Software Repository for Container

FAQs

Issue 01

Date 2020-11-09



HUAWEI TECHNOLOGIES CO., LTD.

Copyright © Huawei Technologies Co., Ltd. 2020. All rights reserved.

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

Trademarks and Permissions


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.

Contents

1 General FAQs 1
1.1 What Is SWR? 1
1.2 What Is an Organization? 1
1.3 Is SWR Charged? 1
1.4 Why Can't I Find the ContainerOps Console? 1
1.5 How Many Images Can Be Stored in the SWR? 1
1.6 Are There Quotas for SWR Resources? 2
1.7 When I Create an Organization, Why Does a Message Display Indicating that the Organization Already Exists? 2
1.8 How Do I Install a Container Engine? 2
1.9 How Do I Create a Container Image? 2
1.10 How Do I Create an Image Package? 7
1.11 Does SWR Accelerate Image Pull? 7
1.12 What Is the Bandwidth of SWR? 7
2 Image Management FAQs 8
2.1 Why Does the Login Command Fail to Be Executed? 8
2.2 What Are the Differences Between Long-Term Valid Login Commands and Temporary Login Commands? 9
2.3 Why Does an Image Fail to Be Uploaded Through a Container Engine Client? 9
2.4 Why Does an Image Fail to Be Uploaded Through SWR Console? 10
2.5 Why Is an Image Pushed Using a Container Engine Client to SWR Different in Size From One Uploaded Through the SWR Console? 11
2.6 Why Is an Image Pushed Using a Container Engine Client to SWR Different in Size From One Viewed Using the **docker images** Command? 11
2.7 How Do I Push an Image to SWR Through APIs? 12
2.8 What Protocol Is Used When I Run the **docker push** Command to Push Images to SWR? 12
2.9 How Do I Pull a Container Image on the SWR Console? 12
2.10 Where Are the Images Pulled by Running the **docker pull** Command Stored? How to Copy the Images? 12
2.11 Why Does the **docker pull** Command Fail to Be Executed? 12
2.12 What Should I Do If Images Cannot Be Downloaded from Private Networks? 13
2.13 Can I Pull Images from Another Region? 14
2.14 Can I Synchronize Images Across Regions? 14

 Copyright © Huawei Technologies Co., Ltd.

2.15 Why Can't I See Images Synchronized to the Target Region After Configuring Image Synchronization?..... 14
2.16 Can Existing Images Be Automatically Synchronized?..... 14
2.17 Why Can't IAM Users Configure Image Synchronization?..... 15
2.18 How Do I Synchronize Images Between Organizations in the Same Region?..... 15
2.19 Can Automatic Workload Update Be Triggered When I Push an Image with an Existing Tag?..... 16
2.20 How Do I Verify that the Configured HUAWEI CLOUD SWR Image Accelerator Has Taken Effect?..... 16

 Copyright © Huawei Technologies Co., Ltd.

1 General FAQs

1.1 What Is SWR?

Software Repository for Container (SWR) provides easy, secure, and reliable management of container images throughout their lifecycles, facilitating the deployment of containerized services.

1.2 What Is an Organization?

Organizations enable efficient management of images. Organizations are used to isolate image repositories. With each organization being limited to one company or department, images can be managed in a centralized and efficient manner. A user can access different organizations as long as the user has sufficient permissions. There are three types of user permissions, namely, **read**, **write**, and **manage**.

1.3 Is SWR Charged?

SWR is currently free of charge.

1.4 Why Can't I Find the ContainerOps Console?

ContainerOps subscription is now suspended. If you have subscribed to ContainerOps before June 25, 2020, you can continue to use it. Sorry for any inconvenience caused.

1.5 How Many Images Can Be Stored in the SWR?

SWR has no limit on the number of images. You can upload images as required.

 Copyright © Huawei Technologies Co., Ltd.

1.6 Are There Quotas for SWR Resources?

No quotas are imposed on SWR image repositories. You can create as many image repositories as you need.

Quotas are imposed on the number of organizations a user can create. For more information on quotas, see **Quotas**.

For more information on quotas imposed on SWR organizations and image tags, see **Table 1-1**. To apply for more quotas of organizations, **create a service ticket**.

Table 1-1 SWR resource quotas

Resource type	Quota
Organization	5

1.7 When I Create an Organization, Why Does a Message Display Indicating that the Organization Already Exists?

This is because the name you specified has been used by another user. Try another name. Organization names must be globally unique. For example, when user A creates an organization named **Organization A**, the name **Organization A** is not allowed to be used by any other users.

1.8 How Do I Install a Container Engine?

The container engine is compatible with almost all operating systems. Select an edition that best suits your needs.

NOTE

- SWR uses container engine 1.11.2 or later to upload images.
- If the container engine client is in a private network, bind an elastic IP address (EIP) to the client. This EIP will allow the client to download installation packages from the website.

On a device running Linux, run the following commands to quickly install the container engine:

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

1.9 How Do I Create a Container Image?

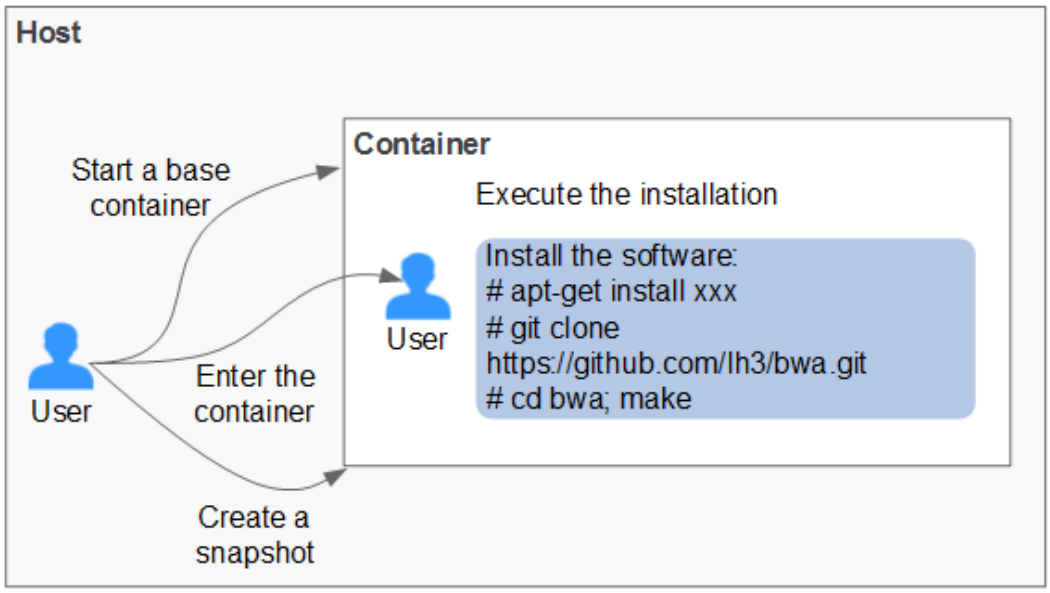
The following two approaches are for you to consider. Approach 1 is for images that will only be updated occasionally whereas approach 2 is for images that will be frequently updated.

Copyright © Huawei Technologies Co., Ltd.

- Approach 1: creating a snapshot. This approach involves three key steps: (1) Start a base container by running a base image (for example, Ubuntu image); (2) install the container engine software inside the base container; (3) create a snapshot of the container.
- Approach 2: creating a Dockerfile to build an image. This approach involves two key steps: (1) Write software installation instructions into a Dockerfile; (2) run the **Docker build** command to build an image from the Dockerfile.

Approach 1: Creating a Snapshot

This approach is suitable for images that will only be updated occasionally.



Procedure:

1. Install the container engine software on a host.
2. Start an empty base container in the interactive mode.

   For example, start a CentOS container in the interactive mode.

   docker run -it centos
3. Run the following commands to install the target software:

   yum install XXX

   git clone https://github.com/lh3/bwa.git

   cd bwa;make

   NOTE

   Install Git in advance and check whether an SSH key is set on the local host.
4. Run the **exit** command to exit the container.
5. Create a snapshot.

   docker commit -m "xx" -a "tsj" container-id tsj/image:tag

   - **-a**: indicates the author of the base image.
   - **container-id**: indicates the ID of the container you have started in step 2. You can run the **docker ps -a** command to query the container ID.

 Copyright © Huawei Technologies Co., Ltd.

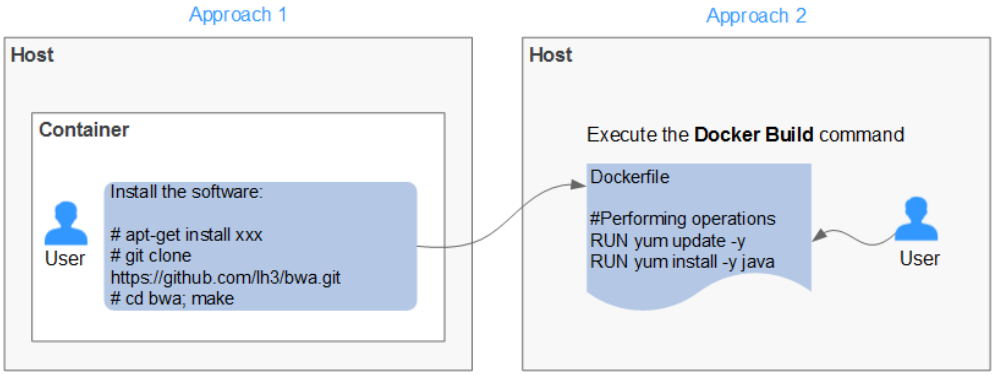
- **-m**: indicates the commit message.
- **tsj/image:tag**: indicates the repository name/image name:tag name.

6. Run the **docker images** command to list the built container image.

Approach 2: Creating a Dockerfile to Build an Image

This approach is suitable for images that will be frequently updated. In **Approach 1**, you create a snapshot of the whole container. This could be demanding if you need to frequently update your images. In this case, **Approach 2** is put forward to automate the image build process.

The idea behind **Approach 2** is to write the process of **Approach 1** into a Dockerfile and then run the **docker build -t tsj/image:tag.** command to automatically build an image from the Dockerfile. In the preceding command, **.** indicates the path to the Dockerfile.



Example Dockerfile:

NOTE

If an external network is required, ensure that network connectivity is available.

```
#Version 1.0.1
FROM centos:latest
MAINTAINER ***u "***u@huaweicloud.com"

# Setting the root user as the executor of subsequent commands
USER root

# Performing operations
RUN yum update -y
RUN yum install -y java

# Using && to concatenate commands
RUN touch test.txt && echo "abc" >>abc.txt

# Setting an externally exposed port
EXPOSE 80 8080 1038

# Adding a network file
ADD https://www.baidu.com/img/bd_logo1.png /opt/

# Setting an environment variable
ENV WEBAPP_PORT=9090

# Setting a work directory
WORKDIR /opt/

# Setting a start command
```

 Copyright © Huawei Technologies Co., Ltd.

```
ENTRYPOINT ["ls"]

# Setting start parameters
CMD ["-a", "-l"]

# Setting a volume
VOLUME ["/data", "/var/www"]

# Setting the trigger operation for a sub-image
ONBUILD ADD . /app/src
ONBUILD RUN echo "on build excuted" >> onbuild.txt
```

For detailed instructions, see *Software Repository for Container Best Practices*.

Basic Syntax of Dockerfile

- FROM:

  It is used to specify the parent image (base image) from which you are building a new image. Except annotations, a Dockerfile must start with a FROM instruction. Subsequent instructions run in this parent image environment until the next FROM instruction appears. You can create multiple images in the same Dockerfile by adding multiple FROM instructions.

- MAINTAINER:

  It is used to specify the information about the author who creates an image, including the username and email address. This parameter is optional.

- RUN:

  It is used to modify an image. Generally, RUN commands are executed to install libraries, and install and configure programs. After a RUN command is executed, an image layer will be created on the current image. The next command will be executed on the new image. The RUN statement can be in one of the following formats:

  - **RUN yum update**: Command that is executed in the **/bin/sh** directory.
  - **RUN ["yum", "update"]**: Directly invoke **exec**.
  - **RUN yum update && yum install nginx**: Use **&&** to connect multiple commands to a RUN statement.

- EXPOSE:

  It is used to specify one or more network ports that will be exposed on a container. If there are multiple ports, separate them by spaces.

  When running a container, you can set **-P** (uppercase) to map the ports specified in EXPOSE to random ports on a host. Other containers or hosts can communicate with the container through the ports on the host.

  You can also use **-p** (lowercase) to expose the ports that are not listed in EXPOSE.

- ADD:

  It is used to add a file to a new image. The file can be a host file, a network file, or a folder.

  - First parameter: source file (folder)

    - If a relative path is used, this path must correspond to the directory where the Dockerfile is located.
    - If a URL is used, the file needs to be downloaded first and then added to the image.

 Copyright © Huawei Technologies Co., Ltd.

  - Second parameter: target path
    - If the source file is in the .zip or .tar file, the container engine decompresses the file and then adds it to the specified location of the image.
    - If the source file is a compressed network file specified by a URL, the file will not be decompressed.
- VOLUME:

  It is used to create a mount point for a specified path (file or folder) in the image. Multiple containers can share data through the same mount point. Even if one of the containers is stopped, the mount point can still be accessed.
- WORKDIR:

  It is used to specify a new work directory for the next command. The directory can be an absolute or a relative directory. WORKDIR can be specified multiple times as required. When a container is started, the directory specified by the last WORKDIR command is used as the current work directory of the container.
- ENV:

  It is used to set an environment variable for running the container. When running the container, you can set **-e** to modify the environment variable or add other environment variables.

  Example:

  docker run -e WEBAPP_PORT=8000 -e WEBAPP_HOST=www.example.com ...
- CMD:

  It is used to specify the default command for starting a container.
- ENTRYPOINT:

  It is used to specify the default command for starting a container. Difference: For ENTRYPOINT, parameters added to the image during container running will be spliced. For CMD, these parameters will be overwritten.
  - If the Dockerfile specifies that the default command for starting a container is **ls -l**, the default command **ls -l** will be run accordingly. For example:
    - **ENTRYPOINT ["ls", "-l"]**: The program and parameter for starting a container are set to be **ls** and **-l** respectively.
    - **docker run centos**: The **docker run centos ls -l** command is run by default for starting a CentOS container.
    - **docker run centos -a**: When the **-a** parameter is added for starting a CentOS container, the **docker run centos ls -l -a** command is run by default.
  - If the Dockerfile specifies that the default command for starting a container is **--entrypoint** but you need to replace the default command, you can add **--entrypoint** parameters to replace the configuration specified in Dockerfile. Example:

    docker run gutianlangyu/test --entrypoint echo "hello world"

 Copyright © Huawei Technologies Co., Ltd.

- USER:

  It is used to specify the user or UID for running the container, and running the RUN, CMD, or ENTRYPOINT command.

- ONBUILD:

  Trigger command. During image build, the image builder of the container engine saves all commands specified by the ONBUILD command to the image metadata. These commands will not be executed in the process of building the current image. These commands will be executed only when a new image uses the FROM instruction to specify the parent image as the current image.

  Using the FROM instruction to build a child image based on the parent image created by the Dockerfile:

  ONBUILD ADD . /app/src: Automatically run the **ADD . /app/src** command.

1.10 How Do I Create an Image Package?

Run the **docker save** command to compress the container image into a .tar or .tar.gz package. The command format is as follows:

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

[OPTIONS] can be set to **--output** or **-o**, indicating that the image is exported to a file.

Example:

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

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

$ docker save --output nginx.tar nginx
$ ls -sh nginx.tar
108M nginx.tar

$ docker save -o nginx-all.tar nginx
$ docker save -o nginx-latest.tar nginx:latest
```

1.11 Does SWR Accelerate Image Pull?

Yes. SWR accelerates the pull of images through proprietary acceleration technology.

1.12 What Is the Bandwidth of SWR?

The bandwidth of SWR dynamically changes based on actual usage.

 Copyright © Huawei Technologies Co., Ltd.

2 Image Management FAQs

2.1 Why Does the Login Command Fail to Be Executed?

Possible causes are as follows:

1. The container engine is not properly installed, in which case the following error is reported:

   ```
   [root@ecs-514e ~]# docker login -u cn-north-1@U00UZ[redacted]ZIO -p 98365d950b2a[redacted]
   413b8a3fa swr.cn-north-1.myhuaweicloud.com
   -bash: docker: command not found
   ```

   Solution: Reinstall the container engine. For details, see **Installing a Container Engine**.

   - It is advised to install container engine 1.11.2 or later because earlier versions do not support image push to SWR.
   - If the container engine client is in a private network, bind an elastic IP address (EIP) to the client. This EIP will allow the client to download installation packages from the website.

2. The temporary login command has expired, or the regional project name, access key (AK), or login key in the command is incorrect, in which case the following error is reported:

   ```
   Get https://swr.cn-[redacted].myhuaweicloud.com/v2/: unauthorized: authentication required
   ```

   Solution: Log in to the SWR console. In the navigation pane on the left, choose **My Images**. On the page displayed, click **Upload Through Client**. Then you can find the information on how to obtain a login command.

   a. To obtain a temporary login command, click **Generate a temporary login command** and then click  to copy the command.

   b. To obtain a long-term valid login command, click **learn how to obtain a login command that has long-term validity**. For details, see **Obtaining a Long-Term Valid Login Command** and follow the guidelines.

3. The image repository address in the login command is incorrect, in which case the following error is reported:

 Copyright © Huawei Technologies Co., Ltd.

```
INFO[0003] Error logging in to v2 endpoint, trying next endpoint: Get https://swr.cn-north.myhuaweicloud.com/v2/: dial tcp: l
ookup swr.cn-north.myhuaweicloud.com on 100.[redacted]:53: no such host
INFO[0003] Error logging in to v1 endpoint, trying next endpoint: Get https://swr.cn-north.myhuaweicloud.com/v1/users/: dial
tcp: lookup swr.cn-north.myhuaweicloud.com on 100.[redacted]:53: no such host
Get https://swr.cn-north.myhuaweicloud.com/v1/users/: dial tcp: lookup swr.cn-north.myhuaweicloud.com on 100.[redacted]:53: no
 such host
```

Solutions:

a. Correct the image repository address in the login command. The format of the address should be **swr.*regional project name*.myhuaweicloud.com**, for example, swr.cn-north-1.myhuaweicloud.com. To obtain the regional project name, go to the **My Credentials** page. On the **Projects** tab page, you can find the project name corresponding to your current region.

b. Generate a temporary login command. For detailed instructions, see **2**.

2.2 What Are the Differences Between Long-Term Valid Login Commands and Temporary Login Commands?

- Temporary login commands will expire after 24 hours.
- Long-term valid login commands are permanently valid.

  After you obtain a long-term valid login command, your temporary login commands will still be valid as long as they are in their validity periods.

2.3 Why Does an Image Fail to Be Uploaded Through a Container Engine Client?

1. **Problem**: When you push an image to SWR through your container engine client, the operation fails with the following information returned.

   ```
   [root@ecs-paas-[redacted] ~]# docker push swr.cn-north-1.myhuaweicloud.com/cwx5/nginx:v1
   The push refers to repository [swr.cn-north-1.myhuaweicloud.com/cwx5/nginx]
   d2f0b6dea592: Preparing
   197c666de9dd: Preparing
   cf5b3c6798f7: Preparing
   denied: you do not have the permission
   ```

   Possible causes: Normally, when you specify an organization that does not exist for the image to be pushed to, SWR will first create an organization with the name you specified and push the image to this organization. Image push may fail when the organization name you specified has already been used by another user or the maximum number of organizations that you are allowed to create, which is set to five by default, has been reached.

   Solutions:

   – If the organization name has been occupied, create a valid organization first, and then push the image. For details on how to create an organization, see **Creating an Organization**.

   – If the maximum number of organizations you are allowed to create has been reached, you can either push the image to an existing organization, or increase the quota by submitting a **service ticket**.

2. **Problem**: When you push an image to SWR through your container engine client, the operation fails with the following information returned.

 Copyright © Huawei Technologies Co., Ltd.

```
[root@ecs-paas- ~]# docker push swr.cn-north-1.myhuaweicloud.com/cwx4/nginx:v2
The push refers to repository [swr.cn-north-1.myhuaweicloud.com/cwx4/nginx]
tag does not exist: swr.cn-north-1.myhuaweicloud.com/cwx4/nginx:v2
```

```
[root@ecs-paas- ~]# docker push swr.cn-north-1.myhuaweicloud.com/cwx1/nginx:v1
The push refers to repository [swr.cn-north-1.myhuaweicloud.com/cwx1/nginx]
An image does not exist locally with the tag: swr.cn-north-1.myhuaweicloud.com/cwx1/nginx
```

Possible cause: The image or image tag to be pushed does not exist.

Solution: Run the **docker images** command to view all the local images. Check the target image name and tag, and push the image again.

3. **Problem**: When you push an image to SWR through your container engine client, the operation fails with the following information returned.

```
[root@ecs-paas- ~]# docker push swr.cn-north-1.myhuaweicloud.com/cwx4/nginx.sh:v1
The push refers to repository [swr.cn-north-1.myhuaweicloud.com/cwx4/nginx.sh]
d2f0b6dea592: Retrying in 1 second
197c666de9dd: Retrying in 1 second
cf5b3c6798f7: Retrying in 1 second
name invalid: 'repository' is invalid
```

Possible cause: The organization name or image name does not comply with the naming rules.

Solution: The regular expressions of the organization (namespace) name and image (repository) name are as follows:

Organization name: ^([a-z]+(?:(?:(?:_|__|[-]*)[a-z0-9]+)+)?)$. Enter a maximum of 64 characters.

Image name: ^([a-z0-9]+(?:(?:(?:_|__|[-]*)[a-z0-9]+)+)?)$. Enter a maximum of 128 characters.

Specify a valid organization name or image name, and push the image again.

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

SWR has strict requirements on image name and address format. Invalid image names or addresses could lead to upload failures.

1. **Problem**: When you upload an image to SWR through the SWR console, an error message is displayed, indicating that the image format is invalid.

   Possible cause: Invalid image address leads to upload failure.

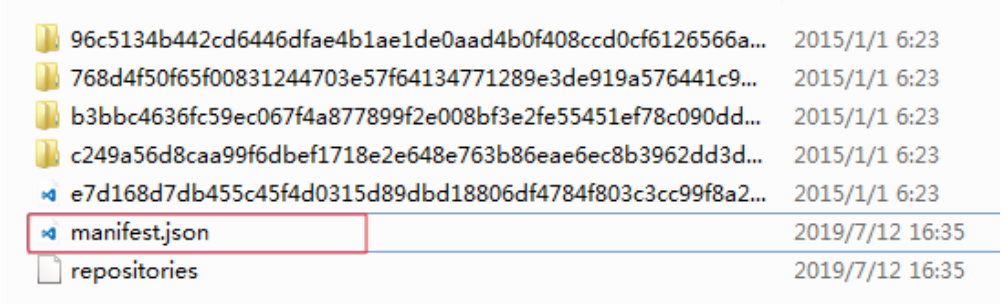
   Solution: Check and modify the image address according to the following sample and description. ***tag*** at the end of the address can be omitted, in which case it indicates that the image tag to be uploaded is the latest. Other parts of the image address are compulsory.

   Example: swr.cn-north-1.myhuaweicloud.com/repo_namespace/repo_name:tag

   - **swr.cn-north-1.myhuaweicloud.com** indicates the SWR image repository address.
   - **repo_namespace** indicates the organization name. Its regular expression is ^([a-z]+(?:(?:(?:_|__|[-]*)[a-z0-9]+)+)?)$. Enter a maximum of 64 characters.
   - **repo_name:tag** indicates the image name and tag. The image name regular expression is ^([a-z0-9]+(?:(?:(?:_|__|[-]*)[a-z0-9]+)+)?)$. Enter a maximum of 128 characters.

 Copyright © Huawei Technologies Co., Ltd.

To view the image address, decompress the image. Open the **manifest.json** file, and check the value of **RepoTags**.



2. **Problem**: When you upload an image to SWR through the SWR console, the upload progress is stuck and the upload task times out at the end.

   Possible cause: Invalid image name leads to upload failures.

   Solution: Modify the image name according to the naming rules, and try uploading the image again.

NOTICE

It is the image name in the **repositories** and **manifest.json** files that should be checked and modified rather than the name of the image file you select and upload on the SWR console.

2.5 Why Is an Image Pushed Using a Container Engine Client to SWR Different in Size From One Uploaded Through the SWR Console?

Image layers are compressed into TGZ packages when images are pushed to SWR using a container engine client, whereas when they are uploaded through the SWR console, they are only packed without being compressed. Therefore, the same image will be of different sizes when it is uploaded in these two different ways.

2.6 Why Is an Image Pushed Using a Container Engine Client to SWR Different in Size From One Viewed Using the docker images Command?

Image layers are compressed into TGZ packages when images are pushed to SWR using a container engine client, whereas images you view locally by running the **docker images** command are not compressed. Therefore, the sizes are different.

 Copyright © Huawei Technologies Co., Ltd.

2.7 How Do I Push an Image to SWR Through APIs?

SWR APIs for image push are currently not available. Instead, you can run the **docker push** command to push images to SWR. For details, see **Uploading an Image Through a Container Engine Client**.

2.8 What Protocol Is Used When I Run the docker push Command to Push Images to SWR?

The HTTPS protocol is used when you run the **docker push** command to push images to SWR.

2.9 How Do I Pull a Container Image on the SWR Console?

Currently, images cannot be directly pulled by using the SWR console. Run the following command to pull an image:

docker pull [*Image repository address*]/[*Organization name*]/[*Image name:tag*]

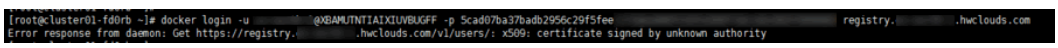
2.10 Where Are the Images Pulled by Running the docker pull Command Stored? How to Copy the Images?

Images pulled by running the **docker pull** command are stored on your local hosts. You can run the **docker save** command to save images into TAR archive files.

2.11 Why Does the docker pull Command Fail to Be Executed?

Problem: When you run the **docker pull** command to pull an image from SWR, error message "x509: certificate signed by unknown certificates" is displayed.

Possible cause: A container engine client and SWR communicate with each other using HTTPS. When the client verifies the server certificate and finds that the root certificate installed on the client is incomplete, the error message "x509: certificate signed by unknown certificates" is displayed.



Solution: If you trust the server, skip certificate authentication. Specifically, manually configure the container engine startup parameters using either of the following two methods. Replace ***Image repository address*** with the actual SWR repository address.

 Copyright © Huawei Technologies Co., Ltd.

- Add the following configuration to the **/etc/docker/daemon.json** file. If the file does not exist, manually create it. Ensure that two-space indents are used in the configuration.

```
{
  "insecure-registries":["Image repository address"]
}
```

- /etc/sysconfig/docker:

```
INSECURE_REGISTRY='--insecure-registry=Image repository address'
```

After configuration, run the **systemctl restart docker** or **service restart docker** command to restart the container engine.

2.12 What Should I Do If Images Cannot Be Downloaded from Private Networks?

This is usually caused by incorrect DNS configurations. You can solve the problem by taking either of the following approaches:

- Approach 1:

  Add a private DNS server address to the **/etc/resolv.conf** file. If you are not sure which private DNS server address to use in your region, see **What Are the Private DNS Server Addresses Provided by the DNS Service**.

  NOTE

  The newly added DNS server address must be placed before all existing DNS server addresses.

  Updates to DNS configurations take effect immediately after the **/etc/resolv.conf** file is saved.

- Approach 2:

  Restarting an Elastic Cloud Server (ECS) will invalidate updates to the **/etc/resolv.conf** file on this ECS. You will have to update the file again. To avoid the repetitive modifications, take the following steps to change the DNS server address of the VPC subnet to the private DNS server address available in your region.

  NOTICE

  Any changes to a VPC subnet will affect all ECSs in this subnet.

Step 1 Log in to the management console and choose **Service List** > **Network** > **Virtual Private Cloud** for VPC console.

Step 2 **Change the DNS server address of the VPC subnet to a private DNS server address**Change the DNS server address of the VPC subnet to a private DNS server address.

Step 3 Restart the ECS, and check whether the private DNS server address is contained in the **/etc/resolv.conf** file and whether this address is placed before other DNS server addresses.

----End

 Copyright © Huawei Technologies Co., Ltd.

2.13 Can I Pull Images from Another Region?

Cross-region image pull over public network is supported. Ensure that you have obtained a correct login command. For details on how to obtain a login command, see **Uploading an Image Through a Container Engine Client** and **Obtaining a Long-Term Valid Login Command**.

2.14 Can I Synchronize Images Across Regions?

Currently, images can be synchronized only among the following regions: CN North-Beijing1, CN North-Beijing4, CN East-Shanghai1, CN East-Shanghai2, CN South-Guangzhou, AP-Hong-Kong, and AP-Singapore.

For other regions, images cannot be automatically synchronized. You can pull the images to be synchronized to your local host, and manually push them to the repositories in the target regions. For more information on how to push images, see **Uploading an Image Through a Container Engine Client**.

2.15 Why Can't I See Images Synchronized to the Target Region After Configuring Image Synchronization?

Problem: After configuring image synchronization, I cannot see images synchronized to the target region I specified. For example, after I successfully configured automatic synchronization for image **nginx_01** from CN North-Beijing4 to CN South-Guangzhou, the **nginx_01** image is not displayed in the CN South-Guangzhou, the target region.

Cause: Images configured with automatic image synchronization will be synchronized to target regions only when the images are updated or have newly pushed tags. In this example, you will find image **nginx_01** synchronized to CN South-Guangzhou after you push a new tag for **nginx_01**.

If you want images be immediately synchronized to target regions, manually synchronize one or multiple image tags in batches on the SWR console.

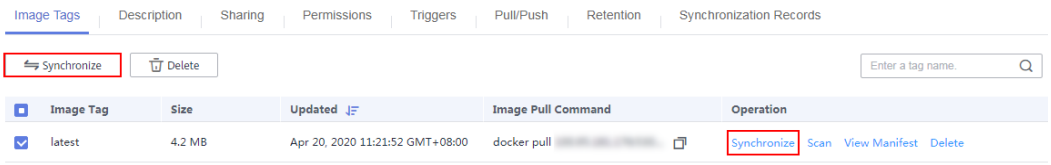
2.16 Can Existing Images Be Automatically Synchronized?

Setting image synchronization enables you to automatically synchronize newly pushed images to the target organizations in the target regions you specified. When images are updated, corresponding images in the target organization are automatically updated, accordingly.

Existing images pushed before you set image synchronization will be not automatically synchronized. To synchronize existing images, on the image details page, select the target image tags and click **Synchronize**.

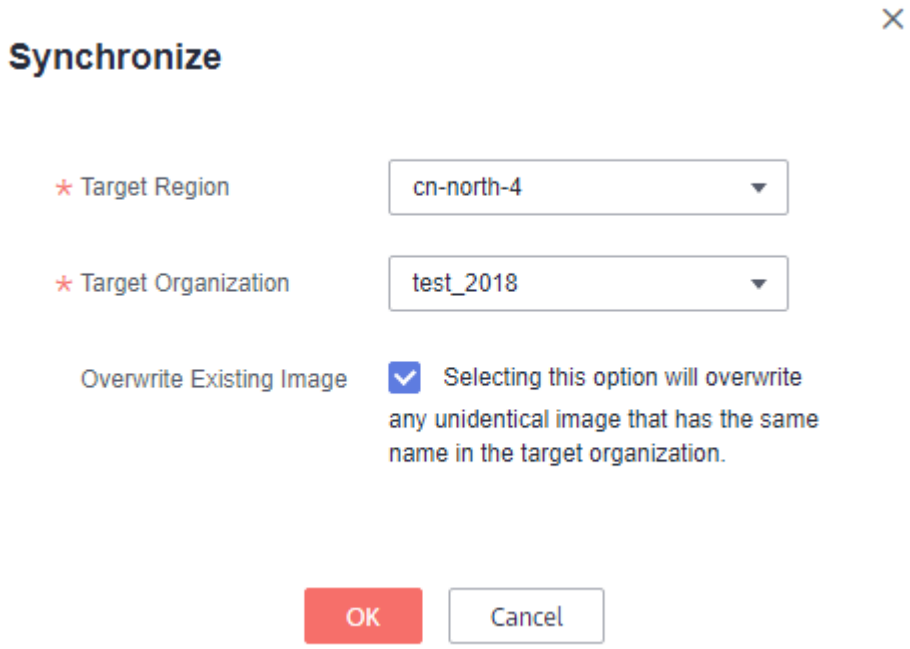
Copyright © Huawei Technologies Co., Ltd.

Figure 2-1 Synchronizing existing images



In the dialog box displayed, select the target region, target organization, and whether to overwrite any unidentical image that has the same name in the target organization. Click **OK**.

Figure 2-2 Synchronizing images



2.17 Why Can't IAM Users Configure Image Synchronization?

Currently, only accounts and IAM users with administrator permissions can configure image synchronization. IAM users without administrator permissions will be able to configure image synchronization in later versions.

2.18 How Do I Synchronize Images Between Organizations in the Same Region?

The image synchronization function enables you to synchronize images only between organizations in different regions. You can share private images with other accounts in the same region and grant them permissions to pull images.

 Copyright © Huawei Technologies Co., Ltd.

Shared images can be viewed on the SWR console by clicking **My Images > Shared Images > *Target Image Name***. On the image details page displayed, information such as image tags and image pull commands is provided.

2.19 Can Automatic Workload Update Be Triggered When I Push an Image with an Existing Tag?

No. Push the image with a new tag to trigger automatic workload update. Triggers enable CCE workloads to be automatically updated when the workloads' corresponding images are updated. Ensure that you push the images with new tags. Existing tags cannot trigger the update.

2.20 How Do I Verify that the Configured HUAWEI CLOUD SWR Image Accelerator Has Taken Effect?

Ping is disabled for SWR accelerator addresses. To verify whether the SWR image accelerator has been successfully configured, run the **docker info** command to see whether the value of **Registry Mirrors** is the SWR accelerator address. If it is, the accelerator has taken effect.

Copyright © Huawei Technologies Co., Ltd.