ALIBABA CLOUD

Alibaba Cloud

Apsara Stack Enterprise

Container Registry

User Guide

Product Version: v3.16.2

Document Version: 20220915

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.What is Container Registry?

Container Registry is a platform that allows you to manage and distribute cloud-native artifacts in a secure and efficient manner. Cloud-native artifacts include container images and Helm charts that meet the standards of Open Container Initiative (OCI). Container Registry provides the following features: image permission management, synchronous image distribution, and content signing. The features allow you to manage the entire lifecycle of container images. Container Registry simplifies the setup and O&M of container registries. Container Registry is integrated with Alibaba Cloud services such as Container Service for Kubernetes (ACK) to help enterprises reduce delivery complexity and create a one-stop solution for cloud-native applications.

Classification of Container Registry Enterprise Edition

Container Registry Standard Edition

Container Registry Standard Edition is suitable for small and medium-sized enterprises. It provides secure management and efficient distribution of container images. You can create up to 15 namespaces on a Container Registry Standard Edition instance, maintain 1,000 image repositories, public or private, in each namespace, and concurrently pull images on up to 100 nodes.

Container Registry Advanced Edition

Container Registry Advanced Edition is suitable for medium and large-sized enterprises. It provides secure management and lifecycle management of OCI artifacts, such as container images, Helm charts, and operators. Container Registry Advanced Edition also supports efficient image distribution scenarios, such as multi-region replication and cross-cloud replication. You can create up to 1,000 namespaces, totally maintain 100,000 image repositories, and concurrently pull images on up to 500 nodes.

Limits for a single user

Item	Standard Edition	Advanced Edition
Number of namespaces	15	1,000
Number of image repositories	2,000	100,000
Number of nodes for concurrent pulls	100	500

Features

Category	Sub-category		Description	Standard Edition	Advanced Edition
		Secure management	You can securely manage container images by namespace.	Support	Support
		Lifecycle management	You can query container image tags and delete image repositories.	Support	Support

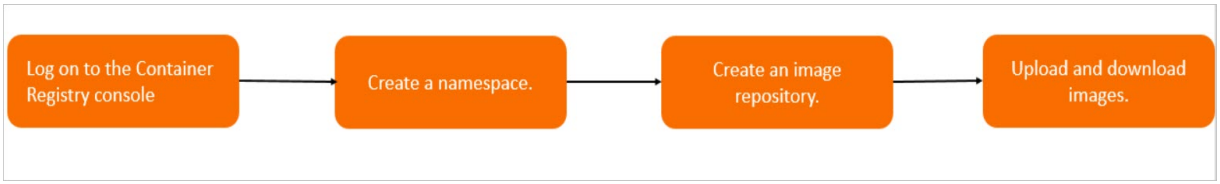
Category	Sub-category		Description	Standard Edition	Advanced Edition
Artifact management	Container image	Fine-grained permission management	• You can manage the permissions on resource sets and departments in the Apsara Uni-manager Management Console. • You can manage user permissions.	Support	Support
		Immutable tags	You can configure image tags to be immutable.	Not support	Support
	Tags of OCI artifacts, such as Helm charts and operators	Secure management	You can securely manage OCI artifacts by namespace.	Not support	Support
		Lifecycle management	You can query artifact tags and delete artifact repositories.	Not support	Support
		Fine-grained permission management	• You can manage the permissions on resource sets and departments in the Apsara Uni-manager Management Console. • You can manage user permissions.	Not support	Support
		Immutable tags	You can configure OCI artifact tags to be immutable.	Not support	Support
Artifact security	Trigger		If a container image is updated, the corresponding event is automatically triggered.	Support	Support
	Manual image replication		You can manually trigger the replication of a container image of a specific tag to implement geo-disaster recovery for container images.	Support	Support
	Automatic image replication		After images are pushed, the images are automatically replicated based on replication rules.	Not support	Support

Category	Sub-category	Description	Standard Edition	Advanced Edition
	Cross-account replication	Images are replicated across Alibaba Cloud accounts.	Not support	Support
Artifact security	Encrypted image distribution	You can configure secure HTTPS protocol to distribute container images.	Support	Support
	Image signing	This feature prevents man-in-the-middle (MITM) attacks and unauthorized image updates or deployments. This ensures image consistency and security from distribution to deployment.	Support	Support
	Image scanning (provided after Container Registry is integrated with Apsara Stack Security Center)	This feature allows you to scan container images to identify vulnerabilities.	Not support	Support
Deployment integration	Image pulls without a secret	You can configure image pulls without a secret in the ACK console. This way, you do not need to specify a secret for each image pull.	Support	Support
	Deployment selection	You can select image repositories and image tags when you configure a Deployment in the ACK console.	Support	Support

2.Getting Started

2.1. Use Container Registry

This topic describes how to use Container Registry.



Standard Edition

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. Create a namespace. For more information, see Create a namespace.
3. Create an image repository. For more information, see Creates an image repository.
4. Upload and download images. For more information, see Upload and download images.

Advanced Edition

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. Create a namespace. For more information, see Create a namespace.
3. Create an image repository. For more information, see Create an image repository.
4. Upload and download images. For more information, see Upload and download images.

2.2. Log on to the Container Registry console

This topic describes how to log on to the Container Registry console.

Prerequisites

- Before you log on to the Apsara Uni-manager Management Console, the endpoint of the console is obtained from the deployment staff.
- We recommend that you use Google Chrome.

Procedure

1. Enter the URL of the Apsara Uni-manager Management Console in the address bar and press the Enter key.
2. Enter your username and password.

   Obtain the username and password that you use to log on to the Apsara Uni-manager Management Console from the operations administrator.

> **Note** The first time that you log on to the Apsara Uni-manager Management Console, you must change the password of your account. For security purposes, your password must meet the minimum complexity requirements. The password must be 10 to 32 characters in length and must contain at least two of the following character types:
>
> - Uppercase or lowercase letters
> - Digits
> - Special characters, including exclamation points (!), at signs (@), number signs (#), dollar signs ($), and percent signs (%)

3. Click **Log On**.
4. If your account has multi-factor authentication (MFA) enabled, perform corresponding operations in the following scenarios:
   - You log on to the Apsara Uni-manager Management Console for the first time after MFA is enabled by the administrator:
     a. On the Bind Virtual MFA Device page, bind an MFA device.
     b. Enter the username and password again as in Step 2 and click **Log On**.
     c. Enter a six-digit MFA verification code and click **Authenticate**.
   - You have enabled MFA and bound an MFA device:

     Enter a six-digit MFA verification code and click **Authenticate**.

     > **Note** For more information, see the *Bind a virtual MFA device to enable MFA* topic in *Apsara Uni-manager Management Console User Guide*.
5. In the top navigation bar, choose **Products > Elastic Computing > Container Registry**.
6. On the **Instances** page, select an instance to perform image-related operations.
   - Click the card of a Container Registry Advanced Edition instance. On the management page of the instance, you can perform operations on the instance. For example, you can create a namespace and upload and download images on the instance.
   - Click the card of a Container Registry Standard Edition instance. On the management page of the instance, you can perform operations on the instance. For example, you can create a namespace and upload and download images on the instance.

3.Container Registry Standard Edition

3.1. Permission management

3.1.1. Create a user and a user group

You can create users and assign different roles to the users to meet different requirements for system access control as an administrator. This topic describes how to create a user and a user group.

Prerequisites

- Before you log on to the Apsara Uni-manager Management Console, the endpoint of the console is obtained from the deployment staff.
- A browser is available. We recommend that you use Google Chrome.

Create an organization

To create a user or a user group, you must create an organization in advance.

1. Enter the URL of the Apsara Uni-manager Management Console in the address bar of your browser and press Enter.
2. Enter your username and password.

   Obtain the username and password that you use to log on to the Apsara Uni-manager Management Console from the operations administrator.

   > **Note** The first time you log on to the Apsara Uni-manager Management Console, you must change the password of your account. For security purposes, your password must meet the minimum complexity requirements. The password must be 10 to 32 characters in length and must contain at least two of the following character types:
   >
   > - Letters
   > - Digits
   > - Special characters, which include ! @ # $ %

3. Click **Log On**.
4. In the top navigation bar, click **Enterprise**.
5. In the left-side navigation pane of the Enterprise page, choose Resources > **Organizations**.
6. In the left-side organization navigation tree, click the name of a parent organization under which you want to create an organization. Click **Create Organization** in the upper-right corner of the page.
7. In the Add Organization dialog box, enter a name for the organization and click **OK**.
8. (Optional) If the multi-cloud management feature is enabled for the primary node, the **Synchronously Create Multi-cloud Organization** check box is displayed after you click **OK**.

   Select **Synchronously Create Multi-cloud Organization**, select clouds on which the organization is synchronously created, and then click **OK**.

Create a user group

A role cannot be directly assigned to a user. You can assign a role to a user group. Then, users in the user group can obtain the permissions of the role.

Relationships between user groups and users:

- A user group can contain zero or more users.
- A user does not necessarily belong to a user group.
- A user can be added to multiple user groups.

Relationships between user groups and organizations:

- A user group belongs to only one organization.
- You can create multiple user groups within an organization.

Relationships between user groups and roles:

- A role can be assigned to multiple user groups.
- When a role is assigned to a user group, the permissions that the role has are automatically granted to the users within the user group.

Relationships between user groups and resource sets:

- A resource set can be added to zero or more user groups.
- A user group can be added to multiple resource sets.

1. In the top navigation bar, click **Enterprise**.
2. In the left-side navigation pane, choose **Users > User Groups**.
3. Click **Create a user group** in the upper-left corner of the page.
4. In the dialog box that appears, configure **User Group Name**, **Organization**, and **Role authorization**.

Parameter	Description
User Group Name	The name of the user group. The name must be 3 to 255 characters in length and can contain letters, digits, underscores (_), hyphens (-), periods (.), and at signs (@).
Organization	The organization to which the user group belongs.
Role authorization	The roles that are assigned to the user group.

5. Click **OK**.

Create a user

You can create users based on your business requirements.

1. In the top navigation bar, click **Enterprise**.
2. In the left-side navigation pane, choose **Users > Users**.
3. Click the **System Users** tab. Click **Create a user**.
4. In the dialog box that appears, configure parameters.

Parameter	Required	Description
User name	Yes	The Apsara Stack tenant account name of the user. The name must be 1 to 64 characters in length and can contain letters, digits, hyphens (-), underscores (_), and periods (.).
Display name	Yes	The display name of the user. The name must be 1 to 128 characters in length and can contain letters, digits, hyphens (-), underscores (_), periods (.), and at signs (@).
Role	Yes	The role to be assigned to the user. **Note** You can enter role names in the field. Fuzzy match is supported.
Organization	Yes	The organization to which the user belongs.
Logon policy	Yes	The logon policy that specifies the logon time of the user and the IP address that the user logs on to. If you do not configure this parameter, the default policy is attached to the created user. **Note** The default policy does not specify the logon time of the user and the IP address that the user logs on to. To specify the logon time and IP address, you can modify the logon policy or create a logon policy.
Phone	Yes	The mobile phone number of the user. If you want to send text messages about the resource requests and usage to the mobile phone number, make sure that the specified mobile phone number is valid. **Note** If the mobile phone number of the user changes, update it on the system in a timely manner.
Landline	No	The landline number of the user. The landline number must be 4 to 20 characters in length and can contain only digits and hyphens (-).
Email	Yes	The email address of the user. If you want to send emails about the resource requests and usage to the email address, make sure that the specified email address is valid. **Note** If the email address changes, update it on the system in a timely manner.

Parameter	Required	Description
DingTalk Key	No	The key of the chatbot for the DingTalk group to which the user belongs. For more information about how to configure the key, see DingTalk development documentation.
Notify User by Email	No	If this option is selected, the Apsara Uni-manager Management Console informs the user configured as the alert contact by email whenever an alert is generated. **Note** You must configure an email server to receive emails. For more information, contact on-site O&M engineers.
Notify User by DingTalk	No	If this option is selected, the Apsara Uni-manager Management Console informs the user configured as the alert contact by DingTalk whenever an alert is generated.

5. Click **Modify**.

3.1.2. Create a role and grant permissions to the role

You can create custom roles, attach permission policies to the roles, and assign the roles to a user to grant permissions to the user. This topic describes how to create a role, assign the role to a user group, and attach a policy to the role.

Context

A role is a set of access permissions. Each role has a range of permissions. A user can have multiple roles, which means that the user is granted all the permissions defined for the roles. A role can be used to grant the same set of permissions to a group of users.

The total number of custom and default roles cannot exceed 20.

Create a role

1. Enter the URL of the Apsara Uni-manager Management Console in the address bar of your browser and press Enter.
2. Enter your username and password.

   Obtain the username and password that you use to log on to the Apsara Uni-manager Management Console from the operations administrator.

> **Note** The first time you log on to the Apsara Uni-manager Management Console, you must change the password of your account. For security purposes, your password must meet the minimum complexity requirements. The password must be 10 to 32 characters in length and must contain at least two of the following character types:
>
> - Letters
> - Digits
> - Special characters: ! @ # $ %

3. Click **Log On**.
4. In the top navigation bar, click **Enterprise**.
5. In the left-side navigation pane, choose **Permissions > Role Permissions**.
6. In the upper-left corner of the **Role authorization** page, click **Create Custom Role**.
7. On the **Create Custom Role** page, click **Advanced Settings**.
8. In the **Advanced Settings** section, click **Create RAM Role & Attach Policy**.
9. In the Basic Configurations step of the **Create RAM Role & Attach Policy** wizard, configure parameters.
   i. Enter a **Role name** and **Description**, and set the **Sharing Scope**. The description is optional. By default, **Global** is selected for **Sharing Scope**, which indicates that the role is visible to and can be assumed by all organizations. Click **Next**.
   ii. In the Configure Custom Policy step, select an existing policy or create a new policy, and then click **Next**.

   > **Note**
   > - You can select a maximum of five policies for a RAM role.
   > - For information about how to create a policy, see Attach a custom policy to the role.

   iii. In the Configure Trust Policy step, select values from the **Trust Organization** and **Trust Cloud Service** drop-down lists separately. Click **Next**.

   > **Note** You can select multiple trust organizations and trust cloud services at a time for a RAM role.

   iv. In the Preview Policy step, check the content of the selected policies and trust policies, and click **Create**.
10. (Optional) Add authorized users to the RAM role.

    When you create a RAM role, you can add authorized users to the role. After you add authorized users to the RAM role, the authorized users are granted the permissions of the role.
    i. In the left-side navigation pane of the Enterprise page, choose Permissions > **Role Permissions**. On the Role authorization page, click the name of the role of which permissions you want to grant to users.
    ii. On the details page of the role, click the **Authorized Personnel** tab.

iii. On the **Authorized Personnel** tab, click **Authorized** in the **Authorize Users** section.

iv. In the **Authorized User** dialog box, select the users to which you want to grant the permissions of the role and click **Confirm authorization**.

To add authorized users to the RAM role, you must create users in advance. For more information, see Create a user and a user group.

Attach a custom policy to the role

When you create a RAM role, in the Configure Custom Policy step you can click **Create Policy** in the upper-right corner of the page to create a policy.

1. In the **Create Policy** dialog box, configure the following parameters and then click **OK**.

Parameter	Description
Policy Name	Specify a name for the custom policy. The name can be up to 15 characters in length. Example: *MangeNameSpace*.
Sharing Scope	Select the sharing scope of the role based on your business requirements. Valid values: ○ **Global**: The role is visible to and can be assumed by all organizations. This is the default option. ○ **Current Organization**: The role is visible to and can be assumed by the organization to which the user belongs. ○ **Subordinate Organization**: The role is visible to and can be assumed by the organization to which the user belongs and the subsidiaries of the organization. **Note** If you use the admin account to log on to the Apsara Uni-manager Management Console, the sharing scope is automatically set to **Global** and cannot be changed. If you use another account to log on to the Apsara Uni-manager Management Console, the preceding three options are displayed for the Sharing Scope parameter.
Description	Enter the description of the policy, such as the capabilities of the policy.

Parameter	Description

Policy Content	The following code provides a sample policy.
	
	{
	"Version": "1",
	"Statement": [
	{
	"Action": [
	"edas:CreateNamespace"
	],
	"Resource": [
	"acs:edas:*:*:namespace/*"
	],
	"Effect": "Allow"
	},
	{
	"Action": [
	"edas:ManageNamespace"
	],
	"Resource": [
	
	"acs:edas:*:*:namespace/$namespace"
	],
	"Effect": "Allow"
	},
	{
	"Action": [
	"edas:ReadNamespace",
	"edas:DeleteNamespace"
	],
	"Resource": [
	
	"acs:edas:*:*:namespace/$namespace"
	],
	"Effect": "Allow"
	}
	]
	}
	
	This policy allows the actions of creating, editing, and deleting a namespace.

Parameter	Description

3.1.3. Repository access control

Users can obtain access to a repository if you grant them permissions. This topic describes how to configure access control for repositories in different scenarios.

Precautions

When you grant permissions to a user, pay attention to the following instructions to make sure that you do not grant excessive permissions to the user.

You can grant a user the AdministratorAccess permission that contains management permissions on all Alibaba Cloud resources. In this case, the user has all permissions on Container Registry, regardless of whether the user is granted permissions before.

System policy configuration

- AliyunContainerRegistryFullAccess

  This policy grants a user the same permissions on image resources as those of an Alibaba Cloud account. The user can perform all operations on image resources.

```
{
  "Statement": [
    {
      "Action": "cr:*",
      "Effect": "Allow",
      "Resource": "*"
    }
  ],
  "Version": "1"
}
```

- AliyunContainerRegistryReadOnlyAccess

  This policy grants a user the read-only permission on all image resources. For example, the user can view the repository list and pull images.

```
{
  "Statement": [
    {
      "Action": [
        "cr:Get*",
        "cr:List*",
        "cr:PullRepository"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ],
  "Version": "1"
}
```

Policy configuration for typical scenarios

- Scenario 1

  Scenario: Grant a user the read-only permission on a namespace that is named juzhong. After the user logs on to the Container Registry instance, the user can pull all images in the namespace juzhong. The user can view information about the namespace and all repositories in the namespace by calling API operations.

```
{
  "Statement": [
    {
      "Action": [
        "cr:Get*",
        "cr:List*",
        "cr:PullRepository"
      ],
      "Effect": "Allow",
      "Resource": [
        "acs:cr:*:*:repository/juzhong/*"
      ]
    }
  ],
  "Version": "1"
}
```

  Notice If you want to allow the user to view all the namespaces in the console, add the following authorization configurations. Then, the user can view all the namespaces and repositories. However, the user can pull only images from the repositories in the namespace juzhong.

```
{
  "Statement": [
    {
      "Action": [
        "cr:Get*",
        "cr:List*",
        "cr:PullRepository"
      ],
      "Effect": "Allow",
      "Resource": [
        "acs:cr:*:*:repository/juzhong/*"
      ]
    },
    {
      "Action": [
        "cr:ListNamespace",
        "cr:ListRepository"
      ],
      "Effect": "Allow",
      "Resource": [
        "*"
      ]
    }
  ],
  "Version": "1"
}
```

- Scenario 2

  Scenario: Grant a user all permissions on a repository, such as the repository nginx in the namespace juzhong in the China (Hangzhou) region.

  > **Notice** If you want to allow the user to manage repositories in the console, add the relevant configurations by referring to scenario 1.

```
{
  "Statement": [
    {
      "Action": [
        "cr:*"
      ],
      "Effect": "Allow",
      "Resource": [
        "acs:cr:cn-hangzhou:*:repository/juzhong/nginx"
      ]
    },
    {
      "Action": [
        "cr:Get*",
        "cr:List*"
      ],
      "Effect": "Allow",
      "Resource": [
        "acs:cr:*:*:repository/juzhong"
      ]
    }
  ],
  "Version": "1"
}
```

- Scenario 3

  Scenario: Grant a user all permissions on a namespace.

  > **Notice** You can implement the scenario only by calling API operations. If you want to allow the user to view all repositories in the console, add the relevant configurations by referring to scenario 1.

```
{
    "Statement": [
        {
            "Action": [
                "cr:*"
            ],
            "Effect": "Allow",
            "Resource": [
                "acs:cr:cn-hangzhou:*:repository/juzhong",
                "acs:cr:cn-hangzhou:*:repository/juzhong/*"
            ]
        }
    ],
    "Version": "1"
}
```

Authentication rules of Container Registry

- ARN format

The following table describes the Alibaba Cloud Resource Name (ARN) format in an authorization policy when you grant a user access to the resources.

Resource type	ARN format in an authorization policy
repository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname

The following table describes the parameters in the ARN format.

Parameter	Description
$regionid	The region ID, which can be replaced by an asterisk (*).
$accountid	The ID of the Alibaba Cloud account, which can be replaced by an asterisk (*).
$namespacename	The name of the namespace.
$repositoryname	The name of the repository.

- Authorization rules

  When you access the Container Registry API as a user or by using Security Token Service (STS), Container Registry checks whether you have obtained the required permissions. The permissions that Container Registry checks vary based on the resources that are requested by the API operation and the syntax of the API operation. The following table describes the authentication rules for different API operations.

API	Action	Resource
Creates a namespace.	cr:CreateNamespace	*
Deletes a namespace.	cr:DeleteNamespace	acs:cr:$regionid:$accountid:repository/$namespacename
Updates a namespace.	cr:UpdateNamespace	acs:cr:$regionid:$accountid:repository/$namespacename
Queries a namespace.	cr:GetNamespace	acs:cr:$regionid:$accountid:repository/$namespacename
Queries namespaces.	cr:ListNamespace	*
Creates a repository.	cr:CreateRepository	acs:cr:$regionid:$accountid:repository/$namespacename
Deletes a repository.	cr:DeleteRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname

API	Action	Resource
Updates a repository.	cr:UpdateRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Queries a repository.	cr:GetRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Queries repositories.	cr:ListRepository	*
Queries repositories in a namespace.	cr:ListRepository	*
Queries the tag information about a repository.	cr:ListRepositoryTag	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Deletes an image tag.	cr:DeleteRepositoryTag	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Queries the manifest information about an image.	cr:GetRepositoryManifest	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Queries the information about image layers.	cr:GetRepositoryLayers	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Queries a temporary authorization token.	cr:GetAuthorizationToken	*
Pulls images.	cr:PullRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname
Pushes images	cr:PushRepository	acs:cr:$regionid:$accountid:repository/$namespacename/$repositoryname

3.2. Create a namespace

This topic describes how to create a namespace in the Container Registry console.

Context

A namespace is a collection of repositories. We recommend that you place the repositories of a company or an organization in the same namespace.

- Sample namespaces that are named after a company: aliyun and alibaba
- Sample namespace that is named after a team or an organization: team

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Standard Edition Instance.
3. In the left-side navigation pane of the management page of the Standard Edition instance, click **Namespace**.
4. On the **Namespace** page, click **Create Namespace**.
5. On the **Create Namespace** page, configure **Organization**, **Resource Set**, **Region**, and Namespace Name, and then click **Submit**.

What's next

You can create image repositories in the namespace.

3.3. Creates an image repository

This topic describes how to create an image repository in the Container Registry console.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Standard Edition Instance.
3. In the left-side navigation pane of the management page of the Standard Edition instance, click **Repository**.
4. On the **Repository** page, click **Create Repository**.
5. On the **Create Repository** page, configure parameters and then click **Submit**.

Parameter	Description
Organization	The organization to which the repository belongs.
Resource Set	The resource set to which the repository belongs.
Region	The region in which the repository resides.
Namespace	The **namespace** to which the repository belongs.
Repository Name	The name of the repository. The name must be 1 to 64 characters in length and can contain lowercase letters, digits, hyphens (-), and underscores (_). The name cannot start with a hyphen (-) or an underscore (_).
Digest	The digest of the repository.
Description	The description of the repository. The description can be up to 2,000 characters in length.

Parameter	Description
Repository Type	Public repositories and private repositories are supported.

What's next

After the repository is created, you can click **Manage** in the **Actions** column of the repository to go to the **Details** page and learn how to manage the repository.

3.4. Configure an access credential

Access credentials ensure the secure upload and download of container images. This topic describes how to configure an access credential.

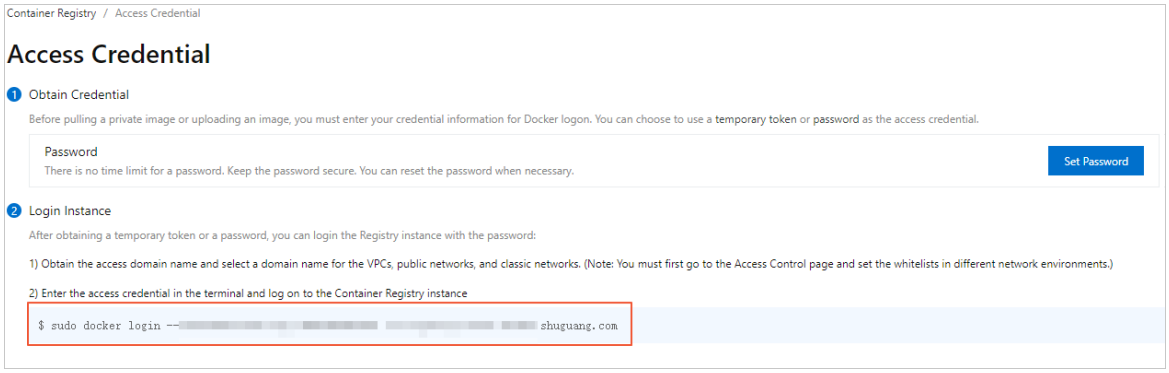
Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Standard Edition Instance.
3. In the left-side navigation pane of the management page of the Standard Edition instance, click **Access Credential**.
4. On the **Access Credential** page, click **Set Password**.
5. In the **Set Password** dialog box, enter the password and confirm the password. Then, click **Confirm**.
6. (Optional)For information about how to set a temporary token, see *GetAuthorizationToken* of *API Operations for Standard Edition* in the *Container Registry Developer Guide*.

Other operations

You can use the password or temporary token that you set to log on to the Container Registry instance.

1. On the Access Credential page, check the logon command.



2. Run the logon command and enter the logon password as prompted in the command output.

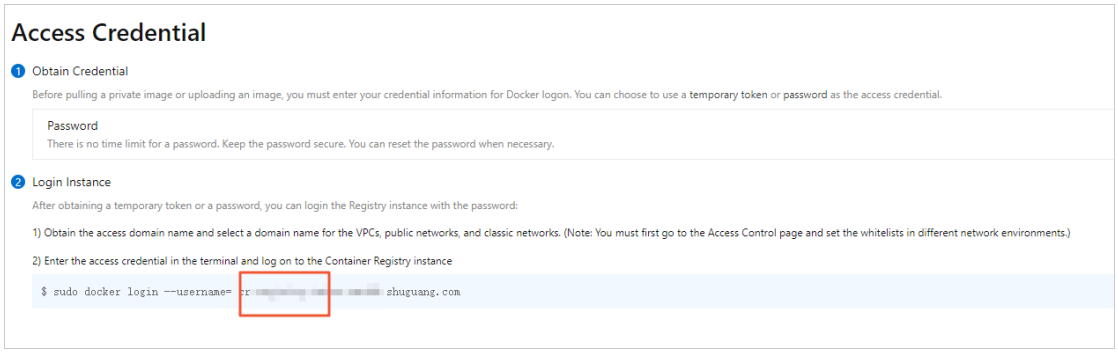
3.5. Upload and download images

This topic describes how to upload and download images.

Procedure

1. Obtain the username.
   i. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
   ii. On the **Instances** page, click the card of a Standard Edition Instance.
   iii. In the left-side navigation pane of the management page of the Standard Edition instance, click **Access Credential**.

   On the **Access Credential** page, view and record the username and the domain name of the repository.



2. Run the following command to log on to the image repository:

```
sudo docker login --username=<Username> <Domain name of the repository>
```

   As prompted in the command output, enter the password that you set when you activated Container Registry. If you forget your password, you can reset it. For more information, see Configure an access credential.

3. Upload and download an image.
   - Upload an image
     a. Run the following command to add a tag to the image:

```
sudo docker tag <ID of the image> <Domain name of the repository>/<Name of the namespace>/<Name of the repository>:<Image version number>
```

     b. Run the following command to upload the image:

```
sudo docker push <Domain name of the repository>/<Name of the namespace>/<Name of the repository>:<Image version number>
```

   - Download an image

     Run the following command to download an image:

```
sudo docker pull <Domain name of the repository>/<Name of the namespace>/<Name of the repository>:<Image version number>
```

4. (Optional) Modify the name of an image.
   i. Run the following command to query the ID of the image:

```
sudo docker images
```

ii. Run the following command to modify the name of the image:

```
sudo docker tag <ID of the image> registryDomain/<Name of the namespace>/<Name of the repository>:<Image version number>
```

3.6. Replicate an image

Container Registry allows you to replicate images across data centers. This way, you can obtain the expected image from the nearest data center when you deploy workloads.

Prerequisites

- A namespace is created. For more information, see Create a namespace.
- An image repository is created in the namespace. For more information, see Creates an image repository.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Standard Edition Instance.
3. In the left-side navigation pane of the management page of the Standard Edition instance, click **Repository**.
4. On the **Repository** page, find the repository that you want to manage and click **Manage** in the Actions column.
5. In the left-side navigation pane of the details page of the repository, click **Image Replication**.
6. On the Image Replication page, click **Create Replication Task**.
7. In the **Select Destination Repository** dialog box, configure parameters and then click **OK**.

Parameter	Description
Version to be Replicated	Select the version of the repository to be replicated.
Destination Repository	Select the region, namespace, and name of the destination repository and enter the version of the destination repository.

Result

On the **Image Replication** page, you can view the status of the replication task.

You can click **Details** in the Actions column of the task to check the replication status of image layers.

3.7. Configure a trigger

Container Registry provides the trigger feature for image repositories. If you create a trigger for an image repository, a notification is pushed to you when an image is built. This facilitates the redeployment of images for applications. If you create a trigger for a container service, the applications on the container service automatically pull and redeploy the new images that are built on the container service. This topic describes how to configure a trigger.

Security rules on trigger usage

- HTTP: Port 80 is used by default.

  If you want to use another port, append the port number to the end of the trigger URL. You can use only the following port numbers: 80, 21, 443, 70, 210, 280, 488, 591, 777, and from 1025 to 65535.

- HTTPS: Port 443 is used by default.

  Only port 443 is supported. If you want to use another port, use HTTP.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Standard Edition instance.
3. In the left-side navigation pane of the management page of the Standard Edition instance, click **Repository**.
4. On the **Repository** page, find the repository that you want to manage and click **Manage** in the **Actions** column.
5. In the left-side navigation pane of the details page of the repository, click **Trigger**.
6. On the **Trigger** page, click **Create**.
7. In the **Create Trigger** dialog box, configure parameters and click **Confirm**.

Parameter	Description
Name	The name of the trigger.
Trigger URL	Enter the URL of the trigger that is created in Container Service for Kubernetes (ACK).

Parameter	Description
Trigger	You can select Every time, Expression trigger, or Tag triggers. ○ Every time: Each time an image is built, an image pull is triggered. ○ Expression trigger: Tags are filtered based on a regular expression. An image pull is triggered only when a tag matches the regular expression. You can enter a simple regular expression, such as `release-v.*`. An image pull is triggered only after an image with a tag that starts with `release-v` is built. Otherwise, no image pull is triggered and the access status code in the access log is **Untriggered**. ○ Tag triggers: An image pull is triggered based on the specified tags. You can specify a maximum of 10 tags based on which an image pull needs to be triggered. Then, an image pull is triggered only when images with the specified tags are built. Otherwise, no image pull is triggered and the access status code in the access log is **Untriggered**.

3.8. Sign container images

When you manage container images, you can use content trust to verify both the integrity and the publisher of images. Image publishers can encrypt images by using digital signatures that are stored in Container Registry.

Install and configure signature tools

1. Install Alibaba Cloud Command Line Interface (CLI). For more information, see Install Alibaba Cloud CLI.
2. Run the following command to configure Alibaba Cloud CLI:

```
./aliyun configure set \ --profile akProfile \ --mode AK \ --region cn-qingdao-env17-d01 \ --access-key-id yourAK \ --access-key-secret yourAK
```

3. After you install and configure the GPG tool, run the following command to export the public key:

```
# Query pubKeyIdgpg --fingerprint# Export public key gpg --armor --output public-key.txt --export yourUser.
```

> **Note** For Apsara Stack instances, instanceId is set to default.

Sign images

1. Run the following command to obtain the image tag URL:

```
echo image://region/instanceId/namespace/repo@digest | tee imageURL.txt#e.g.#echo image://cn-qingdao-env17-d01/default/kritis-test/busybox2@sha256:2f122941b5850006dbb7adda78d2ea5b382841ca6569fd174bd24c14bfff3dca > imageURL.txt
```

2. Run the following command to use GPG to sign the unique URL of the image. By default, imageURL.txt.asc is generated.

```
gpg --armor --sign imageURL.txt
```

The imageURL.txt.asc file includes the following content:

```
-----BEGIN PGP MESSAGE-----owGbwMvMwMEo63vGqaX74wXGNXVJ3CmpaYmlOSV6JRUlcZy7vmfmJqanWunr
****.......=O2TP-----END PGP MESSAGE-----
```

3. Run the following command to create a metadata namespace:

```
./aliyun cr CreateMetadataNamespace --force --version 2018-12-01 --endpoint cr.inter.en
v17e.shuguang.com --NamespaceName kritis-test-2 --Description "for test"
```

4. Run the following command to create a signature note:

```
cat <<EOF > note.json| jq{ "name": "/namespaces/kritis-test-2/notes/image-sign", "LongD
escription": "long", "ShortDescription": "short", "ExpirationTime": "2021-01-01T00:00:0
0Z", "Kind": "ATTESTATION", "Attestation": { "Hint": { "HumanReadableName": "ACR" } }}E
OF
```

5. Run the following command to create an occurrence for the image signature and save the occurrence to the metadata service:

```
cat <<EOF > occurrence.json | jq{ "Name": "/namespaces/kritis-test-2/occurrences/random
Id1", "NoteName": "/namespaces/kritis-test-2/notes/image-sign", "ResourceUri": "image:/
/cn-qingdao-env17-d01/default/kritis-test/busybox2@sha256:2f122941b5850006dbb7adda78d2e
a5b382841ca6569fd174bd24c14bff****", "Kind": "ATTESTATION", "Attestation": { "Signature
s": [ { "Signature": $(cat imageURL.txt.asc|jq -R --slurp), "PublicKeyId": "E5B5FF2AFC3
A1D70FE3CE57C1D4DCC42848****" } ] }}EOF
```

```
./aliyun cr CreateMetadataOccurrence --force --version 2018-12-01 --endpoint cr.inter.e
nv17e.shuguang.com --NamespaceName kritis-test-2 --OccurrenceName randomId1 --Occurrenc
e "$(cat occurrence.json)"
```

6. Run the following command to query the image signature:

```
# Obtain a list of occurrences for a specified note: ./aliyun cr ListMetadataOccurrence
s --force --version 2018-12-01 --endpoint cr.inter.env17e.shuguang.com --NamespaceName
kritis-test-2 --NoteName image-sign --PageNo 1 --PageSize 5# Obtain a list of occurrenc
es for a specified note that occurs on a resource: ./aliyun cr ListMetadataOccurrences
--force --version 2018-12-01 --endpoint cr.inter.env17e.shuguang.com --NamespaceName kr
itis-test-2 --NoteName image-sign --PageNo 1 --PageSize 5 --ResourceURIs '["image://cn-
qingdao-env17-d01/default/kritis-test/busybox2@sha256:2f122941b5850006dbb7adda78d2ea5b3
82841ca6569fd174bd24c14bfff****"]'
```

Appendix 1: Use GPG commands to generate publicKeyData

1. Run the following command to find the local GPG key user that you want to use:

> **Note** The user in the following example is `abcdef@example.com`.

```
gpg --list-keyspub rsa2048 2020-01-08 [SC] [Valid until: 2022-01-07] 7726310BC6E11E9B57
B9CC08E2932E4363F3***uid [absolute] abcdef <abcdef@example.com>sub rsa2048 2020-01-08 [
E] [Valid until: 2022-01-07]
```

2. Export the public key of this user and encode the content of the public key in the Base64 format to generate publicKeyData.

```
$ gpg --armor --export <user> |base64 | tr -d '\n'# For example, the user in the preced
ing example is abcdef@example.com, so the command is: gpg --armor --export abcdef@examp
le.com |base64 | tr -d '\n'# export publicKeyData=$(gpg --armor --export abcdef@example
.com |base64 | tr -d '\n')
```

Appendix 2: Obtain an image digest value

You can use the following method to find an image digest value. In this example, the image URL is registry.acs.example.com/kritis-test/alpine:3.11.

```
$ docker pull registry.acs.example.com/kritis-test/alpine:3.11$ docker images --digests | g
rep registry.acs.example.com/kritis-test/alpineregistry.acs.example.com/kritis-test/alpine
3.11 sha256:ddba4d27a7ffc3f86dd6c2f92041af252a1f23a8e742c90e6e1297bfa1bc0c45 e7d92cdc71fe 2
months ago 5.59MB
```

The output shows that the image digest value is sha256:ddba4d27a7ffc3f86dd6c2f92041af252a1f23a8e742c90e6e1297bfa1bc0c45. When you create resources, the following image ID must be used: registry.acs.example.com/kritis-test/alpine:sha256:ddba4d27a7ffc3f86dd6c2f92041af252a1f23a8e742c90e6e1297bfa1bc0c45.

4.Container Registry Advanced Edition

4.1. Permission management

4.1.1. Create a user and a user group

You can create users and assign different roles to the users to meet different requirements for system access control as an administrator. This topic describes how to create a user and a user group.

Prerequisites

- Before you log on to the Apsara Uni-manager Management Console, the endpoint of the console is obtained from the deployment personnel.
- A browser is available. We recommend that you use Google Chrome.

Create an organization

To create a user or a user group, you must create an organization in advance.

1. Enter the URL of the Apsara Uni-manager Management Console in the address bar and press Enter.
2. Enter your username and password.

   Obtain the username and password that you use to log on to the Apsara Uni-manager Management Console from the operations administrator.

   > **Note** When you log on to the Apsara Uni-manager Management Console for the first time, you must change the password of your username. For security purposes, your password must meet the minimum complexity requirements. The password must be 10 to 32 characters in length and must contain at least two of the following character types:
   >
   > - Letters
   > - Digits
   > - Special characters, which include ! @ # $ %

3. Click **Log On**.
4. In the top navigation bar, click **Enterprise**.
5. In the left-side navigation pane of the Enterprise page, choose Resources > **Organizations**.
6. In the left-side organization navigation tree, click the name of a parent organization under which you want to create an organization. Click **Create Organization** in the upper-right corner of the page.
7. In the Add Organization dialog box, enter a name for the organization and click **OK**.
8. (Optional) If the multi-cloud management feature is enabled for the primary node, the **Synchronously Create Multi-cloud Organization** check box is displayed after you click **OK**.

   Select **Synchronously Create Multi-cloud Organization**, select clouds on which the organization is synchronously created, and then click **OK**.

Create a user group

A role cannot be directly assigned to a user. You can assign a role to a user group. Then, users in the user group can obtain the permissions of the role.

Relationships between user groups and users:

- A user group can contain zero or more users.
- A user does not necessarily belong to a user group.
- A user can be added to multiple user groups.

Relationships between user groups and organizations:

- A user group belongs to only one organization.
- You can create multiple user groups within an organization.

Relationships between user groups and roles:

- A role can be assigned to multiple user groups.
- When a role is assigned to a user group, the permissions that the role has are automatically granted to the users within the user group.

Relationships between user groups and resource sets:

- A resource set can be added to zero or more user groups.
- A user group can be added to multiple resource sets.

1. In the top navigation bar, click **Enterprise**.
2. In the left-side navigation pane, choose **Users > User Groups**.
3. Click **Create a user group** in the upper-left corner of the page.
4. In the dialog box that appears, configure **User Group Name**, **Organization**, and **Role authorization**.

Parameter	Description
User Group Name	The name of the user group. The name must be 3 to 255 characters in length and can contain letters, digits, underscores (_), hyphens (-), periods (.), and at signs (@).
Organization	The organization to which the user group belongs.
Role authorization	The roles that are assigned to the user group.

5. Click **OK**.

Create a user

You can create users based on your business requirements.

1. In the top navigation bar, click **Enterprise**.
2. In the left-side navigation pane, choose **Users > Users**.
3. Click the **System Users** tab. Click **Create a user**.
4. In the dialog box that appears, configure parameters.

Parameter	Required	Description
User name	Yes	The Apsara Stack tenant account name of the user. The name must be 1 to 64 characters in length and can contain letters, digits, hyphens (-), underscores (_), and periods (.).
Display name	Yes	The display name of the user. The name must be 1 to 128 characters in length and can contain letters, digits, hyphens (-), underscores (_), periods (.), and at signs (@).
Role	Yes	The role to be assigned to the user. **Note** You can enter role names in the field. Fuzzy match is supported.
Organization	Yes	The organization to which the user belongs.
Logon policy	Yes	The logon policy that specifies the logon time of the user and the IP address that the user logs on to. If you do not configure this parameter, the default policy is attached to the created user. **Note** The default policy does not specify the logon time of the user and the IP address that the user logs on to. To specify the logon time and IP address, you can modify the logon policy or create a logon policy.
Phone	Yes	The mobile phone number of the user. If you want to send text messages about the resource requests and usage to the mobile phone number, make sure that the specified mobile phone number is valid. **Note** If the mobile phone number of the user changes, update it on the system in a timely manner.
Landline	No	The landline number of the user. The landline number must be 4 to 20 characters in length and can contain only digits and hyphens (-).
Email	Yes	The email address of the user. If you want to send emails about the resource requests and usage to the email address, make sure that the specified email address is valid. **Note** If the email address changes, update it on the system in a timely manner.

Parameter	Required	Description
DingTalk Key	No	The key of the chatbot for the DingTalk group to which the user belongs. For more information about how to configure the key, see DingTalk development documentation.
Notify User by Email	No	If this option is selected, the Apsara Uni-manager Management Console informs the user configured as the alert contact by email whenever an alert is generated. **Note** You must configure an email server to receive emails. For more information, contact on-site O&M engineers.
Notify User by DingTalk	No	If this option is selected, the Apsara Uni-manager Management Console informs the user configured as the alert contact by DingTalk whenever an alert is generated.

5. Click **OK**.

4.1.2. Create a role and grant permissions to the role

You can create custom roles, attach permission policies to the roles, and assign the roles to a user to grant permissions to the user. This topic describes how to create a role, assign the role to a user group, and attach a policy to the role.

Context

A role is a set of access permissions. Each role has a range of permissions. A user can have multiple roles, which means that the user is granted all of the permissions defined for each role. A role can be used to grant the same set of permissions to a group of users.

The total number of custom and default roles cannot exceed 20.

Create a role

1. Enter the URL of the Apsara Uni-manager Management Console in the address bar and press Enter.
2. Enter your username and password.

   Obtain the username and password that you use to log on to the Apsara Uni-manager Management Console from the operations administrator.

> **Note** When you log on to the Apsara Uni-manager Management Console for the first time, you must change the password of your username. For security purposes, your password must meet the minimum complexity requirements. The password must be 10 to 32 characters in length and must contain at least two of the following character types:
>
> - Letters
> - Digits
> - Special characters, which include ! @ # $ %

3. Click **Log On**.
4. In the top navigation bar, click **Enterprise**.
5. In the left-side navigation pane, choose **Permissions > Role Permissions**.
6. In the upper-left corner of the **Role authorization** page, click **Create Custom Role**.
7. On the **Create Custom Role** page, click **Advanced Settings**.
8. In the **Advanced Settings** section, click **Create RAM Role & Attach Policy**.
9. In the Basic Configurations step of **Create RAM Role & Attach Policy** wizard, configure parameters.
   i. Enter a value for **Role name** and **Description** separately. Select **Sharing Scope**. By default, **Global** is select for **Sharing Scope**, which indicates that the role is visible to and used by all organizations. Click **Next**.
   ii. In the Configure Custom Policy step, select an existing policy or create a new policy, and then click **Next**.

   > **Note** You can select a maximum of five policies for a RAM role.

   iii. In the Configure Trust Policy step, select values from the **Trust Organization** and **Trust Cloud Service** drop-down lists separately. Click **Next**.

   > **Note** You can select multiple trust organizations and trust cloud services at a time for a RAM role.

   iv. In the Preview Policy step, check the content of selected policies and trust policies, and click **Create**.
10. (Optional)Add authorized users to the RAM role.

    When you create a RAM role, you can add authorized users to the role. After you add authorized users to the RAM role, the users are granted the permissions of the role.

    i. **>**
    ii. Create a user and a user group
11. i.
    ii.

iii.

	■ ■ ■ Note

	{
	"Version": "1",
	"Statement": [
	{
	"Action": [
	"edas:CreateNamespace"
	],
	"Resource": [
	"acs:edas:*:*:namespace/*"
	],
	"Effect": "Allow"
	},
	{
	"Action": [
	"edas:ManageNamespace"
	],
	"Resource": [
	"acs:edas:*:*:namespace/$namespace"
	],
	"Effect": "Allow"
	},
	{
	"Action": [
	"edas:ReadNamespace",
	"edas:DeleteNamespace"
	],
	"Resource": [
	"acs:edas:*:*:namespace/$namespace"
	],
	"Effect": "Allow"
	}
	]
	}

iv.

4.1.3. Repository access control

You can create users and grant different permissions to the users. This way, you can control the access of the users to Container Registry resources. This topic describes how to configure access control for repositories in different scenarios.

Precautions

When you authorize a user, pay attention to the following instructions to make sure that you do not grant excessive permissions to the user.

You can grant a user the AdministratorAccess permission that contains management permissions on all Alibaba Cloud resources. In this case, the user has all permissions on Container Registry, regardless of whether the user is granted permissions before.

System policy configuration

- AliyunContainerRegistryFullAccess

  This policy grants a user the same permissions on image resources as those of an Alibaba Cloud account. The user can perform all operations on image resources.

```
{
  "Statement": [
    {
      "Action": "cr:*",
      "Effect": "Allow",
      "Resource": "*"
    }
  ],
  "Version": "1"
}
```

- AliyunContainerRegistryReadOnlyAccess

  This policy grants a user the read-only permission on all image resources. For example, the user can view the repository list and pull images.

```
{
  "Statement": [
    {
      "Action": [
        "cr:Get*",
        "cr:List*",
        "cr:Pull*"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ],
  "Version": "1"
}
```

Policy configuration for typical scenarios

Grant the user the read and write permissions on a namespace of a Container Registry Advanced Edition instance.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cr:ListInstance*",
                "cr:GetInstance*",
                "cr:ListSignature*"
            ],
            "Resource": "*"
        },
        {
            "Action": [
                "cr:*"
            ],
            "Effect": "Allow",
            "Resource": [
                "acs:cr:*:*:repository/$instanceid/$namespace/*",
                "acs:cr:*:*:repository/$instanceid/$namespace"
            ]
        },
        {
            "Action": [
                "cr:List*"
            ],
            "Effect": "Allow",
            "Resource": [
                "acs:cr:*:*:repository/$instanceid/*",
                "acs:cr:*:*:repository/$instanceid/*/*"
            ]
        }
    ],
    "Version": "1"
}{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cr:ListInstance*",
                "cr:GetInstance*",
                "cr:ListSignature*"
            ],
            "Resource": "*"
        },
        {
            "Action": [
                "cr:*"
            ],
            "Effect": "Allow",
            "Resource": [
                "acs:cr:*:*:repository/$instanceid/$namespace/*",
```

```
                "acs:cr:*:*:repository/$instanceid/$namespace"
            ]
        },
        {
            "Action": [
                "cr:List*"
            ],
            "Effect": "Allow",
            "Resource": [
                "acs:cr:*:*:repository/$instanceid/*",
                "acs:cr:*:*:repository/$instanceid/*/*"
            ]
        }
    ],
    "Version": "1"
}
```

Authentication rules of Container Registry

- ARN format

  The following table describes the Alibaba Cloud Resource Name (ARN) format in an authorization policy when you use RAM to authorize access to the resources.

Resource type	ARN format in an authorization policy
*	acs:cr:$regionid:$accountid:*
instance	acs:cr:$regionid:$accountid:instance/$instanceid
repository	acs:cr:$regionid:$accountid:repository/$instanceid/* acs:cr:$regionid:$accountid:repository/$instanceid acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/* acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename

  The following table describes the parameters in the ARN format.

Parameter	Description
regionid	The region ID, which can be replaced by an asterisk (*).
accountid	The ID of the Alibaba Cloud account, which can be replaced by an asterisk (*).

Parameter	Description
instanceid	The ID of the Container Registry Enterprise Edition instance.
namespacename	The name of the namespace.
repositoryname	The name of the repository.

- Authorization rules

  When you access the Container Registry API as a RAM user or by using Security Token Service (STS), Container Registry checks whether you have obtained the required permissions. The permissions that Container Registry checks vary based on the resources that are requested by the API operation and the syntax of the API operation. The following table describes the authentication rules for different API operations.

  > **Note** The asterisk (*) is used as a wildcard.

API	Action	Resource
Queries a pair of temporary username and password that you use to log on to a Container Registry instance.	cr:GetAuthorizationToken	*
Queries a namespace.	cr:GetNamespace	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename
Creates a namespace.	cr:CreateNamespace	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename
Queries namespaces.	cr:ListNamespace	acs:cr:$regionid:$accountid:repository/$instanceid/*
Deletes a namespace.	cr:DeleteNamespace	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename
Updates a namespace.	cr:UpdateNamespace	acs:cr:$regionid:$accountid:chart/$instanceid/$chartnamespacename
Queries a repository.	cr:GetRepository	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname
Queries image repositories.	cr:ListRepository	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/*

API	Action	Resource
Deletes an image repository.	cr:DeleteRepository	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname
Updates an image repository.	cr:UpdateRepository	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname
Queries image tags in an image repository.	cr:ListRepositoryTag	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname
Deletes an image from an image repository.	cr:DeleteRepositoryTag	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname
Queries the information about image layers of an image tag.	cr:GetRepositoryLayers	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname
Queries the manifest information of an image tag.	cr:GetRepositoryManifest	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname
Updates a trigger for an image repository.	cr:UpdateWebHook	acs:cr:$regionid:$accountid:repository/$instanceid/$namespacename/$repositoryname

4.2. Create a namespace

This topic describes how to create a namespace on a Container Registry Advanced Edition instance.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Advanced Edition instance, choose **Repository > Namespace**.
4. On the **Namespace** page, click **Create Namespace**.
5. On the **Create Namespace** page, configure the Namespace and Default Repository Type parameters, set whether to automatically create repositories, and then click **Submit**.
   - By default, if you push an image to a repository that does not exist in a namespace, Container Registry automatically creates the repository based on the repository name that you specify.

     To disable this feature, turn off Auto Create for the namespace.
   - By default, a repository that is automatically created upon image push is private.

You can set Default Repository Type to Public for a namespace to change the default repository type.

4.3. Create an image repository

This topic describes how to create an image repository on a Container Registry Advanced Edition instance.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Repositories**.
4. On the **Repositories** page, click **Create Repository**.
5. On the Create Image Repository page, configure the **Organization**, **Resource Set**, **Region**, **Namespace**, **Repository Name**, **Repository Type**, **Image Version**, **Digest**, and **Description** parameters, and then click **Submit**.

4.4. Configure an access credential

You can configure an access credential to set a password that you use to log on to a Container Registry Advanced Edition instance. This method can ensure that you securely control the upload and download of container images. This topic describes how to configure an access credential.

Context

Access credentials are independent of Alibaba Cloud accounts and passwords. Access credentials are available in two types:

- Password: A password is valid permanently. Keep it safe. If the password is lost, you can reset it.
- Temporary token: A temporary token is valid for one hour. If the temporary token is obtained by using Security Token Service (STS), the temporary token is valid so long as the STS token is valid.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Access Credential**.
4. On the **Access Credential** page, click **Set Password**.
5. In the **Set Password** dialog box, enter the password and confirm the password. Then, click **Confirm**.
6. (Optional) For information about how to set a temporary token, see *GetAuthorizationToken* of *API Management for Advanced Edition* in the *Container Registry Developer Guide*.

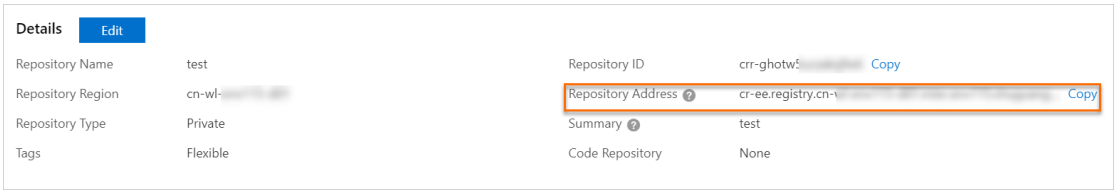
4.5. Upload and download images

This topic describes how to upload an image to and download an image from a Container Registry Advanced Edition instance.

Procedure

1. Obtain the address of the repository.
   i. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
   ii. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
   iii. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Repositories**.
   iv. On the **Repositories** page, click the name of the repository.

   On the **Details** page, obtain the address of the repository.



2. Run the following command to log on to the Advanced Edition instance.

```
docker login --username=<Your Alibaba Cloud account> <Address of the repository>
```

   As prompted in the command output, enter the password that you set when you activated Container Registry. If you forget your password, you can reset it. For information about how to reset a password, see Configure an access credential.

3. Upload and download an image.
   - Upload an image
     a. Run the following command to add a tag to the image:

```
docker tag <ID of the image> <Address of the repository>/<Name of the namespace>/
<Name of the repository>:<Version number of the image>
```

     b. Run the following command to upload the image to the repository:

```
 docker push <Address of the repository>/<Name of the namespace>/<Name of the rep
ository>:<Version number of the image>
```

   - Download an image

     Run the following command to download an image from the image repository:

```
docker pull <Address of the repository>/<Name of the namespace>/<Name of the reposito
ry>:<Version number of the image>
```

4.6. Configure a trigger

Container Registry provides the trigger feature for image repositories. If you create a trigger for an image repository, a notification is pushed to you when an image is built. This facilitates the redeployment of images for applications. If you create a trigger for a container service, the applications on the container service automatically pull and redeploy the new images that are built on the container service. This topic describes how to configure a trigger.

Security rules on trigger usage

- HTTP: Port 80 is used by default.

  If you want to use another port, append the port number to the end of the trigger URL. You can use only the following port numbers: 80, 21, 443, 70, 210, 280, 488, 591, 777, and from 1025 to 65535.

- HTTPS: Port 443 is used by default.

  Only port 443 is supported. If you want to use another port, use HTTP.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Repositories**.
4. On the **Repositories** page, click the name of the repository.
5. In the left-side navigation pane of the details page of the repository, click **Trigger**.
6. On the **Trigger** page, click **Create**.
7. In the **Create Trigger** dialog box, configure parameters and click **Confirm**.

Parameter	Description
Name	The name of the trigger.
Trigger URL	Enter the URL of the trigger that is created in Container Service for Kubernetes (ACK).

Parameter	Description
Trigger	You can select Every time, Expression trigger, or Tag triggers. ○ Every time: Each time an image is built, an image pull is triggered. ○ Expression trigger: Tags are filtered based on a regular expression. An image pull is triggered only when a tag matches the regular expression. You can enter a simple regular expression, such as `release-v.*`. An image pull is triggered only after an image with a tag that starts with `release-v` is built. Otherwise, no image pull is triggered and the access status code in the access log is **Untriggered**. ○ Tag triggers: An image pull is triggered based on the specified tags. You can specify a maximum of 10 tags based on which an image pull needs to be triggered. Then, an image pull is triggered only when images with the specified tags are built. Otherwise, no image pull is triggered and the access status code in the access log is **Untriggered**.

4.7. Replicate images between instances that belong to the same account

This topic describes how to manually and automatically replicate an image across regions between instances that belong to the same account.

Automate image replication across regions between instances that belong to the same account

You can configure a replication rule to automate image replication between two instances that belong to the same account. After you upload an image to the source instance that resides in one region, the system automatically replicates the image to the destination instance that resides in the other region.

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Distribution > Instance Replication**. On the page that appears, click **Create Rule**.
4. In the **Instance Information** step of the Create Rule wizard, set the Rule Name parameter, select **Same Account** for the **Synchronization Scenario** parameter, select the region and name of the target instance, and then click **Next**.

   A cloud department is an organization on the cloud platform. A cloud department is a collection of multiple persons and can represent a group company, a group branch, or a department in a

company.

5. In the **Replication Information** step, set **Replication Level** to **Namespace** or **Repository**, select a namespace or repository, and enter a regular expression to filter repository versions. Then, click **Create Rule**.
   When you upload an image to the source instance, the image is automatically replicated to the destination instance. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Distribution > Replication Record**. If the status of the replication task on the **Replication Record** page is Completed and the image exists in the destination instance, the image is automatically replicated between the instances that belong to the same account.

Manually replicate an image across regions between instances that belong to the same account

You can configure a replication rule to manually replicate an image from a source instance to a destination instance across regions.

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Repositories**.
4. On the **Repositories** page, click the name of the repository.
5. In the left-side navigation pane of the management page of the repository, click **Tags**.
6. On the **Tags** page, click **Replicate** in the **Actions** column of the image.
7. In the **Select Destination Repository** dialog box, select **Same Account** for the **Synchronization Scenario** parameter and select the region and name of the destination instance. Select a namespace and a repository, enter a regular expression to filter repository versions, set **Overwrite existing images that have the same name**, and then click **OK**.
   After you manually create the replication rule, image replication between the instances is immediately triggered. On the Tags page, you can click Image Sync Records in the Actions column of the image. If the status of the replication task on the **Replication Record** page is Completed and the image exists in the destination instance, the image is manually replicated between the instances that belong to the same account.

4.8. Replicate images between instances that belong to different accounts

This topic describes how to manually and automatically replicate an image between instances that belong to different accounts.

Prerequisites

You have obtained the permissions on the multi-cloud management platform. For more information, see *Multi-Cloud management platform* in *Apsara Uni-manager Management Console User Guide*.

Automate image replication between instances that belong to different accounts

You can configure a replication rule to automate image replication between two instances that belong to different accounts. After you upload an image to the source instance in one account, the system automatically replicates the image to the destination instance that belong to the other account.

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Distribution > Instance Replication**. On the page that appears, click **Create Rule**.
4. In the **Instance Information** step of the Create Rule wizard, set the Rule Name parameter and select Across Account for the Synchronization Scenario parameter. Select a destination cloud platform and a cloud department, select the name and ID of the target instance, and then click **Next**.

   A cloud department is an organization on the cloud platform. A cloud department is a collection of multiple persons and can represent a group company, a group branch, or a department in a company.
5. In the **Replication Information** step, set Replication Level to **Namespace** or **Repository**, select a namespace or repository, and enter a regular expression to filter repository versions. Then, click **Create Rule**.
   When you upload an image to the source instance, the image is automatically replicated to the destination instance. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Distribution > Replication Record**. If the status of the replication task on the **Replication Record** page is Completed and the image exists in the destination instance, the image is automatically replicated between the instances that belong to different accounts.

Manually replicate an image between instances that belong to different accounts

You can configure a replication rule to manually replicate an image from a source instance to a destination instance across accounts.

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Repositories**.
4. On the **Repositories** page, click the name of the repository.
5. In the left-side navigation pane of the management page of the repository, click **Tags**.
6. On the **Tags** page, click **Replicate** in the **Actions** column of the image.
7. In the Select Destination Repository dialog box, select Across Account for the Synchronization Scenario parameter and select a destination cloud platform and a cloud department. Select the name and ID of the target instance, select a namespace and a repository, and enter a regular expression to filter repository versions. Set whether to overwrite existing images that have the

same name and then click **OK**.
After you manually create the replication rule, image replication between the instances across accounts is immediately triggered. On the Tags page, you can click Image Sync Records in the Actions column of the image. If the status of the replication task on the **Replication Record** page is Completed and the image exists in the destination instance, the image is manually replicated between the instances that belong to different accounts.

4.9. Configure a repository to be immutable

Container Registry allows you to configure a repository to be immutable to prevent image tags from being overwritten due to unintended operations. After you configure a repository to be immutable, the existing and newly added image tags in the repository cannot be overwritten except for the tags of the latest. After you push an image with a tag other than latest to the repository, you cannot push another image with the same tag to the repository. This ensures that the images in the repository are consistent with the images that have been deployed in containers. This topic describes how to configure a repository to be immutable.

Procedure

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Repositories**.
4. On the Repositories page, find the repository that you want to configure and click **Manage** in the **Actions** column.
5. On the page that appears, click **Edit** in the Details section.
6. In the **Modify Settings** dialog box, select **Immutable** and click **Confirm**.

> **Note** To configure a repository to be mutable, clear **Immutable**.

Verification

1. Run the `docker push` command to push an image whose tag is not latest to the repository.

```
docker push <Address of the repository>/<Name of the namespace>/<Name of the repository>:v1
```

2. Push another image whose tag is the same as the tag of the image that you pushed in Step . The push request is denied. The following error is returned, which indicates that the image tag cannot be overwritten:

```
The requested tag already exists and cannot be overwritten.
```

3. Run the `docker push` command to push an image whose tag is latest to the repository.

```
docker push <Address of the repository>/<Name of the namespace>/<Name of the repository>:latest
```

4. Push another image whose tag is the same as the tag of the image that you pushed in Step . The push request is allowed. The image that you pushed in Step is overwritten by the image that you pushed in this step.
5. Configure the repository to be mutable by following the preceding procedure.
6. Push another image whose tag is the same as the tag of the image that you pushed in Step . The push request is allowed. The image that you pushed in Step is overwritten by the image that you pushed in this step.

4.10. Delete image tags

Container Registry allows you to delete multiple image tags of a Container Registry Enterprise Edition instance at a time. This topic describes how to delete multiple image tags at a time by configuring a tag retention policy.

Configure a tag retention policy

After you configure a tag retention policy, the image tags that do not match the policy are deleted.

1. Log on to the Container Registry console. For more information, see Log on to the Container Registry console.
2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Tags**.
4. On the **Tags** page, click **Create Retention Policy**.
5. In the **Create Retention Policy** dialog box, configure parameters in the **Set Deletion Scope** step and then click **Next**.

   The retention policy can be applied to a namespace or an image repository.

   - **Namespace**: The retention policy is applied to all image repositories in the specified namespace.
   - **Repository**: If you select Repository and then select namespaces and repositories in the namespaces, the retention policy is applied to the specified repositories in the namespaces.
6. In the **Configure Retention Policy** step, configure parameters and then click **Save**.

Parameter	Description
Retain the Latest Images (required)	Set the number of the latest images that you want to retain.
Retain Image Tags (optional)	In addition to the number of the latest images that you want to retain, you can specify the image tags that you want to retain. **Note** The default value of this parameter is an asterisk (*), which indicates that all image tags are retained in addition to the specified number of the latest images.

Manually trigger a tag deletion task

1. Log on to the Container Registry console. For more information, see Log on to the Container

Registry console.

2. On the **Instances** page, click the card of a Container Registry Advanced Edition instance.
3. In the left-side navigation pane of the management page of the Container Registry Advanced Edition instance, choose **Repository > Tags**.
4. On the **Tags** page, find the retention policy that you want to apply and click **Execute** in the **Actions** column.
5. In the **Tips** message, click **OK**.

4.11. Use the aliyun-acr-credential-helper component to pull images without a secret

You can use the aliyun-acr-credential-helper component to pull private images without a secret from instances of Container Registry Standard Edition and Advanced Edition. This component is automatically installed in Container Service for Kubernetes (ACK) clusters. This topic describes how to use the aliyun-acr-credential-helper component to pull a private image without a secret in different scenarios.

Prerequisites

- An ACK cluster is created. For more information, see *Create a Kubernetes cluster* in *Container Service for Kubernetes User Guide*.
- A kubectl client is connected to your cluster. For more information, see *Use kubectl to connect to a Kubernetes cluster* in *User Guide for Container Service for Kubernetes*.

Precautions

- If you want to use the aliyun-acr-credential-helper component, do not specify the imagePullSecret parameter. If the imagePullSecret parameter is specified in the template of a Kubernetes resource, such as a Deployment, the component becomes invalid.
- If a Kubernetes resource, such as a Deployment, uses custom service accounts, you must modify the service-account parameter in the configuration file of the aliyun-acr-credential-helper component. This way, the component is authorized to pull images by using the custom service accounts.
- Check whether the private image that you want to pull resides in the same region as your ACK cluster. By default, you can pull private images only from Container Registry instances that reside in the same region as your ACK cluster. If you want to pull images from Container Registry instances that reside in different regions from your ACK cluster, see Scenario 2 in this topic.
- After you create a service account in a cluster, it takes some time for the aliyun-acr-credential-helper component to renew the token of the service account. The new token for pulling private images is generated based on the default permissions of your ACK cluster. Applications with the service account can use the token to pull images only after the token is renewed. If you create an application immediately after you create a service account, the application will fail to pull images because it is unauthorized.
- By default, the configuration of the aliyun-acr-credential-helper component overwrites the imagePullSecret parameter of default service accounts in all namespaces. These service accounts are automatically modified when the service-account parameter of the **acr-configuration** ConfigMap in the kube-system namespace is changed.

- When you modify the **acr-configuration** ConfigMap in the kube-system namespace, make sure that you use the same indentation as the example in this topic. We recommend that you paste the YAML code provided in this topic to the editor, replace the corresponding values, and apply the configuration. This ensures that the format of the ConfigMap is valid.

Upgrade and configure the aliyun-acr-credential-helper component

Before you use the aliyun-acr-credential-helper component to pull images, you must update and configure the component by performing the following steps:

1. Configure the acr-configuration ConfigMap.

   Configure the acr-configuration ConfigMap in the ACK console

   i. Log on to the ACK console. For more information, see *Log on to the ACK console* in the *User Guide for Container Service for Kubernetes*.

   ii. In the left-side navigation pane, click **Clusters**.

   iii. On the **Clusters** page, click the name of the cluster for which you want to configure the acr-configuration ConfigMap.

   iv. In the left-side navigation pane of the **Cluster Information** page, choose **Configurations > ConfigMap**.

   v. In the upper part of the **ConfigMap** page, select **kube-system** from the Namespace drop-down list and find the **acr-configuration** ConfigMap. Then, modify the ConfigMap by using one of the following methods:

   - Method 1: Click **Edit** in the Actions column and edit the names and values of the **ConfigMap**.
   - Method 2: Click **Edit YAML** in the Actions column and edit the names and values of the **ConfigMap**.

   The following table describes the names and values of the acr-configuration ConfigMap.

Name of the acr-configuration ConfigMap	Description of the name	Value
service-account	The service accounts that are used by the aliyun-acr-credential-helper component to pull images.	Default value: **default**. **Note** Separate multiple service accounts with commas (,). Enter an asterisk (*) to specify all service accounts in all namespaces.

Name of the acr-configuration ConfigMap	Description of the name	Value

acr-registry-info	The information about Container Registry instances. Each instance can be specified by three fields of the String type in a YAML file.
	
	Note Set the three fields based on the following descriptions:
	■ instanceId: the ID of the Container Registry instance. This field is required for Container Registry Enterprise Edition instances.
	■ domains: the domain names of the Container Registry instance. This field is optional. By default, all domain names of the instance are specified. You can specify one or more domain names. Separate multiple domain names with commas (,).
		The value is empty by default. This means that images are pulled from the default repository of the Container Registry instance that resides in the same region as your ACK cluster.
		
		Sample configurations for a Container Registry Advanced Edition instance:
		`- instanceId: "cri-xxx"`
		`  domains: "xxx.com,yyy.com"`
		
		Sample configurations for a Container Registry Standard Edition instance:
		`- instanceId: ""`
		`  domains: "xxx.com,yyy.com"`

Name of the acr-configuration ConfigMap	Description of the name	Value
watch-namespace	The namespaces from which images can be pulled without a secret.	Default value: **default**. **Note** If the value is set to all, images can be pulled from all namespaces without a secret. Separate multiple namespaces with commas (,).
expiring-threshold	The duration after which the cache token expires.	Default value: **15m**. We recommend that you use the default value.

Configure the acr-configuration ConfigMap by using kubectl

i. Run the following command to go to the editing page of the acr-configuration ConfigMap:

```
kubectl edit cm acr-configuration -n kube-system
```

ii. Configure the parameters of the acr-configuration ConfigMap based on your requirements.

Sample configurations for Container Registry Advanced Edition and Standard Edition:

- Advanced Edition

```
apiVersion: v1
data:
  acr-api-version: "2018-12-01"
  acr-registry-info: |-
    - instanceId: "cri-xxx"
  expiring-threshold: 15m
  service-account: default
  watch-namespace: all
kind: ConfigMap
metadata:
 name: acr-configuration
  namespace: kube-system
 selfLink: /api/v1/namespaces/kube-system/configmaps/acr-configuration
```

- Standard Edition

```
apiVersion: v1
data:
  acr-api-version: "2018-12-01"
  acr-registry-info: |-
    - instanceId: ""
  expiring-threshold: 15m
  service-account: default
  watch-namespace: all
kind: ConfigMap
metadata:
 name: acr-configuration
 namespace: kube-system
 selfLink: /api/v1/namespaces/kube-system/configmaps/acr-configuration
```

Scenario 1: Pull private images from Container Registry Advanced Edition and Container Registry Standard Edition instances

ACK allows you to pull private images from both Container Registry Advanced Edition and Standard Edition instances, only from Container Registry Advanced Edition instances, or only from Container Registry Standard Edition instances. Modify the **acr-configuration** ConfigMap based on your business requirements. For more information, see Update and configure the aliyun-acr-credential-helper component. Sample configurations:

- Pull private images from Container Registry Advanced Edition instances.

```
data:
    service-account: "default"
    watch-namespace: "all"
    expiring-threshold: "15m"
    notify-email: "cs@aliyuncs.com"
    acr-registry-info: |
      - instanceId: "cri-xxx"
        domains: "xxx.com,yyy.com"
```

- Pull private images from Container Registry Standard Edition instances.

```
data:
    service-account: "default"
    watch-namespace: "all"
    expiring-threshold: "15m"
    notify-email: "cs@aliyuncs.com"
    acr-registry-info: |
      - instanceId: ""
        domains: "xxx.com,yyy.com"
```

- Pull private images from both Container Registry Advanced Edition and Standard Edition instances.

```
data:
    service-account: "default"
    watch-namespace: "all"
    expiring-threshold: "15m"
    notify-email: "cs@aliyuncs.com"
    acr-registry-info: |
      - instanceId: ""
      - instanceId: "cri-xxxx"
```