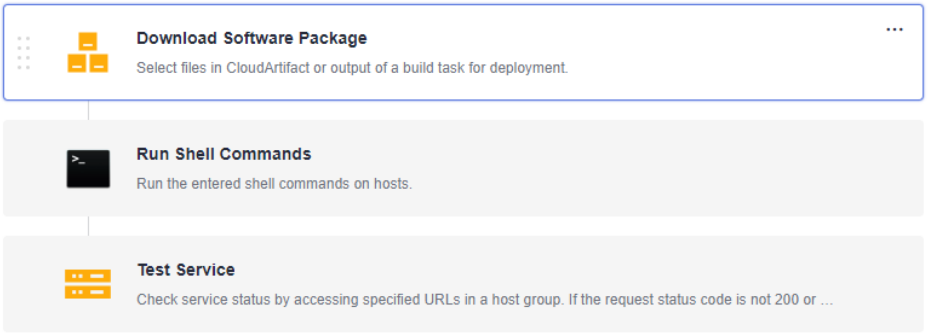
Table 4-4 Template parameters

Parameter	Description
host_group	Target environment where the application is deployed.
package_url	Software package download link. To obtain it, go to the **Artifact** > **Release Repos** page.
service_port	Application port.

4.2.7 Deploying a General Application

Deploy a general application using Shell scripts.

The related deployment action is as follows.



Step 1 **Download the software package**.

Step 2 **Execute the deployment script**.

Step 3 **Perform the health test**.

----End

Table 4-5 describes the parameters to be set in the template.

Table 4-5 Template parameters

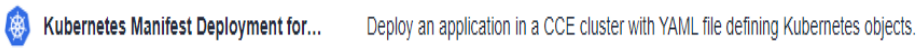
Parameter	Description
host_group	Target environment where the application is deployed.

Parameter	Description
package_url	Software package download link. To obtain it, go to the **Artifact** > **Release Repos** page.
app_name	Application name to obtain the process ID and stop the process.
service_port	Application port.

4.2.8 Deploying an Application in Kubernetes (CCE Cluster) Using Manifest

This section introduces how to deploy an application in a Huawei Cloud CCE cluster with manifest file defining Kubernetes objects.

The related deployment action is as follows.



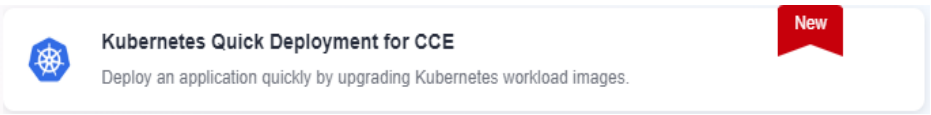
NOTE

For details, see **Deploying an Application in Kubernetes (CCE Cluster) Using Manifest**.

4.2.9 Deploying an Application in Kubernetes (CCE Cluster) Quickly

Deploy an application quickly by upgrading Kubernetes workload images.

The related deployment action is as follows.



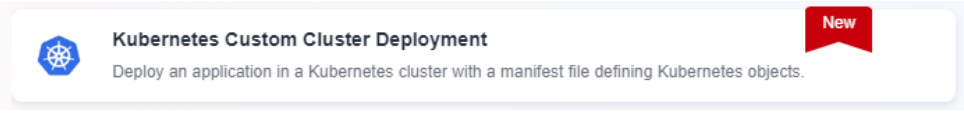
NOTE

For details, see **Deploying an Application in Kubernetes (CCE Cluster) Quickly**.

4.2.10 Deploying an Application with a Custom Kubernetes Cluster

Deploy an application in a Kubernetes cluster with a manifest file defining Kubernetes objects.

The related deployment action is as follows.



NOTE

For details, see **Deploying an Application with a Custom Kubernetes Cluster**.

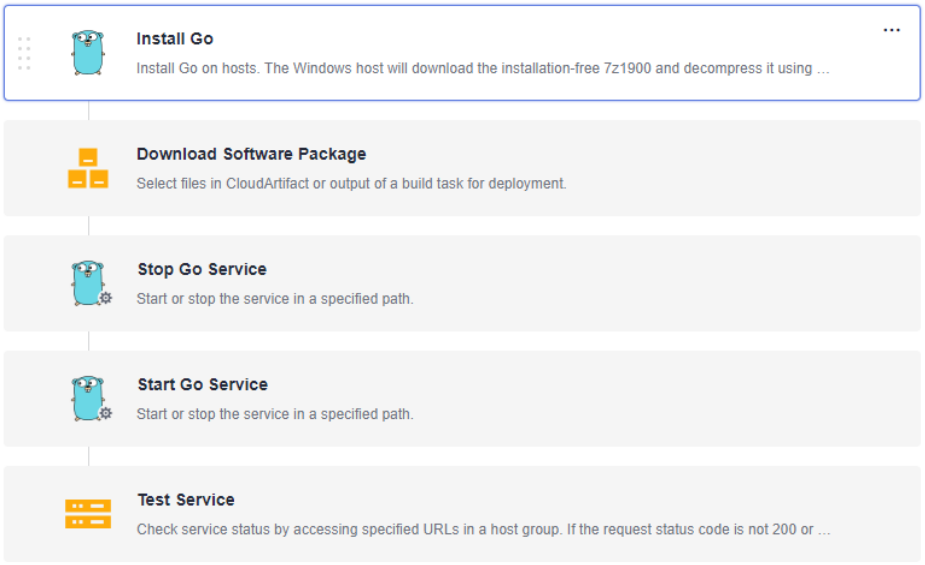
4.2.11 Deploying a Go Application

Deploy a Go web service on the host and start the service.

NOTE

Go has been installed on the target host. Remove the **Install Go** action from the template.

The related deployment action is as follows.



Step 1 **Install Go**.

Step 2 **Download the software package**.

Step 3 **Stop Go**.

Step 4 **Start Go**.

Step 5 **Perform the health test**.

----End

The table below describes the parameters to be set in the template.

Table 4-6 Template parameters

Parameter	Description
host_group	Target environment where the application is deployed.
package_url	Software package download link. To obtain it, go to the **Artifact** > **Release Repos** page.
app_name	Application name to obtain the process ID and stop the process.

Parameter	Description
service_port	Application port.

4.3 Custom Templates

4.3.1 Customizing a Template

A custom template can be used in either of the following scenarios:

- If system templates cannot meet your requirements, you can **use a blank template to create a custom template**.
- **Create a custom template using an existing application**. This custom template can be used by other members in the project to quickly create applications.

A custom template can be directly used during application creation.

Customizing from a Blank Template

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Choose **CICD** > **Deploy**.

Step 3 Click **Create Application**, enter basic information, and click **Next**.

Step 4 Click **Blank Template** to enter the **Deployment Actions** tab page.

Step 5 On the **Deployment Actions** page, add deployment actions based on service requirements.

Step 6 Click **Save as Custom Template**. In the dialog box displayed, enter the template name and description, and click **OK**.

----End

Customizing a Template from an Existing Application

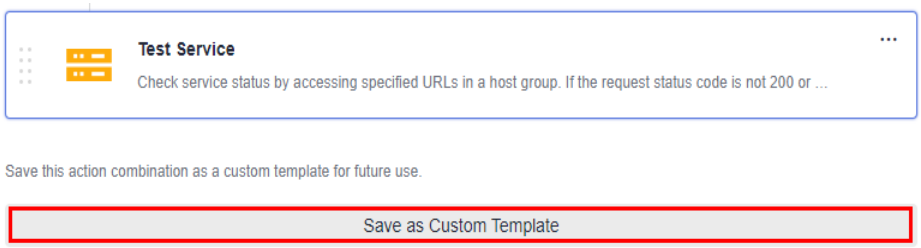
Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Choose **CICD** > **Deploy**.

Step 3 On the **Applications** tab page, select an application to be deployed and find its **Operation** column. Click **More** and choose **Edit** to view the detailed configuration information of the application.

Step 4 Edit the deployment actions based on service requirements.

Step 5 Click **Save as Custom Template**. In the dialog box displayed, enter the template name and description, and click **OK**.



----End

Creating a Custom Template on the Application Creation Page

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Choose **CICD** > **Deploy**.

Step 3 Click **Create Application**. The **Basic Information** page is displayed.

Step 4 Click **Next**, select **Custom Templates** in the **Template** column, and click **New Template** on the right of the custom template. If no template is available, you can click **Create One** to create a custom template.

Step 5 Configure basic information, template orchestration, and parameter settings as needed.

Step 6 Click **Save** in the upper right corner.

----End

Creating Custom Templates on the Orchestration Template Management Page

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

Step 2 Choose **CICD** > **Deploy**.

Step 3 Click **Orchestration Template Management** and click **All**.

Step 4 Click **Create Custom**.

Step 5 Configure basic information, template orchestration, and parameter settings as needed.

Step 6 Click **Save** in the upper right corner.

----End

4.3.2 Favoriting, Editing, or Deleting a Custom Template

This section describes how to add a custom template to favorites and edit or delete a custom template.

Procedure

Step 1 Go to the CodeArts homepage and click the target project name to access the project.

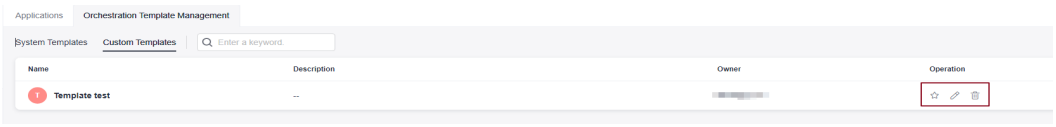
Step 2 Choose **CICD** > **Deploy**.

Step 3 Choose **Orchestration Template Management** > **Custom Templates**.

Click ☆ next to a template to add it to your favorites. Then the template will be pinned on the top of the custom template list.

Click ✎ next to a template and edit the template.

Click 🗑 next to a template to delete the template. Note that the template cannot be restored after being deleted.



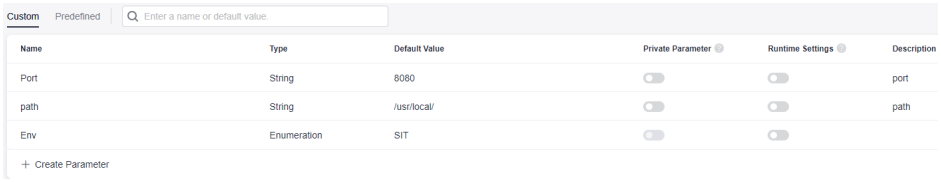
----End

5 Managing Parameters

This section describes how to set application parameters. By setting application parameters, you can deploy applications based on customized parameters.

Application parameters are classified into the following types:

- **Custom**: Add parameters as required. Parameter types include string, enumeration, and environment.
- **Predefined**: The parameter values are generated and cannot be deleted or modified.

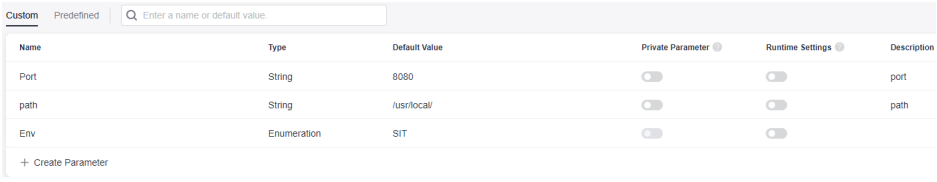


Configuring Parameters

This section describes how to create and configure user-defined parameters in an application.

Step 1 Select the target application, click ***, and click **Edit**. The **Deployment Actions** page is displayed.

Step 2 Click the **Parameters** tab.



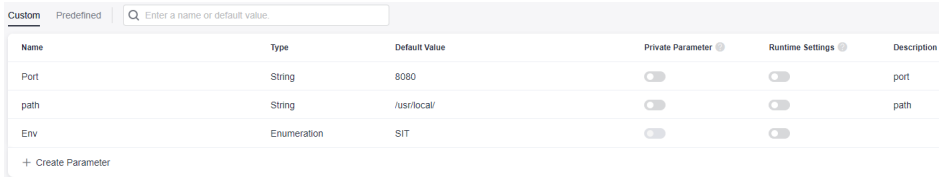
The following parameters are provided.

Basic Information	Description
Create Parameter	You can click **Create Parameter** to add a parameter.
Name	Parameter name. You can change the name of a custom parameter. **NOTE** The name of a custom parameter cannot be the same as that of a predefined parameter.
Type	Parameter types include string, enumeration, and environment.
Default Value	Enter or select a parameter value. **NOTE** If no environment is available when you select an environment type, you need to manually create an environment.
Private Parameter	If a parameter is private, the system encrypts the input for storage and only decrypts the parameter when using it. **Runtime Settings** cannot be set.
Runtime Settings	If this parameter is enabled, the parameter value can be changed during application deployment and the parameter value will be reported to the application. Runtime parameters need to be entered during execution.
Description	Parameter description.
Operation	Click  to delete a parameter.

Step 3 Click **Create Parameter** to add a parameter. Customize the parameter name, type (**String** by default), and default value as required, and set private parameters and runtime settings.

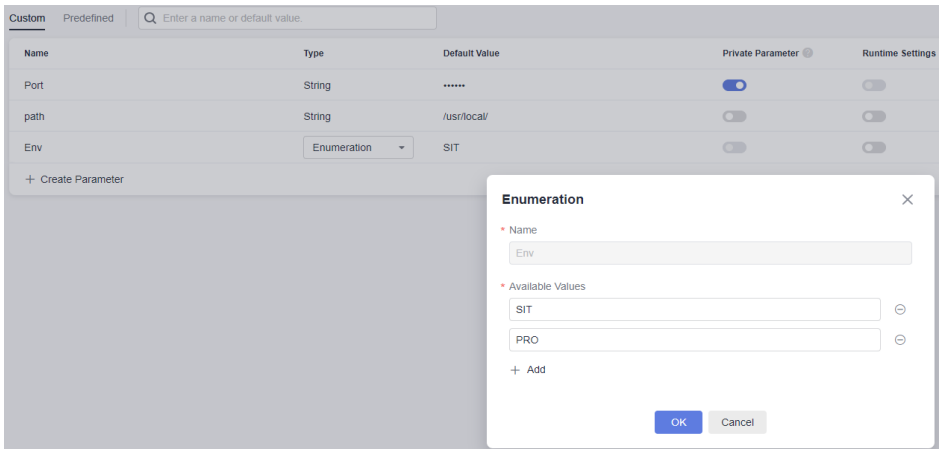
- **String**

  The parameter value is a character string. You can customize the value in the **Default Value** column, and enable the private parameter or runtime settings.

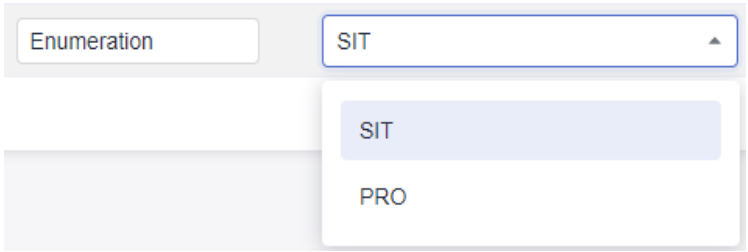


- **Enumeration**

  After you set the type to **Enumeration**, the enumeration dialog box is displayed for you to set **Available Values**. To set multiple values, click the + Add icon. To delete a value, click ⊖.



After the setting, select a value from the **Default Value** drop-down list, as shown in the following figure.



- **Environment**

  Select an environment from the **Default Value** drop-down list. You can select an environment created in the application from the drop-down list.

  NOTE

  If there is no option in the drop-down list, create an environment on the **Environment Management** tab page. Then return to the parameter configuration page and click to refresh the environment to the drop-down list.

Step 4 After modifying all information, click **Save**.

----End

Using a Parameter

This section uses an example to describe how to use custom parameters.

Step 1 Select the target application, click , and click **Edit**. The **Deployment Actions** page is displayed.

Step 2 Click **Parameters**, create custom parameters of **Environment**, **Enumeration**, and **String** respectively and enable their **Runtime Settings**.

Step 3 Click **Deployment Actions** and reference custom parameters in the actions details.

Step 4 Click **Save and Deploy**. In the dialog box that is displayed, assign values to the parameters again.

Step 5 Click **OK** to save and deploy the application.

----End

NOTE

1. When CodeArts Pipeline is associated with an application, parameters can be dynamically bound.
2. When CodeArts Pipeline is running, the entered parameter values will be replaced in the application and run.
3. After you add a task action of the deployment type to the pipeline task and select an application with **Runtime Settings** parameters, parameters can be dynamically configured when the pipeline is executed.

6 Managing Permissions

The permissions of CodeArts Deploy are three-layered from top to bottom to manage user behaviors.

Project-Level Permissions

Path for project-level permissions:

Step 1 Log in to the CodeArts platform.

Step 2 Click the target project name to access the project.

Step 3 Choose **Settings** > **General** > **Service Permissions**. The **Permission** page is displayed.

----End

Table 6-1 Project-level permissions

Description										
Role/ Operation	**View**	**Create**	**Edit**	**Delete**	**Deploy**	**Clone**	**Disable**	**Create Environment**	**Assign Permissions**	**Manage Groups**
Project manager	√	√	√	√	√	√	√	√	√	√
Project administrator	√	√	√	√	√	√	√	√	√	√
Product manager	√	×	×	×	×	×	×	×	×	×

Description										
Test manager	√	×	×	×	×	×	×	×	×	×
O&M manager	√	×	×	×	×	×	×	√	×	×
System engineer	√	√	√	√	√	√	√	×	×	√
Committer	√	√	√	√	√	√	×	×	×	√
Developer	√	√	√	√	√	√	×	×	×	√
Tester	√	×	×	×	×	×	×	×	×	×
Participant	√	×	×	×	×	×	×	×	×	×
Viewer	√	×	×	×	×	×	×	×	×	×

Application-Level Permissions

Path for application-level permissions:

Step 1 Log in to the CodeArts platform.

Step 2 Click the target project name to access the project.

Step 3 Choose **CICD** > **Deploy**.

Step 4 Click the target application name to access the application.

Step 5 Click **Edit**. The deployment actions page is displayed.

Step 6 Click **Permissions**. The application permission management page is displayed.

----End

Table 6-2 Default application-level permissions

Description								
Role/ Operation	**View**	**Edit**	**Delete**	**Deploy**	**Clone**	**Disable**	**Create Environment**	**Assign Permissions**

Description								
App creator	√	√	√	√	√	√	√	√
Project administrator	√	√	√	√	√	√	√	√
Project manager	√	√	√	√	√	√	√	√
Product manager	√	×	×	×	×	×	×	×
Test manager	√	×	×	×	×	×	×	×
O&M manager	√	×	×	×	×	×	√	×
System engineer	√	√	√	√	√	√	×	×
Committer	√	√	√	√	√	×	×	×
Developer	√	√	√	√	√	×	×	×
Tester	√	×	×	×	×	×	×	×
Participant	√	×	×	×	×	×	×	×
Viewer	√	×	×	×	×	×	×	×

NOTE

- Roles with the **Assign Permissions** permission can modify the permission matrix, but permissions of the **Project manager** and **App creator** roles cannot be modified.
- **Committer**, **project_admin**, **Project manager**, **Developer** , and **System engineer** have the permission to create applications.
- If you do not have the **Edit** permission, the editing page cannot be displayed.
- If you have the **Edit** permission but do not have **Assign Permissions**, the **Edit**, **Delete**, **View**, **Deploy**, and **Clone** buttons are unavailable.

Resource-Level Permissions

Path for host cluster permissions:

Step 1 Log in to the CodeArts platform.

Step 2 Click the target project name to access the project.

Step 3 Choose **Settings > General > Basic Resources**. The **Host Clusters** page is displayed by default.

Step 4 Click the *** icon in the operation column of a cluster, click **Manage Permissions**, and configure operation permissions for each role.

----End

Table 6-3 Host cluster permission

Description						
Role/ Permission	**View**	**Edit**	**Delete**	**Add Host**	**Clone Host**	**Assign Permissions**
Host cluster creator	√	√	√	√	√	√
Project administrator	√	√	√	√	√	√
Project manager	√	√	√	√	√	√
Product manager	√	×	×	×	×	×
Test manager	√	×	×	×	√	×
O&M manager	√	×	×	×	√	×
System engineers	√	×	×	×	×	×
Committer	√	×	×	×	×	×
Developer	√	√	√	√	√	×
Tester	√	×	×	×	√	×
Participant	√	×	×	×	√	×
Viewer	√	×	×	×	√	×

NOTE

Roles with **Manage Permissions** can modify the permission matrix (including the permission to create host clusters), but permissions of the **Project admin** and **Host cluster creator** roles cannot be modified.

Only the **Project admin**, **Project manager**, **Operation manager**, and **Developer** have the permission to create host clusters.

Path for environment permissions:

Step 1 Log in to the CodeArts platform.

Step 2 Click the target project name to access the project.

Step 3 Choose **CICD** > **Deploy**.

Step 4 Click the target application name to access the application.

Step 5 Click **Edit**. The **Deployment Actions** page is displayed.

Step 6 Choose **Environment Management**. The **Environment Management** page is displayed.

Step 7 Click the icon in the operation column of an environment to configure operation permissions for each role.

----End

Table 6-4 Environment permissions

Description					
Role/ Permission	**View**	**Edit**	**Delete**	**Deploy**	**Assign Permissions**
Environment creator	√	√	√	√	√
Project administrator	√	√	√	√	√
Project manager	√	√	√	√	√
Product manager	√	×	×	×	×
Test manager	√	×	×	×	×
O&M manager	√	√	√	√	√
System engineers	√	√	√	√	×

Description					
Committer	√	√	√	√	×
Developer	√	√	√	√	×
Tester	√	×	×	×	×
Participant	√	×	×	×	×
Viewer	√	×	×	×	×

NOTE

Roles with the **Assign Permissions** permission can modify the permission matrix, but permissions of the **Project creator** and **App creator** roles cannot be modified.

