ALIBABA CLOUD

Alibaba Cloud

Elastic Container Instance
Image

Document Version: 20220708

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
⚠ Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	⚠ **Danger:** Resetting will result in the loss of user configuration data.
🔔 **Warning**	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	🔔 **Warning:** Restarting will cause business interruption. About 10 minutes are required to restart an instance.
📢 **Notice**	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	📢 **Notice:** If the weight is set to 0, the server no longer receives new requests.
ⓘ **Note**	A note indicates supplemental instructions, best practices, tips, and other content.	ⓘ **Note:** You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.Overview of the image cache feature

When you create an elastic container instance, you can use the image cache feature to accelerate image pulling and reduce the instance startup period. This topic describes the basic features and billing items of image caches. This topic also describes how to create and use image caches.

Features

When you run containers, Elastic Container Instance must pull the specified images. It can take a long time to pull large images over an unstable network when elastic container instances start. To accelerate the creation of elastic container instances, Elastic Container Instance provides the image cache feature. The image cache feature allows you to create snapshots for the required images and then cache the snapshots before you can use the cached snapshots to create elastic container instances. This prevents or minimizes the downloads of images lays and accelerates the creation of instances.

For example, if you create an elastic container instance from a Flink image that is 386.26 MB in size on Docker Hub, it takes 50 seconds to pull the image. If you cache the image snapshot, it requires only 5 seconds to pull the image.

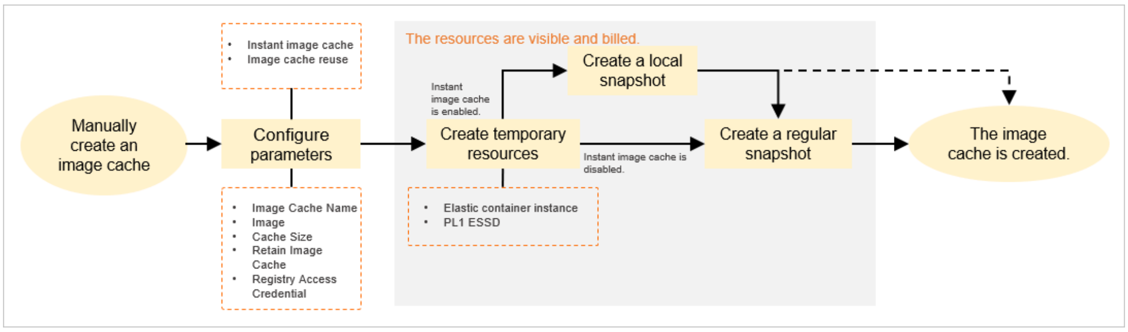
> **Note**
>
> The amount of time reduced varies based on the number of the images in the image cache, the size of the image used to create the elastic container instance, and the network over which Elastic Container Instance connects to the image repository.

Modes of creating an image cache

An image cache can be manually or automatically created.

- Manually create an image cache

  You can configure parameters such as the name and size of your image cache based on your business requirements. The following figure shows the procedure of manually creating an image cache.



Manually create an image cache
Configure parameters
Instant image cache
Image cache reuse
Image Cache Name
Image
Cache Size
Retain Image Cache
Registry Access Credential
The resources are visible and billed.
Instant image cache is enabled.
Create a local snapshot
Create temporary resources
Instant image cache is disabled.
Create a regular snapshot
Elastic container instance
PL1 ESSD
The image cache is created.

  - A snapshot is created for each image cache. The image cache snapshots that you manually create are managed by yourself.

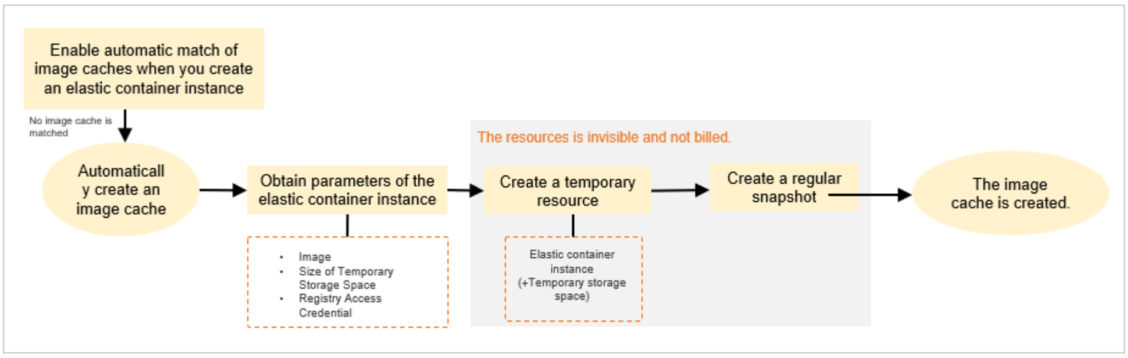
  - When you manually create an image cache, the system creates a temporary elastic container instance and attaches a temporary performance level-1 (PL1) enhanced SSD (ESSD) to the temporary instance. The temporary instance has two vCPUs and 4 GiB of memory. The temporary ESSD is used to create the image cache snapshot. After the image cache is created, the temporary instance and ESSD are automatically released.
  - The instant image cache feature can be used to reduce the period of creating an image cache. After the instant image cache feature is enabled, the system instantly creates a temporary local snapshot. After the local snapshot is created, you can use the local snapshot to create elastic container instances.

    > **Note**
    >
    > After the local snapshot is created, the system starts to create a regular snapshot. While the system is creating the regular snapshot, you can use the local snapshot to create elastic container instances. After the regular snapshot is created, the system deletes the local snapshot and you can use the regular snapshot to create elastic container instances.

  - The image cache reuse feature can be used to accelerate the creation of an image cache. After you enable this feature, the system automatically matches existing image caches when an image cache is created. If existing image caches have duplicate image layers with the image cache that you want to create, the system reuses the duplicate image layers to accelerate the creation of the new image cache.

- Automatically create an image cache

  You can enable automatic match of image caches when you create an elastic container instance. Then, if no image cache is matched, the system automatically creates an image cache while the system creates the elastic container instance. The following figure shows the procedure of automatically creating an image cache.


Enable automatic match of image caches when you create an elastic container instance
No image cache is matched
Automaticall y create an image cache
Obtain parameters of the elastic container instance
Image
Size of Temporary Storage Space
Registry Access Credential
The resources is invisible and not billed.
Create a temporary resource
Elastic container instance (+Temporary storage space)
Create a regular snapshot
The image cache is created.

  - A snapshot is created for each image cache. The image cache snapshots that the system automatically creates are managed by Alibaba Cloud.

- When the system automatically creates an image cache, the system creates a temporary elastic container instance and uses the storage space of the instance for the creation of the image cache snapshot. After the system automatically creates the image cache, the temporary instance is automatically released.

  > ? Note
  >
  > By default, a temporary elastic container instance has a storage space of 40 GiB, including 20 GiB for the system and 20 GiB for the image cache snapshot. If the image cache that you want to create is no greater than 20 GiB, you do not need to declare additional temporary storage space. If the image cache that you want to create is greater than 20 GiB, you must declare additional temporary storage space.

The following table compares manual image caches with automatic image caches.

Item	Manual image cache	Automatic image cache
Name	You can customize a name.	The system automatically generates a name in the format of `auto-create-for-instance ID`.
Size	The default value is 20 GiB. You can customize the size.	The default value is 20 GiB. If you declare the size of the temporary storage space when you create the elastic container instance, the size of the image cache = 20 GiB + the size of the temporary storage space.
Retention period	By default, manual image caches are permanently retained. You can also configure the retention period in days. Manual image caches are automatically deleted after their retention periods end.	The retention period is managed by Alibaba Cloud. If an automatic image cache is not used within 30 days after it is created, the system automatically deletes the image cache.
Image cache reuse	You can enable this feature to reuse duplicate image layers of existing image caches and accelerate the creation of an image cache.	This feature is not supported.
Instant image cache	You can enable this feature to create a temporary local snapshot and reduce the period used to create an image cache.	This feature is not supported.

Item	Manual image cache	Automatic image cache
Quota limit	You can view the quota by calling the ListUsage API operation. You can also view the quota by logging on to Elastic Container Instance console and clicking **Equity quota** in the left-side navigation pane. If the quota cannot meet your business requirements, submit a ticket to request a quota increase.	The quota is half of the quota for manual image caches and does not consume the quota for manual image caches.
Cost of use	You are charged for the creation and use of manual image caches.	You are charged only for the use, not for the creation of automatic image caches.

Modes of using an image cache

You can use the image cache feature to accelerate the creation of an elastic container instance. If you want to use the image cache feature to create an elastic container instance, you can configure automatic match of image caches or specify the image cache to be used to create the instance.

- Configure automatic match of image caches

  The system automatically matches the most suitable image cache. The system matches image caches in the following order:

  i. Filters all qualified image caches in the region. The size of each image cache satisfies the size requirements of the temporary storage space of the elastic container instance.

  ii. Selects the most suitable image cache based on the match policies. Image caches are matched based on the following attributes in descending order of priority: the match degree, the size, and the creation time of the image cache.

     - Match degree: indicates how an image cache matches the elastic container instance in the image repository and version. The image cache with the highest match degree is assigned the highest priority.

     - Image cache size: indicates the size of the image cache. The image cache whose size is similar to the size of the elastic container instance is assigned the highest priority.

     - Creation time: indicates the time when the image cache is created. The most recently created image cache is assigned the highest priority.

  > Note
  >
  > If no image cache is matched, the system automatically creates an image cache when the system creates the elastic container instance. Images are pulled when the system creates the instance. We recommend that you configure the image pulling policy to IfNotPresent to prevent repeatedly downloading image layers and affecting the use of the image cache feature.

- Specify the image cache to be used to create the instance

  Specify the image cache that is to be used to create the elastic container instance. The image cache must be in the Ready state.

Limits

- A single image cache can contain up to 20 images.
- To create an image cache, you need to pull container images. Therefore, the creation period of the image cache depends on factors such as the number of the images to be pulled, the size of the image used to create the image cache, and network conditions.
- When you manually create an image cache, the container image that you specify is used. When the system automatically creates an image cache, the container image that you declare in the elastic container instance is used.
  - If you select a private image, which is not hosted in Alibaba Cloud Container Registry (ACR), you must provide the access credentials including the IP address, username, and password of the repository to which the private image belongs.
  - If the image that you select such as a Docker image needs to be pulled over the Internet, you must configure an elastic IP address (EIP) or a NAT gateway for the elastic container instance. For more information, see Enable Internet access.
  - If the image cannot be pulled due to some reasons such as a timeout error of the remote repository, we recommend that you use ACR to upload the image to an Alibaba Cloud image repository.
- For manual image caches, we recommend that you configure the retention period when you create the image caches to avoid unnecessary snapshot fees. For automatic image caches, Alibaba Cloud manages the snapshots. You do not need to care for the retention period of the snapshots.

Billing

The billing of image caches involves the creation and use of image caches.

> ? **Note**
>
> For automatic image caches, you are charged only for the use cost, not for the creation cost.

- Creation cost of manual image caches
  - Cost of the temporary elastic container instance and ESSD: You are charged for the temporary elastic container instance and ESSD that are generated when you create an image cache.
  - Cost of the snapshot: A snapshot is created for each image cache. You are charged for the retention of the snapshot.
  - Cost for the local snapshot: If the instant image cache feature is enabled, you are charged for the local snapshot that is instantly created for use during the creation of the image cache.
- Use cost of image caches
  - When you use a manual image cache to create an elastic container instance, the system automatically attaches a disk of the same size as the image cache to the elastic container instance. You are charged for the disk and the elastic container instance.

- If you declare a temporary storage space when you use an automatic image cache to create an elastic container instance, you are charged for the temporary storage space and the elastic container instance. If you do not declare a temporary storage space, you are charged only for the elastic container instance.

For more information, see Billing.

2.Manage an image cache

Image caches accelerate image pulling and reduce the startup time of elastic container instances. This topic describes how to create, query, update, and delete an image cache.

Create an image cache

You can manually create an image cache by using the Elastic Container Instance console or by calling an API operation in OpenAPI Explorer.

> Note
>
> The system can automatically create image caches. You can enable automatic match of image caches when you create an elastic container instance. Then, if no image cache is matched, the system automatically creates an image cache.

Create an image cache by calling an API operation

You can call the CreateImageCache API operation to create an image cache. The following table describes the parameters of the operation. For more information, see CreateImageCache.

Parameter	Type	Example	Description
RegionId	String	cn-hangzhou	The region ID of the image cache.
ImageCacheName	String	testcache	The name of the image cache.
Image.N	RepeatList	registry-vpc.cn-hangzhou.aliyuncs.com/eci_open/nginx:1.15.10-perl	Container image N used to create the image cache.
SecurityGroupId	String	sg-uf66jeqopgqa9hdn****	The ID of the security group.
VSwitchId	String	vsw-uf6h3rbwbm90urjwa****	The ID of the vSwitch
ImageCacheSize	Integer	20	The size of the image cache. The default value is 20 GiB.

Parameter	Type	Example	Description
RetentionDays	Integer	7	The retention period of the image cache. When the specified retention period ends, the image cache is deleted. Unit: days.
AutoMatchImageCache	Boolean	false	Specifies whether to enable image cache reuse. If you enable this feature and the image cache that you want to create and an existing image cache contain duplicate image layers, the system reuses the duplicate image layers to create the new image cache. This accelerates the creation of the image cache. Default value: false.
Flash	Boolean	true	Specifies whether to enable the instant image cache feature. This feature can accelerate the creation of image caches. Default value: false.

When you create an image cache, you can use Container Registry (ACR) Enterprise Edition to pull images without the need to enter the password. The following table describes the parameters configured to use ACR. For more information, see Configure password-free access to pull images from a Container Registry Enterprise Edition instance.

Parameter	Type	Example	Description
AcrRegistryInfo.N.Domain.N	String	test****-registry.cn-beijing.cr.aliyuncs.com	Domain name N of Container Registry Enterprise Edition instance N. All domain names of instance N are displayed by default. You can specify domain names. Separate multiple domain names with commas (,).
AcrRegistryInfo.N.InstanceName	String	test****	The name of Container Registry Enterprise Edition instance N.
AcrRegistryInfo.N.InstanceId	String	cri-nwj395hgf6f3****	The ID of Container Registry Enterprise Edition instance N.

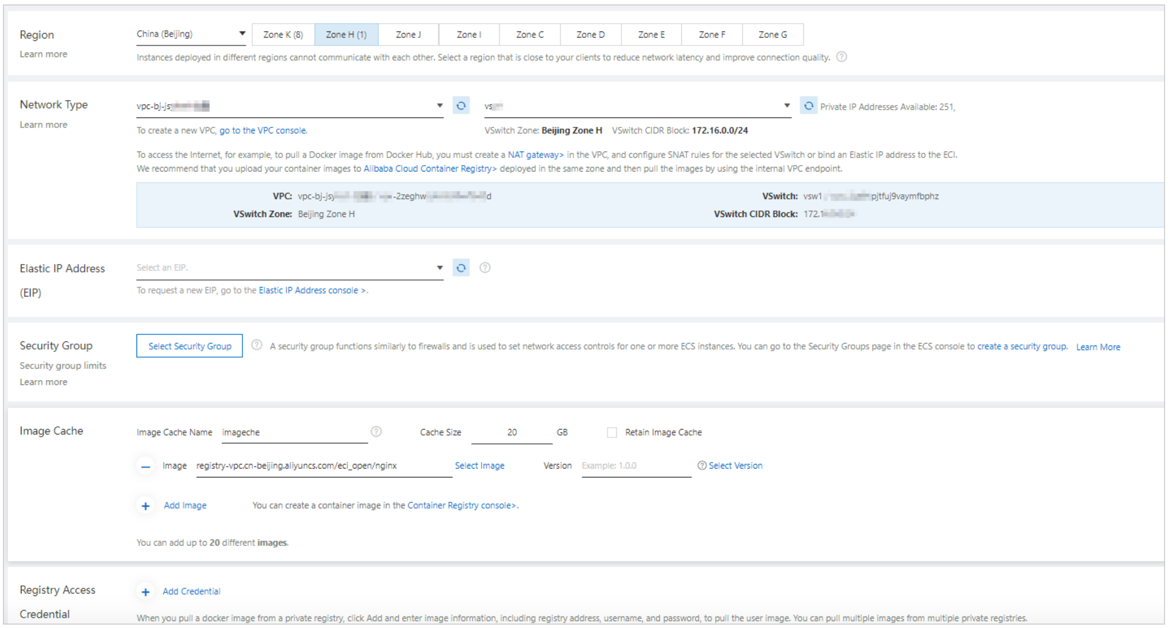
Parameter	Type	Example	Description
AcrRegistryInfo.N.RegionId	String	cn-beijing	The region ID of Container Registry Enterprise Edition instance N.

The following example shows how to use an SDK for Python and the ACR Enterprise Edition to pull images without the need to enter the password when an image cache is created.

```
from aliyunsdkcore.client import AcsClient
from aliyunsdkeci.request.v20180808.CreateImageCacheRequest import CreateImageCacheRequest

client = AcsClient('', '', 'cn-beijing')

request = CreateImageCacheRequest()
request.set_accept_format('json')

request.set_SecurityGroupId("sg-2zeh4cev9y7ulbr****")
request.set_VSwitchId("vsw-2zejlv7xjnw61w6zr****")
request.set_ImageCacheName("acr-test-cache")
request.set_Images([
    "****-registry-vpc.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0",
])
request.set_AcrRegistryInfos([
  {
    "InstanceId": "cri-nwj395hgf6f*****",
    "RegionId": "cn-beijing"
  }
])

response = client.do_action_with_exception(request)
print(str(response, encoding='utf-8'))
```

Create an image cache by using the Elastic Container Instance console

1. In the left-side navigation pane of the Elastic Container Instance console, click Image cache. On the **Image cache** page, click **Create image cache**.
2. On the page that appears, configure parameters.



Region
Learn more

China (Beijing) ▾ | Zone K (8) | Zone H (1) | Zone J | Zone I | Zone C | Zone D | Zone E | Zone F | Zone G

Instances deployed in different regions cannot communicate with each other. Select a region that is close to your clients to reduce network latency and improve connection quality.

Network Type
Learn more

vpc-bj-js[redacted] ▾ vs[redacted] ▾ Private IP Addresses Available: 251,

To create a new VPC, go to the VPC console. VSwitch Zone: **Beijing Zone H** VSwitch CIDR Block: **172.16.0.0/24**

To access the Internet, for example, to pull a Docker image from Docker Hub, you must create a NAT gateway> in the VPC, and configure SNAT rules for the selected VSwitch or bind an Elastic IP address to the ECI.
We recommend that you upload your container images to Alibaba Cloud Container Registry> deployed in the same zone and then pull the images by using the internal VPC endpoint.

VPC: vpc-bj-jsy[redacted]-2zeghw[redacted]d **VSwitch:** vsw1[redacted]pjtfuj9vaymfbphz
VSwitch Zone: Beijing Zone H **VSwitch CIDR Block:** 172.1[redacted]

Elastic IP Address (EIP)

Select an EIP. ▾

To request a new EIP, go to the Elastic IP Address console >.

Security Group
Security group limits
Learn more

Select Security Group

A security group functions similarly to firewalls and is used to set network access controls for one or more ECS instances. You can go to the Security Groups page in the ECS console to create a security group. Learn More

Image Cache

Image Cache Name imageche Cache Size 20 GB □ Retain Image Cache

Image registry-vpc.cn-beijing.aliyuncs.com/eci_open/nginx Select Image Version Example: 1.0.0 Select Version

Add Image You can create a container image in the Container Registry console>.

You can add up to **20** different **images**.

Registry Access Credential

Add Credential

When you pull a docker image from a private registry, click Add and enter image information, including registry address, username, and password, to pull the user image. You can pull multiple images from multiple private registries.

3. Read and select Elastic Container Instance (ECI) Service Agreement. Click **Create** and follow the on-screen instructions to complete the subsequent operations.

4. Go back to the **Image cache** page to view the new image cache.

   The state of the image cache is displayed in the Status column. When the state changes to **Ready**, the image cache is created. Click the ID of the image cache to go to the details page. View the basic information and events about the image cache.

Query an image cache

After you create an image cache, you can query the information about the image cache. You can use the image cache if the image cache is in the Ready state.

You can use one of the following methods to query an image cache:

- Call the DescribeImageCaches API operation. For more information, see DescribeImageCaches.
- In the Elastic Container Instance console, go to the **Image cache** page to view the information about an image cache such as its name and status. You can also click the ID of an image cache to view details such as events about the image cache.

Update an image cache

If an image cache is in the Ready or UpdateFailed state, you can call the UpdateImageCache API operation to update the image cache, such as updating the container image, retention period, and image repository of the image cache. For more information, see UpdateImageCache.

Deletes an image cache

A snapshot is created for each image cache. To retain an image cache, you are charged for the snapshot. If you do not specify a retention period for an image cache, we recommend that you delete the image cache when it is no longer needed.

Note

Automatic image cache snapshots are managed by Alibaba Cloud. The system automatically deletes automatic image cache snapshots that are not used within 3 days after they are created.

You can use one of the following methods to delete an image cache:

- Call the DeleteImageCache API operation. For more information, see DeleteImageCache.
- In the Elastic Container Instance console, go to the **Image cache** page, select the image cache that you want to delete, and click **Delete** in the Action column.

3.Use the image cache feature to accelerate the creation of an elastic container instance

When you create an elastic container instance, you can use the image cache feature to accelerate image pulls and reduce the startup time of the elastic container instance. This topic describes how to use the image cache feature to accelerate the creation of an elastic container instance.

Background information

You can choose one of the following modes to use the image cache feature when you create an elastic container instance:

- Enable automatic match of image caches

  The system automatically matches the most suitable image cache based on the match policy to create the elastic container instance. If no image cache is matched, the system automatically creates an image cache when the system creates the elastic container instance.

- Specify an image cache that is used to create the instance

  You can specify an image cache that is used to create the elastic container instance. The image cache must be in the Ready state.

For more information, see Overview of the image cache feature.

Enable automatic match of image caches

Create an elastic container instance by calling an API operation in OpenAPI Explorer

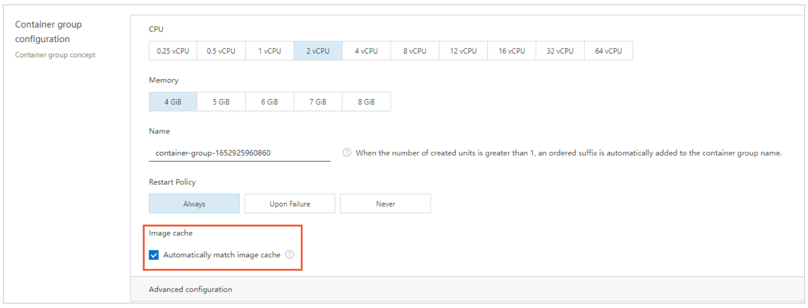
When you call the CreateContainerGroup API operation in OpenAPI Explorer to create an elastic container instance, you can use the AutoMatchImageCache parameter to enable automatic match of image caches. The following table describes the parameter. For more information, see CreateContainerGroup.

Parameter	Type	Example	Description
AutoMatchImageCache	Boolean	true	Specifies whether to enable automatic match of image caches.

Create an elastic container instance by using the Elastic Container Instance console

1. In the left-side navigation pane of the Elastic Container Instance console, click Container Group. On the **Container Group** page, click **Create Container Group**.
2. In the "Basic configuration" step of the instance creation wizard, configure parameters.

In the Container Group Configuration section, you can select **Automatically Match Image Cache**. Then, the system automatically matches the most suitable image cache and uses it to create the elastic container instance. If no image cache is matched, the system automatically creates an image cache.



Container group configuration

Container group concept

Name: container-group-1652925960860

When the number of created units is greater than 1, an ordered suffix is automatically added to the container group name.

Image cache

☑ Automatically match image cache

Advanced configuration

3. Click **Configuration confirmation** and complete the subsequent operations as prompted.

Specify an image cache

If you call the CreateContainerGroup API operation to create an elastic container instance, you can use the ImageSnapshotId parameter to specify an image cache. The following table describes the parameter. For more information, see CreateContainerGroup.

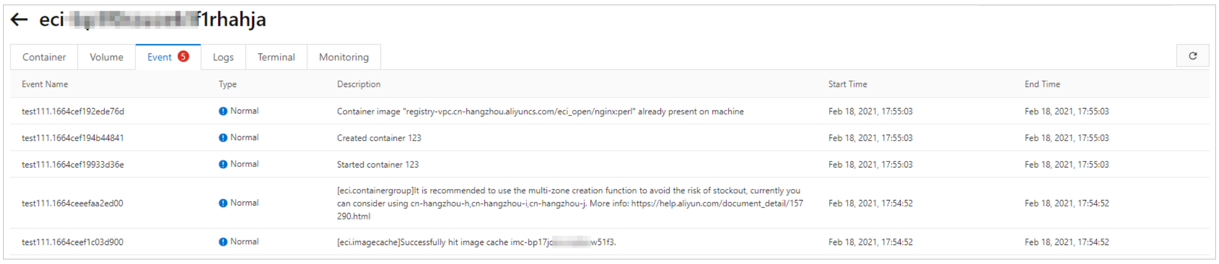
Note

You can specify an image cache only if you call the CreateContainerGroup API operation in OpenAPI Explorer to create an elastic container instance. You cannot specify an image cache if you create an instance by using the Elastic Container Instance console.

Parameter	Type	Example	Description
ImageSnapshotId	String	imc-2zebxkiifuyzzlhl****	The ID of the image cache.

View the result

In the Elastic Container Instance console, click Container Group in the left-side navigation pane to go to the **Container Group** page. Find the elastic container instance that you created by using an image cache and click the instance ID to go to the details page. On the **Events** tab, you can find that an image cache was matched when the elastic container instance was created. The system nearly instantaneously pulled the image cache.



← eci[redacted]f1rhahja

Event Name	Type	Description	Start Time	End Time
test111.1664cef192ede76d	Normal	Container image "registry-vpc.cn-hangzhou.aliyuncs.com/eci_open/nginx:perl" already present on machine	Feb 18, 2021, 17:55:03	Feb 18, 2021, 17:55:03
test111.1664cef194b44841	Normal	Created container 123	Feb 18, 2021, 17:55:03	Feb 18, 2021, 17:55:03
test111.1664cef19933d36e	Normal	Started container 123	Feb 18, 2021, 17:55:03	Feb 18, 2021, 17:55:03
test111.1664ceeefaa2ed00	Normal	[eci.containergroup]It is recommended to use the multi-zone creation function to avoid the risk of stockout, currently you can consider using cn-hangzhou-h,cn-hangzhou-i,cn-hangzhou-j. More info: https://help.aliyun.com/document_detail/157290.html	Feb 18, 2021, 17:54:52	Feb 18, 2021, 17:54:52
test111.1664ceef1c03d900	Normal	[eci.imagecache]Successfully hit image cache imc-bp17jc[redacted]w51f3.	Feb 18, 2021, 17:54:52	Feb 18, 2021, 17:54:52

4.Create and use an image cache (K8s)

4.1. Manage ImageCache

Alibaba Cloud provides the image cache feature of Elastic Container Instance in the manner of ImageCache CustomResourceDefinition (CRD). This way, you can use the image cache feature of Elastic Container Instance to accelerate the creation of pods in Kubernetes. This topic describes the ImageCache resource and how to create, query, and delete ImageCache.

Preparations

Make sure that your Kubernetes cluster supports ImageCache. To check whether a Kubernetes cluster supports ImageCache, log on to the cluster and run the following command:

```
kubectl get crd/imagecaches.eci.alibabacloud.com
```

- If an error is returned, the cluster does not support ImageCache. Sample output:

  ```
  Error from server (NotFound): customresourcedefinitions.apiextensions.k8s.io "imagecaches
  .eci.alibabacloud.com" not found
  ```

  If your cluster does not support ImageCache, update Virtual Kubelet to a version that supports ImageCache.

  > **Note**
  >
  > We recommend that you update Virtual Kubelet to the latest version to use new features. For more information about Virtual Kubelet versions, see ack-virtual-node.

- If imagecaches.eci.alibabacloud.com is returned, the cluster supports ImageCache. Sample output:

  ```
  NAME                                                CREATED AT
  imagecaches.eci.alibabacloud.com                    2019-09-27T01:15:07Z
  ```

  If the ImageCache CRD in your cluster is not of the latest version, we recommend that you run the kubectl apply command to update the ImageCache CRD to the latest version.

  ```
  kubectl apply -f imagecache-crd-sample.yaml
  ```

  Sample imagecache-crd-sample.yaml file:

```
apiVersion: apiextensions.k8s.io/v1beta1
kind: CustomResourceDefinition
metadata:
  name: imagecaches.eci.alibabacloud.com
spec:
  group: eci.alibabacloud.com
  version: v1
  names:
    kind: ImageCache
    plural: imagecaches
    shortNames:
    - ic
    categories:
    - all
  scope: Cluster
  subresources:
    status: {}
  validation:
    openAPIV3Schema:
      required:
      - spec
      properties:
        spec:
          type: object
          required:
          - images
          properties:
            imagePullSecrets:
              type: array
              items:
                type: string
            images:
              minItems: 1
              type: array
              items:
                type: string
            imageCacheSize:
              type: integer
            retentionDays:
              type: integer
  additionalPrinterColumns:
  - name: Age
    type: date
    JSONPath: .metadata.creationTimestamp
  - name: CacheId
    type: string
    JSONPath: .status.imageCacheId
  - name: Phase
    type: string
    JSONPath: .status.phase
  - name: Progress
    type: string
    JSONPath: .status.progress
```

Configure ImageCache

After you create the ImageCache CRD, you can use it to manage ImageCache. For more information, see API operations of Kubernetes ImageCache.

Sample YAML configuration file of ImageCache:

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/imc-enable-reuse: "true" # Enables image cache reuse
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

The following table describes the parameters.

Parameter	Type	Required	Description
spec.images	String[]	Yes	The container images that are used to create the ImageCache.
spec.imagePullSecrets	String[]	No	The secrets of private image repositories. - If an image in spec.images is specified from a private image repository, you must configure a secret for the private image repository in the `namespace:secretName` format. - If all images in spec.images are public images, you do not need to specify this parameter.
spec.imageCacheSize	int	No	The size of the ImageCache. Unit: GB. Default value: 20. Valid values: 20 to 32768.

Parameter	Type	Required	Description
spec.retentionDays	int	No	The retention period of the ImageCache. When the specified retention period ends, the image cache expires and is deleted. Unit: days. This parameter is left empty by default, which indicates that the ImageCache never expires.

ⓘ Note

You can add annotations to the YAML configuration file of the ImageCache to make full use of the image cache feature. For example, in the preceding YAML file, the `k8s.aliyun.com/imc-enable-reuse: "true"` annotation is added to enable image cache reuse. For more information, see Annotations supported by ImageCache.

Create an ImageCache

1. Create an ImageCache.

```
kubectl create -f imagecache-secrets-test.yaml
```

   Sample imagecache-secrets-test.yaml file:

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/imc-enable-reuse: "true" # Enables image cache reuse
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

2. View the status of the ImageCache.

```
kubectl get imagecache imagecache-sample-test
```

   Sample output:

```
NAME                                  AGE    CACHEID                                   PHASE    P
ROGRESS
imagecache-sample-test                20h    imc-2zeditzeoemfhqor****                  Ready
100%
```

Query ImageCache

You can query details about an ImageCache or the ImageCache list in a cluster.

- Query details about the ImageCache list in a cluster

  ```
  kubectl get imagecache
  ```

- Query details about an ImageCache

  ```
  kubectl get imagecache/imagecache-sample-test -o yaml
  ```

Delete an ImageCache

Alibaba Cloud automatically creates a snapshot for each ImageCache. You are charged for the snapshot for each retained ImageCache. We recommend that you manually delete the ImageCache that is no longer needed in a timely manner if you did not configure the retention period when you created the ImageCache.

To delete an ImageCache, run the following command:

```
kubectl delete imagecache/imagecache-sample-test
```

4.2. Use ImageCache to accelerate the creation of pods

Alibaba Cloud provides the image cache feature of Elastic Container Instance in the manner of ImageCache CustomResourceDefinition (CRD). This way, you can use the image cache feature of Elastic Container Instance to accelerate the creation of pods in Kubernetes. This topic describes how to use ImageCache to accelerate the creation of pods.

Description

ImageCache is a cluster-level resource. You can use ImageCache to accelerate the creation of pods in different namespaces in a cluster.

When you create a pod based on ImageCache, you can add annotations to the metadata in the pod configuration file to enable automatic match of ImageCache or specify the ImageCache that you want to use. Take note of the following annotations:

- k8s.aliyun.com/eci-image-cache: Enables automatic match of ImageCache and select the most suitable ImageCache based on the match policy to create the pod. If no ImageCache is matched, the system automatically creates an ImageCache when the system creates the pod.
- k8s.aliyun.com/eci-image-snapshot-id: specifies an ImageCache to create the pod.

Note

If both `k8s.aliyun.com/eci-image-cache` and `k8s.aliyun.com/eci-image-snapshot-id` are configured, k8s.aliyun.com/eci-image-snapshot-id is used. For more information, see Overview of the image cache feature.

When you create a pod by using ImageCache, we recommend that you take the following measures:

- Specify an image in the ImageCache for the container to improve the match degree.
- Set the image pulling policy (ImagePullPolicy) of the container to IfNotPresent to prevent repeated downloads of image layers.

Automatically match ImageCache

When you create a pod, you can add an annotation to enable automatic match of ImageCache to accelerate the creation of the pod. Then, Elastic Container Instance selects the most suitable ImageCache from your ImageCache list based on the match policies. ImageCache is matched based on the following elements in descending order of priority: the match degree of the ImageCache, the size of the ImageCache, and the point in time when the ImageCache is created.

Note

If no ImageCache is matched, the system automatically creates an ImageCache when the system creates the pod.

- Sample Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
      annotations:
        k8s.aliyun.com/eci-image-cache: "true"   # Enables automatic match of ImageCache.
    spec:
      nodeName: virtual-kubelet
      containers:
      - name: nginx
        image: nginx:1.7.9
        imagePullPolicy: IfNotPresent
```

- Sample pod

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    k8s.aliyun.com/eci-image-cache: "true"  # Enables automatic match of ImageCache.
  name: nginx-auto-match
spec:
  containers:
  - image: nginx:1.7.9
    imagePullPolicy: IfNotPresent
    name: nginx
    resources:
      limits:
        cpu: 300m
        memory: 200Mi
      requests:
        cpu: 200m
        memory: 100Mi
  nodeName: virtual-kubelet
```

Specify the ImageCache that is used to create the pod

When you create a pod, you can add an annotation to specify the ImageCache that is used to accelerate the creation of the pod.

> **Notice**
>
> Make sure that the specified ImageCache is in the Ready state. If the ImageCache is in another state, you cannot create the pod.

- Sample Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
      annotations:
        k8s.aliyun.com/eci-image-snapshot-id: imc-2ze5tm5gehgtiiga****  # Specifies an ImageCache.
    spec:
      nodeName: virtual-kubelet
      containers:
      - name: nginx
        image: nginx:1.7.9
        imagePullPolicy: IfNotPresent
```

- Sample pod

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    k8s.aliyun.com/eci-image-snapshot-id: imc-2ze5tm5gehgtiiga****  # Specifies an ImageCache.
  name: nginx-imagecache-id
spec:
  containers:
  - image: nginx:1.7.9
    imagePullPolicy: IfNotPresent
    name: nginx
    resources:
      limits:
        cpu: 300m
        memory: 200Mi
      requests:
        cpu: 200m
        memory: 100Mi
  nodeName: virtual-kubelet
```

4.3. Annotations supported by ImageCaches

If you want to accelerate the creation of pods in a Kubernetes cluster, you can use the image cache feature in the manner of ImageCache CustomResourceDefinition (CRD). When you create an ImageCache, you can add annotations based on your business requirements. This topic describes the annotations that are supported by ImageCaches. This topic also provides examples on how to configure the annotations.

Annotations supported by ImageCaches

The following table describes the annotations that are supported by ImageCaches.

Annotation	Example	Description
k8s.aliyun.com/vswitch-ids	"vsw-bp1dktddjsg5nktv****,vsw-bp1xpiowfm5vo8o3c****"	Specifies the IDs of vSwitches. You can specify multiple vSwitches for multiple zones. If you do not specify this parameter, the ID of the vSwitch configured in eci-profile is used.
k8s.aliyun.com/security-group-id	"sg-bp1dktddjsg5nktv****"	Specifies the ID of the security group. If you do not specify this parameter, the ID of the security group configured in eci-profile is used.
k8s.aliyun.com/resource-group-id	"rg-aek2z3elfs4****"	Specifies the ID of the resource group. If you do not specify this parameter, the ID of the resource group configured in the eci-profile is used.
k8s.aliyun.com/acr-instance-ids	"cri-j36zhodptmyq****"	Specifies the IDs of Alibaba Cloud Container Registry Enterprise Edition instances. These instances are used to pull images without the need to use passwords. You can specify Container Registry Enterprise Edition instances that reside in regions different from the region of the pod. To do this, you must prefix the region ID of the Container Registry Enterprise Edition instance to the ID of the Container Registry Enterprise Edition instance. Example: `"cn-beijng:cri-j36zhodptmyq****"` .

Annotation	Example	Description
k8s.aliyun.com/plain-http-registry	"harbor***.pre.com,192.168.XX.XX:5000,reg***.test.com:80"	Specifies the domain name or IP address of the self-managed image repository. When you create an elastic container instance by using an image in a self-managed image repository that uses the HTTP protocol, you must add this annotation. This way, Elastic Container Instance uses the HTTPS protocol to pull the image. This can prevent the image from failing to pull due to different protocols.
k8s.aliyun.com/insecure-registry	"harbor***.pre.com,192.168.XX.XX:5000,reg***.test.com:80"	Specifies the domain name or IP address of the self-managed image repository. When you create an elastic container instance by using an image in a self-managed image repository that uses a self-signed certificate, you must add this annotation to skip the certificate authentication. This can prevent the image from failing to pull due to certificate authentication failures.
k8s.aliyun.com/imc-enable-reuse	"true"	Specifies whether to enable the image cache reuse feature. If you enable this feature and the image cache that you want to create and an existing image cache contain duplicate image layers, the system reuses the duplicate image layers to create the new image cache. This accelerates the creation of the image cache.
k8s.aliyun.com/imc-enable-flash	"true"	Specifies whether to enable the instant image cache feature. If you enable this feature, the system creates a temporary local snapshot. This reduces the time required to create the image cache. You are charged for the instant availability of the local snapshot.

Annotation	Example	Description
k8s.aliyun.com/imc-retention-days	"7"	Specifies the retention period of the image cache. Unit: days. Expired image caches are automatically cleared. This annotation is left empty by default, which indicates that the ImageCache never expires.
k8s.aliyun.com/imc-size	"25"	Specifies the size of the image cache. Unit: GiB. Default value: 20. Valid values: 20 to 32768.
k8s.aliyun.com/eip-instance-id	"eip-bp1q5n8cq4p7f6dzu****"	Specifies the ID of the elastic IP address (EIP) that is associated with the elastic container instance. If no NAT gateway is configured, you can associate an existing EIP with the elastic container instance to pull images over the Internet.
k8s.aliyun.com/auto-create-eip	"true"	Specifies whether to automatically create an EIP and associate the EIP with the elastic container instance. If no NAT gateway is configured, you can set this annotation to true. This allows the system to create an EIP and associate the EIP with the elastic container instance to pull images over the Internet.
k8s.aliyun.com/eip-bandwidth	"5"	Specifies the bandwidth limit for the EIP when the system creates an EIP.
k8s.aliyun.com/eip-internet-charge-type	PayByBandwidth	Specifies the billing method for network usage of the EIP when the system creates an EIP. Valid values: ● PayByBandwidth ● PayByTraffic

Annotation	Example	Description
k8s.aliyun.com/eip-isp	BGP	Specifies the line type of the EIP when the system creates an EIP. This annotation is suitable only for pay-as-you-go EIPs. Valid values: ● BPG: BGP (Multi-ISP) line ● BGP_PRO: BGP (Multi-ISP) Pro line
k8s.aliyun.com/eip-common-bandwidth-package-id	"cbwp-2zeukbj916scmj51m****"	Specifies the ID of the EIP bandwidth plan.

Create an ImageCache by using the multi-zone feature

When you create an ImageCache, the system creates a temporary elastic container instance. To ensure the ImageCache can be created, you can specify multiple vSwitches that reside in different zones.

> **Note**
>
> If you do not specify vSwitches, the vSwitch configured in eci-profile is used. If multiple zones are configured in eci-profile, you do not need to specify vSwitches.

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/vswitch-ids: "vsw-bp1dktddjsg5nktv****,vsw-bp1xpiowfm5vo8o3c****" # Spec
ifies multiple vSwitches.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

Configure a security group and resource group

By default, the security group and resource group that are configured in eci-profile are used when you create an ImageCache. You can configure a security group and resource group based on your business requirements.

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/security-group-id: "sg-bp1dktddjsg5nktv****" # Specifies a security grou
p
    k8s.aliyun.com/resource-group-id: "rg-aek2z3elfs4****"  # Specifies a resource group.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

Pull Container Registry images without the need to use passwords

Container Registry is a secure platform that allows you to efficiently manage and distribute cloud-native artifacts such as container images and Helm charts that meet the standards of Open Container Initiative (OCI). Container Registry integrates with Alibaba Cloud Container Service for Kubernetes (ACK) to provide end-to-end acceleration capabilities, including acceleration of distribution of large images at scale and image building based on multiple code sources. For more information about Container Registry, see What is Container Registry?

When you create an ImageCache, you can configure a Container Registry Enterprise Edition instance to pull images from the instance without the need to use passwords.

> **Note**
>
> Container Registry Enterprise Edition instances can be used across regions. Therefore, you can specify a Container Registry Enterprise Edition instance that resides in a region different from the region of the ImageCache. To do this, you must prefix the region ID of the Container Registry Enterprise Edition instance to the ID of the Container Registry Enterprise Edition instance. Example: `k8s.aliyun.com/acr-instance-ids: "cn-beijing:cri-j36zhodptmyq****"` .

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/acr-instance-ids: "cri-j36zhodptmyq****" # Specifies the ID of the Conta
iner Registry Enterprise Edition instance.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

Use self-managed image repositories

If a self-managed image repository uses the HTTP protocol or a self-signed certificate when you use an image in the image repository to create an ImageCache, you must configure annotations to prevent the image from failing to pull.

- The self-managed image repository uses the HTTP protocol.

  By default, elastic container instances pull images over the HTTPS protocol. When you create an ImageCache, you must add the `k8s.aliyun.com/plain-http-registry` annotation. This allows the elastic container instance to interact with the image repository over the HTTP protocol.

  ```
  apiVersion: eci.alibabacloud.com/v1
  kind: ImageCache
  metadata:
    name: imagecache-sample
    annotations:
      k8s.aliyun.com/plain-http-registry: "192.168.XX.XX:5000"  # Specifies the domain name
  or IP address of the self-managed image repository to pull an image over the HTTP protoco
  l.
  spec:
    images:
    - 192.168.XX.XX:5000/test/nginx:latest

    imagePullSecrets:
    - default:secret1
    - default:secret2
    - kube-system:secret3
    imageCacheSize:
     25
    retentionDays:
     7
  ```

- The self-managed image repository uses a self-signed certificate.

  If the self-managed image repository uses a self-issued certificate, the certificate authentication fails when the system attempts to pull an image from the image repository. You must add the `k8s.aliyun.com/insecure-registry` annotation to skip certificate authentication.

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/insecure-registry: "harbor***.pre.com"  # Specifies the domain name or
IP address of the self-managed image repository to skip certificate authentication when t
he system pulls an image from the image repository.
spec:
  images:
  - harbor***.pre.com/test/nginx:latest

  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

Note

- If you want to pull images of multiple containers from different image repositories, you can specify multiple domain names and IP addresses of image repositories. Separate multiple domain names and IP addresses with commas (,). Example: `harbor***.pre.com,192.168.XX.XX` .
- If the domain name or IP address of the image repository contains a port number, you must specify the domain name or IP address with its port number. For example, if the IP address of the image repository is `192.168.XX.XX:5000/nginx:latest` , set the value of the annotation to `192.168.XX.XX:5000` .

Enable the image cache reuse feature

The time required to create an ImageCache depends on factors such as the image size and network conditions. You can enable the image cache reuse feature to reduce the time required to create an ImageCache.

After you enable the image cache reuse feature, the system matches existing ImageCaches when you create an ImageCache. If an existing ImageCache contains an image layer that you want to use, the system reuses the image layer to reduce the time required to create the ImageCache.

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/imc-enable-reuse: "true" # Enables the image cache reuse feature.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

Enable the instant image cache feature

The time required to create an ImageCache depends on factors such as the image size and network conditions. You can enable the instant image cache feature to reduce the time required to create an ImageCache.

After you enable the instant image cache feature, the system creates a temporary local snapshot for you to create the ImageCache. After the local snapshot is created, the ImageCache is available for use.

Note

- You are charged based on the size of the local snapshot and the number of times that the local snapshot is used. The fee for a local snapshot is calculated by using the following formula: The fee = Unit price per time × 1 time + Unit price of an instant snapshot × Snapshot size × Usage duration.
- After the local snapshot is created, the system begins to create a regular snapshot. After the regular snapshot is created, the system deletes the local snapshot.

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/imc-enable-flash: "true" # Enables the instant image cache feature.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

Configure the size and retention period of the ImageCache

By default, an ImageCache is 20 GiB in size and is permanently retained after it is created. If you want to customize the size and retention period of the ImageCache, you can specify the imageCacheSize and retentionDays parameters in the spec section, or add annotations.

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/imc-enable-reuse: "7" # Specifies the retention period of the image cach
e.
    k8s.aliyun.com/imc-size: "25" # Specifies the size of the image cache.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
```

Configure an EIP to pull images over the Internet

EIPs are public IP addresses that you can purchase and use as independent resources. When an EIP is associated with a cloud resource, the cloud resource can use the EIP to connect to the Internet. For more information about EIPs, see What is an EIP?

When you create an ImageCache, make sure that you are connected to the Internet before you pull images over the Internet. You can be connected to the Internet by configuring a NAT gateway or configuring an EIP. If your virtual private cloud (VPC) is not associated with a NAT gateway, you can configure an EIP to pull images over the Internet. You can use one of the following methods to configure an EIP.

ⓘ **Note**

When you create an ImageCache, the system creates a temporary elastic container instance and associates the EIP with the elastic container instance. If you use an existing EIP, the EIP is retained after the ImageCache is created. If the system creates an EIP and associates the EIP with the temporary elastic container instance, the EIP and the temporary elastic container instance are released after the ImageCache is created.

- Associate an existing EIP with the temporary elastic container instance

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/eip-instance-id: "eip-bp1q5n8cq4p7f6dzu****" # Associates an existing
EIP with the temporary elastic container instance.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

- Automatically create an EIP and associate the EIP with the temporary elastic container instance

```
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-sample
  annotations:
    k8s.aliyun.com/auto-create-eip: "true" # Automatically creates an EIP and associates
the EIP with the temporary elastic container instance.
    k8s.aliyun.com/eip-bandwidth: "10" # Specifies the bandwidth limit of the EIP.
    k8s.aliyun.com/eip-internet-charge-type: PayByBandwidth # Specifies the billing metho
d for network usage of the EIP.
    k8s.aliyun.com/eip-common-bandwidth-package-id: "cbwp-2zeukbj916scmj51m****" # Specif
ies the ID of the EIP bandwidth plan.
spec:
  images:
  - centos:latest
  - busybox:latest
  imagePullSecrets:
  - default:secret1
  - default:secret2
  - kube-system:secret3
  imageCacheSize:
   25
  retentionDays:
   7
```

4.4. kubernetes ImageCache API

This topic describes Kubernetes ImageCache APIs.

Data types

ImageCacheList

Parameter	Type	Description
apiVersion	String	The version of the API. Set the value to eci.alibabacloud.com/v1.
kind	String	The resource type. Set the value to ImageCacheList.
metadata	ListMeta	The standard list metadata.
Items	ImageCache[]	The image caches.

ImageCache

Parameter	Type	Description
apiVersion	String	The version of the API. Set the value to eci.alibabacloud.com/v1.
kind	String	The resource type. Set the value to ImageCache.
metadata	ObjectMeta	The metadata of the object.
spec	ImageCacheSpec	The details of the image cache.
status	mageCacheStatus	The status of the image cache.

ImageCacheSpec

Parameter	Type	Required	Description
images	String[]	Yes	The images in the image cache.
imagePullSecrets	String[]	No	The secrets of the private image repositories. Specify each secret in the namespace:secretName format.
imageCacheSize	Int	No	The size of the image cache. Unit: GB. Value range: 20 to 32768. Default value: 20.
retentionDays	Int	No	The retention period of the image cache in days. The parameter value ranges from 1 to 65536. By default, the parameter is empty, which indicates that the image cache does not expire.

ImageCacheStatus

Parameter	Type	Description
phase	String	The status of the image cache.

Parameter	Type	Description
progress	String	The creation progress of the image cache.
imageCacheId	String	The ID of the image cache.
startTime	Time	The time when the image cache was created.
lastUpdatedTime	Time	The time when the image cache was last updated.
events	Event[]	The details of the events that occurred when the image cache was created.
expireDateTime	Time	The time when the image cache expires. An image cache may expire at one of the following points in time: • The time when the retention period that is specified during the creation of the image cache ends. • 24 hours after the image cache fails to be created.

CreateImageCache

Creates an image cache.

HTTP request

```
POST /apis/eci.alibabacloud.com/v1/imagecaches
```

curl request example

```
curl -X POST -H 'Content-Type: application/yaml' --data '
apiVersion: eci.alibabacloud.com/v1
kind: ImageCache
metadata:
  name: imagecache-secrets-test
spec:
  images:
  - registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0
  imagePullSecrets:
  - default:acr-test
' http://127.0.0.1:8001/apis/eci.alibabacloud.com/v1/imagecaches
```

curl response example

```
{
  "apiVersion": "eci.alibabacloud.com/v1",
  "kind": "ImageCache",
  "metadata": {
    "creationTimestamp": "2019-10-10T03:50:47Z",
    "generation": 1,
    "name": "imagecache-secrets-test",
    "resourceVersion": "647052328",
    "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test11",
    "uid": "2506ba37-eb11-11e9-9576-f2ef49bf****"
  },
  "spec": {
    "imagePullSecrets": [
      "default:acr-test"
    ],
    "images": [
      "registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0"
    ]
  }
}
```

ListImageCache

Queries image caches.

HTTP request

```
GET /apis/eci.alibabacloud.com/v1/imagecaches
```

curl request example

```
curl -X GET 'http://127.0.0.1:8001/apis/eci.alibabacloud.com/v1/imagecaches'
```

curl response example

```
{
  "apiVersion": "eci.alibabacloud.com/v1",
  "items": [
    {
      "apiVersion": "eci.alibabacloud.com/v1",
      "kind": "ImageCache",
      "metadata": {
        "annotations": {
          "kubectl.kubernetes.io/last-applied-configuration": "{\"apiVersion\":\"eci.alibabacloud.com/v1\",\"kind\":\"ImageCache\",\"metadata\":{\"annotations\":{},\"name\":\"imagecache-secrets-test\"},\"spec\":{\"imagePullSecrets\":[\"default:acr-test\"],\"images\":[\"registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0\"]}}\n"
        },
        "creationTimestamp": "2019-10-10T01:43:48Z",
        "generation": 1,
        "name": "imagecache-secrets-test",
        "resourceVersion": "646375979",
        "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test",
        "uid": "67460461-eaff-11e9-9774-b2c06862****"
      },
      "spec": {
        "imagePullSecrets": [
          "default:acr-test"
        ],
        "images": [
          "registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0"
        ]
      },
      "status": {
        "imageCacheId": "imc-2ze5xnx4yiply807****",
        "lastUpdatedTime": "2019-10-10T01:47:14Z",
        "phase": "Ready",
        "progress": "100%",
        "startTime": "2019-10-10T01:43:49Z"
      }
    }
  ],
  "kind": "ImageCacheList",
  "metadata": {
    "continue": "",
    "resourceVersion": "647062964",
    "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches"
  }
}
```

GetImageCache

Queries information of an image cache.

HTTP request

```
GET /apis/eci.alibabacloud.com/v1/imagecaches/{name}
```

curl request example

```
curl -X GET 'http://127.0.0.1:8001/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test'
```

curl response example

```
{
  "apiVersion": "eci.alibabacloud.com/v1",
  "kind": "ImageCache",
  "metadata": {
    "annotations": {
      "kubectl.kubernetes.io/last-applied-configuration": "{\"apiVersion\":\"eci.alibabacloud.com/v1\",\"kind\":\"ImageCache\",\"metadata\":{\"annotations\":{},\"name\":\"imagecache-secrets-test\"},\"spec\":{\"imagePullSecrets\":[\"default:acr-test\"],\"images\":[\"registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0\"]}}\n"
    },
    "creationTimestamp": "2019-10-10T01:43:48Z",
    "generation": 1,
    "name": "imagecache-secrets-test",
    "resourceVersion": "646375979",
    "selfLink": "/apis/eci.alibabacloud.com/v1/imagecaches/imagecache-secrets-test",
    "uid": "67460461-eaff-11e9-9774-b2c06862****"
  },
  "spec": {
    "imagePullSecrets": [
      "default:acr-test"
    ],
    "images": [
      "registry.cn-shanghai.aliyuncs.com/baz/nginx:1.0"
    ]
  },
  "status": {
    "imageCacheId": "imc-2ze5xnx4yiply807****",
    "lastUpdatedTime": "2019-10-10T01:47:14Z",
    "phase": "Ready",
    "progress": "100%",
    "startTime": "2019-10-10T01:43:49Z"
  }
}
```

DeleteImageCache

Deletes an image cache.

HTTP request

```
DELETE /apis/eci.alibabacloud.com/v1/imagecaches/{name}
```

curl request example

```
curl -X DELETE -H 'Content-Type: application/yaml' 'http://127.0.0.1:8001/apis/eci.alibabpa
cloud.com/v1/imagecaches/imagecache-secrets-test'
```

curl response example

```
  "kind": "Status",
  "apiVersion": "v1",
  "metadata": {},
  "status": "Success",
  "details": {
    "name": "imagecache-secrets-test",
    "group": "eci.alibabacloud.com",
    "kind": "imagecaches",
    "uid": "67460461-eaff-11e9-9774-b2c06862****"
  }
}
```

5.Configure password-free access to pull images from a Container Registry Enterprise Edition instance

To pull images from a Container Registry instance, you can configure password-free access to the instance to simplify configurations and accelerate the pulling of images. This topic describes how to configure password-free access to pull images from a Container Registry Enterprise Edition instance.

Prerequisites

The following requirements are met:

- Container Registry is activated. A RAM role of Elastic Container Instance is authorized to access Container Registry.
- A Container Registry Enterprise Edition instance is created and an image repository is configured. For more information, see Use a Container Registry Enterprise Edition instance to push and pull images.

Background information

Container Registry provides Container Registry Personal Edition instances and Container Registry Enterprise Edition instances. Container Registry Enterprise Edition is an enterprise-grade platform used to manage the lifecycle of cloud native application artifacts. These artifacts include container images, Helm charts, and Open Container Initiative (OCI) artifacts. Container Registry Enterprise Edition can seamlessly integrate with Container Service for Kubernetes (ACK) in large-scale business deployment scenarios to simplify application delivery for enterprises. For more information, see What is Container Registry?

When you pull an image from an image repository of a Container Registry instance, the following situations exist:

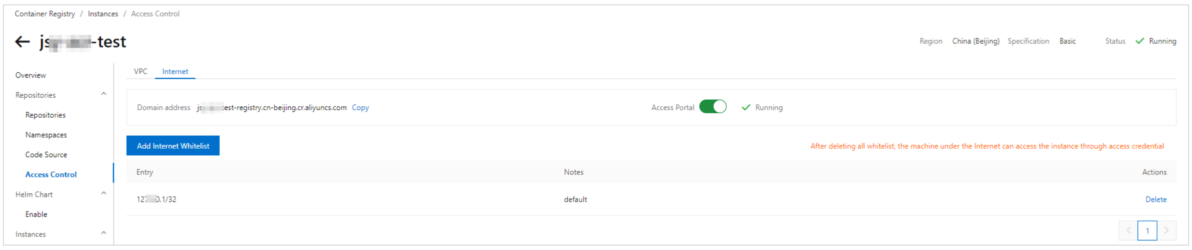
- For images in a Container Registry Personal Edition instance that belongs to the same account as the elastic container instance, you can pull the images without using a password.
- For non-Container Registry images such as Docker images, you cannot pull the images without using a password. When you call an API operation to create an elastic container instance, you can use the ImageRegistryCredential parameter to pass in a password.

Configure password-free access to a Container Registry Enterprise Edition instance

In the Container Registry console, find the instance to which you want to configure password-free access and configure the following network access control settings:

- Settings for access over the Internet

  After you enable Internet access, you can access images in the Container Registry Enterprise Edition instance across regions by using public domain names. For more information, see Configure access over the Internet.



Container Registry / Instances / Access Control

← js[redacted]-test

Region China (Beijing) Specification Basic Status ✓ Running

Domain address js[redacted]test-registry.cn-beijing.cr.aliyuncs.com Copy

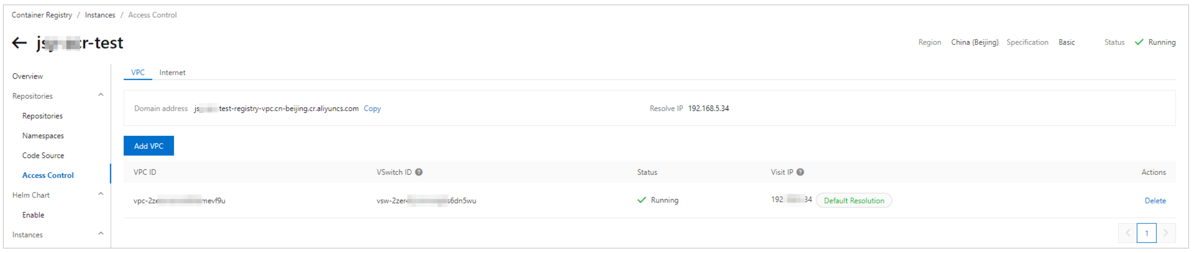
Access Portal ● ✓ Running

After deleting all whitelist, the machine under the Internet can access the instance through access credential

Entry	Notes	Actions
12[redacted]0.1/32	default	Delete

- Settings for access over virtual private clouds (VPCs)

  You must grant required permissions before you can allow access to the Container Registry Enterprise Edition instance over VPCs. For more information, see Configure access over VPCs.



Container Registry / Instances / Access Control

← js[redacted]r-test

Region China (Beijing) Specification Basic Status ✓ Running

Domain address js[redacted]test-registry-vpc.cn-beijing.cr.aliyuncs.com Copy

Resolve IP 192.168.5.34

VPC ID	VSwitch ID	Status	Visit IP	Actions
vpc-2ze[redacted]mevf9u	vsw-2zer[redacted]s6dn5wu	✓ Running	192[redacted]34 Default Resolution	Delete

After you configure the Container Registry Enterprise Edition instance, you can record the instance information such as the instance ID, instance name, and domain name for subsequent use.

Use the Kubernetes method to pull images from Container Registry Enterprise Edition instances without using passwords

You can add annotations to specify a Container Registry Enterprise Edition instance from which you want to pull images.

> **Note**
>
> You can specify only one Container Registry Enterprise Edition instance when you use the Kubernetes method. If you have multiple Container Registry Enterprise Edition instances that contain different images, we recommend that you put the images into a single Container Registry Enterprise Edition instance. If you want to configure multiple Container Registry Enterprise Edition instances, we recommend that you call an API operation.

Example:

1. Prepare the YAML file.

   The following sample code provides an example of test_cri.yaml:

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    k8s.aliyun.com/acr-instance-id: "cri-j36zhodptmyq****"      # Specify the ID of the 
Container Registry Enterprise Edition instance.
  name: cri-test
spec:
  containers:
  - image: test****-registry.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0   # Pull an 
image over the Internet.
    imagePullPolicy: Always
    name: nginx
  restartPolicy: Never
```

> ? Note
>
> Container Registry Enterprise Edition instances can be used across regions. Therefore, you can specify a Container Registry Enterprise Edition instance that resides in a region different from the region of the pod. To do this, you must prefix the region ID of the Container Registry Enterprise Edition to the ID of the Container Registry Enterprise Edition instance. Example: `k8s.aliyun.com/acr-instance-id: "cn-beijing:cri-j36zhodptmyq****"` .

2. Create a pod.

```
kubectl apply -f test_cri.yaml
```

Call an API operation to pull images from Container Registry Enterprise Edition instances without using passwords

When you call the CreateContainerGroup API operation to create an elastic container instance, you can use AcrRegistryInfo-related parameters to configure password-free access. The following table describes the parameters. For more information, see CreateContainerGroup.

> ? Note
>
> When you use AcrRegistryInfo-related parameters to configure password-free access, you must specify the AcrRegistryInfo.N.InstanceId parameter.

Parameter	Type	Example	Description
AcrRegistryInfo.N.RegionId	String	cn-beijing	The region ID of Container Registry Enterprise Edition instance N.
AcrRegistryInfo.N.InstanceId	String	cri-nwj395hgf6f3****	The ID of Container Registry Enterprise Edition instance N.

Parameter	Type	Example	Description
AcrRegistryInfo.N.Domain.N	RepeatList	test****-registry.cn-beijing.cr.aliyuncs.com	Domain name N of Container Registry Enterprise Edition instance N. All domain names of instance N are displayed by default. You can specify domain names. Separate multiple domain names with commas (,).
AcrRegistryInfo.N.InstanceName	String	test****	The name of Container Registry Enterprise Edition instance N.

The following examples demonstrate how to specify AcrRegistryInfo-related parameters:

- Example 1: Specify the region ID, ID, name, and domain names of the Container Registry Enterprise Edition instance.

```
'Container.1.Image': 'test****-registry.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0',
'Container.1.Name': 'c1',
'Container.2.Image': 'test****-registry-vpc.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0
',
'Container.2.Name': 'c2',

#AcrRegistryInfo
'AcrRegistryInfo.1.RegionId':'cn-beijing',
'AcrRegistryInfo.1.InstanceId': 'cri-nwj395hg********',
'AcrRegistryInfo.1.Domain.1': 'test****-registry-vpc.cn-beijing.cr.aliyuncs.com',
'AcrRegistryInfo.1.Domain.2': 'test****-registry.cn-beijing.cr.aliyuncs.com'
```

- Example 2: Specify the ID and name of the Container Registry Enterprise Edition instance.

```
'Container.1.Image': 'test****-registry.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0',
'Container.1.Name': 'c1',
'Container.2.Image': 'test****-registry-vpc.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0
',
'Container.2.Name': 'c2',

#AcrRegistryInfo
'AcrRegistryInfo.1.InstanceId': 'cri-nwj395hg********',
'AcrRegistryInfo.1.InstanceName': 'test****'
```

- Example 3: Specify only the ID of the Container Registry Enterprise Edition instance.

```
'Container.1.Image': 'test****-registry.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0',
'Container.1.Name': 'c1',
'Container.2.Image': 'test****-registry-vpc.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0
',
'Container.2.Name': 'c2',

#AcrRegistryInfo
'AcrRegistryInfo.1.InstanceId': 'cri-nwj395hg********'
```

You can also use SDKs to specify AcrRegistryInfo-related parameters. The following sample code provides an example on how to use SDK for Python to specify AcrRegistryInfo-related parameters.

```
#!/usr/bin/env python
#coding=utf-8

from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.acs_exception.exceptions import ClientException
from aliyunsdkcore.acs_exception.exceptions import ServerException
from aliyunsdkeci.request.v20180808.CreateContainerGroupRequest import CreateContainerGroup
Request

client = AcsClient('<accessKeyId>', '<accessSecret>', 'cn-beijing')

request = CreateContainerGroupRequest()
request.set_accept_format('json')

request.set_SecurityGroupId("sg-2zeh4cev9y7ulbr*****")
request.set_VSwitchId("vsw-2zejlv7xjnw61w6z*****")
request.set_ContainerGroupName("test-cri")
request.set_Containers([
  {
    "Image": "test****-registry.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0",
    "Name": "nginx"
  },
  {
    "Image": "test****-registry-vpc.cn-beijing.cr.aliyuncs.com/eci_test/nginx:1.0",
    "Name": "nginx2"
  }
])
request.set_AcrRegistryInfos([
  {
    "RegionId": "cn-beijing",
    "InstanceId": "cri-nwj395hgf6f*****",
    "Domains": [
      "test****-registry-vpc.cn-beijing.cr.aliyuncs.com",
      "test****-registry.cn-beijing.cr.aliyuncs.com"
    ]
  }
])

response = client.do_action_with_exception(request)
# python2:  print(response)
print(str(response, encoding='utf-8'))
```

6.Use self-managed image repositories

In scenarios in which Kubernetes is involved, when you use an image in a self-managed image repository to create an elastic container instance-based pod, the image may fail to be pulled due to different protocols used by the elastic container instance and the image repository or due to certificate authentication failures. This topic describes how to use an image in a self-managed image repository to create an elastic container instance-based pod in case the self-managed image repository uses the HTTP protocol and a self-signed certificate.

Scenarios

This topic is applicable to the scenarios in which Kubernetes is involved. Make sure that you have connected Elastic Container Instance to Kubernetes before you perform the operations described in this topic. For more information, see the following topics:

- Connect Kubernetes to Elastic Container Instance
- Overview

Description

In scenarios in which Kubernetes is involved, when you use an image in a self-managed image repository to create an elastic container instance-based pod, an alert event named ErrImagePull may be triggered, and the image cannot be pulled.

If the network connectivity between the elastic container instance and the image repository is normal, this issue may occur due to the following causes:

- The self-managed image repository uses the HTTP protocol.

  Elastic container instances use the HTTPS protocol to pull images by default. When you create an elastic container instance-based pod, you must manually configure the Elastic Container Instance to use the HTTP protocol to interact with the image repository.

- The self-managed image repository uses the HTTPS protocol but uses a self-signed certificate.

  If the image repository uses a self-signed certificate, the certificate authentication fails when you pull images. Therefore, when you create an elastic container instance-based pod, you must make configurations to skip the certificate authentication.

Therefore, if a self-managed image repository uses the HTTP protocol or a self-signed certificate when you use an image in the image repository to create an elastic container instance-based pod, you must add annotations to prevent image full failures. The following table describes the annotations:

Annotation	Example	Description

Annotation	Example	Description
k8s.aliyun.com/plain-http-registry	"harbor***.pre.com,192.168.XX.XX:5000,reg***.test.com:80"	The endpoint of the self-managed image repository. When you create an elastic container instance-based pod by using an image in a self-managed image repository over HTTP, you must add this annotation to the instance. This allows Elastic Container Instance to pull the image over HTTP. This prevents image pull failures caused by different protocols.
k8s.aliyun.com/insecure-registry	"harbor***.pre.com,192.168.XX.XX:5000,reg***.test.com:80"	The endpoint of the self-managed image repository. When you create an elastic container instance-based pod by using an image in a self-managed image repository that uses a self-signed certificate, you must add this annotation to the instance to skip the certificate authentication. This prevents image pull failures caused by certificate authentication failures.

Note

- If you want to pull images of multiple containers from different image repositories, you can specify multiple endpoints of image repositories. Separate multiple endpoints with commas (,). Example: `harbor***.pre.com,192.168.XX.XX` .
- If the image repository endpoint contains a port number, you must specify the endpoint with its port number. For example, if the image repository endpoint is `192.168.XX.XX:5000/nginx:latest` , set the annotation to `192.168.XX.XX:5000` .

Examples

When you make configurations, add annotations to the metadata field of the pods. For example, when you configure a Deployment, add annotations to the spec.template.metadata field. The following code provides sample YAML files.

- A self-managed image repository uses the HTTP protocol.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
  labels:
    app: nginx
spec:
  replicas: 4
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      name: nginx-test
      annotations:
        k8s.aliyun.com/plain-http-registry: "192.168.XX.XX:5000"
      labels:
        app: nginx
        alibabacloud.com/eci: "true"
    spec:
      containers:
      - name: nginx
        image: 192.168.XX.XX:5000/test/nginx:latest
```

- A self-managed image repository uses a self-signed certificate.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
  labels:
    app: nginx
spec:
  replicas: 4
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      name: nginx-test
      annotations:
        k8s.aliyun.com/insecure-registry: "harbor***.pre.com"
      labels:
        app: nginx
        alibabacloud.com/eci: "true"
    spec:
      containers:
      - name: nginx
        image: harbor***.pre.com/test/nginx:latest
```