Alibaba Cloud
Resource Access Management

FAQ

Issue: 20181115

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 due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall 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 the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. 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 bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, 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 content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, 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 the Alibaba Cloud website, product programs, or content 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 contact Alibaba Cloud directly if you discover any errors in this document.

Generic conventions

Table -1: Style conventions

Style	Description	Example
	This warning information 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.
	This warning information 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 restore business.
	This indicates warning information, supplementary instructions, and other content that the user must understand.	 **Note:** Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	 **Note:** You can use **Ctrl** + **A** to select all files.
>	Multi-level menu cascade.	**Settings** > **Network** > **Set network type**
Bold	It is used for buttons, menus, page names, and other UI elements.	Click **OK**.
`Courier font`	It is used for commands.	Run the `cd /d C:/windows` command to enter the Windows system folder.
Italics	It is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	It indicates that it is a optional value, and only one item can be selected.	`ipconfig` *[-all|-t]*
{} or {a|b}	It indicates that it is a required value, and only one item can be selected.	`swich` *{stand | slave}*

Contents

1 Authorization for ECS instances

Questions

View ECS permission definitions

See *Authorization rules* in the ECS OpenAPI document.

Assign full ECS service management permissions to a subaccount

Add the system authorization policy “AliyunECSFullAccess” to the subaccount (or the group to which the subaccount belongs) on the RAM console.

Assign the ECS read-only permission to a subaccount

Create a subaccount on the RAM console and add the system authorization policy “AliyunECSReadOnlyAccess” to the subaccount.

For more information about how to add an authorization policy, see *Authorization*.

Allow a RAM user to view ECS instances in the Qingdao region but disallow the user to view disk or snapshot information

The permission for viewing ECS resource lists can be assigned based on region and resource type.

The following example describes how to authorize a subaccount to view only ECS instance information in the Qingdao region.

```
"Statement": [

    "Effect": "Allow",
    "Action": "ecs:DescribeRegions",
    "Resource": "*"


    "Effect": "Allow",
    "Action": "ecs:Describe*",
```

```
      "Resource": "acs:ecs:cn-qingdao:*:instance/*"

  "Version": "1"
```

Authorize a RAM user to manage two specified ECS instances

Assume that 10 ECS instances have been bought under your tenant account. As a RAM administrator, you want to authorize a RAM user to use only two of the ECS instances. In this case, you can create the following authorization policy:

Note:
he authorized RAM user can view all the ECS instances but can perform operations (such as the StopInstance operation) on only two of them. Currently, you cannot authorize a RAM user to view only the ECS instances that the user can operate.

Assume that the IDs of your ECS instances are i-001 and i-002. You must first create an authorization policy, which includes the permissions for managing i-001 and i-002 and viewing all ECS resources.

```
"Statement": [

    "Action": "ecs:*",
    "Effect": "Allow",
    "Resource": [
                "acs:ecs:*:*:instance/i-001",
                "acs:ecs:*:*:instance/i-002"



    "Action": "ecs:Describe*",
    "Effect": "Allow",
    "Resource": "*"


"Version": "1"
```

Then, add the authorization policy for the user.

Authorize a RAM user to create snapshots

If a RAM user cannot create disk snapshots after being assigned the ECS administrator permissions, you must assign disk permissions to the user because snapshots are created based on disks.

Assume that you want to authorize the RAM user to manage the ECS instance whose ID is inst-01, and to create snapshots for the disk whose ID is dist-01. In this case, you can create the following authorization policy:

```
"Statement": [

    "Action": "ecs:*",
    "Effect": "Allow",
    "Resource": [
      "acs:ecs:*:*:instance/inst-01"



    "Action": "ecs:CreateSnapshot",
    "Effect": "Allow",
    "Resource": [
      "acs:ecs:*:*:disk/dist-01",
      "acs:ecs:*:*:snapshot/*"



    "Action": [
      "ecs:Describe*"

    "Effect": "Allow",
    "Resource": "*"


"Version": "1"
```

Then, add the authorization policy for the user.

2 Authorization for OSS instances

Questions

View OSS permission definitions

See *Access control* in the OSS product document.

Assign the OSS read-only permission to a RAM user

Create a RAM user in the RAM console and add the system authorization policy AliyunOSSReadOnlyAccess to the user. For more information about how to add an authorization policy, see *Authorization*.

Assign the full OSS management permission to a RAM user

Add the system authorization policy AliyunOSSFullAccess to the RAM user in the RAM console.

Authorize a RAM user to list and read resources in a bucket

If you need to authorize a RAM user (such as an application that represents you) to list and read the resources in a bucket using the OSS SDK or OSS CMD, you must create an authorization policy.Resource in, then you need to create a custom Authorization Policy to complete.

Assume that your bucket is named “myphotos”. Create the authorization policy as follows:

```
{
    "Version": "1",
    "Statement ":[
        {
            "Effect": "Allow",
            "Action": "oss:ListObjects",
            "Resource": "acs:oss:*:*:myphotos"
        },
        {
            "Effect": "Allow",
```

```
            "Action": "oss:GetObject",
            "Resource": "acs:oss:*:*:myphotos/*"
        }
    ]
}
```

If you want the authorized RAM-user to perform operations on the OSS console, add the GetBucketAcl and GetObjectAcl permissions to the authorization policy. (The console needs to call additional OSS APIs to optimize the operation experience.) The following provides an example of the authorization policy definition that allows the RAM-user to perform operations on the OSS console:

```
{
    "Version": "1",
    "Statement ":[
        {
            "Effect": "Allow",
            "Action": "oss:ListBuckets",
            "Resource": "acs:oss:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "oss:ListObjects",
                "oss:GetBucketAcl"
            ],
            "Resource": "acs:oss:*:*:myphotos"
        },
        {
            "Effect": "Allow",
            "Action": [
                "oss:GetObject",
                "oss:GetObjectAcl"
            ],
            "Resource": "acs:oss:*:*:myphotos/*"
        }
    ]
}
```

Apply IP address-specific access control in OSS

Example 1: Apply IP address-specific access control using the `Allow` command

The IP address segments `42.120.88.0/24` and `42.120.66.0/24` are allowed to read the information in the myphotos directory.

```
{
    "Version": "1",
    "Statement ":[
        {
            "Sid": "To allow listing all buckets",
            "Effect": "Allow",
            "Action": [
                "oss:ListBuckets"
            ],
            "Resource": [
```

```
                "acs:oss:*:*:*"
            ]
        },
        {
            "Sid": "To allow only the users in the specified IP
address segment to obtain the information in the myphotos directory",
            "Effect": "Allow",
            "Action": [
                "oss:ListObjects",
                "oss:GetObject"
            ],
            "Resource": [
                "acs:oss:*:*:myphotos",
                "acs:oss:*:*:myphotos/*"
            ],
            "Condition ":{
                "IpAddress": {
                    "acs:SourceIp": "42.120.88.0/24", "42.120.66.0/24"
                }
            }
        }
    ]
}
```

Example 2: Apply IP address-specific access control using the `Deny` command

If the IP address of a user is not within the `42.120.88.0/24` segment, the user cannot perform any OSS operations. Create an authorization policy as follows:

```
{
    "Version": "1",
    "Statement ":[
        {
            "Sid": "To allow listing all buckets",
            "Effect": "Allow",
            "Action": [
                "oss:ListBuckets"
            ],
            "Resource": [
                "acs:oss:*:*:*"
            ]
        },
        {
            "Sid": "To allow obtaining the information in the myphotos
 directory",
            "Effect": "Allow",
            "Action": [
                "oss:ListObjects",
                "oss:GetObject"
            ],
            "Resource": [
                "acs:oss:*:*:myphotos",
                "acs:oss:*:*:myphotos/*"
            ]
        },
        {
            "Sid": "To disallow IP addresses not in the 42.120.88.0/24
 segment to access OSS",
            "Effect": "Deny",
            "Action": "oss:*",
```

```
            "Resource": [
                "acs:oss:*:*:*"
            ],
            "Condition ":{
                "NotIpAddress": {
                    "acs:SourceIp": ["42.120.88.0/24"]
                }
            }
        }
    ]
}
```

NOTE: A policy with the Deny command has a higher priority than the policy with the Allow command. (If the accessing operation of a user meets any policy with the Deny command, the user is disallowed to access the content.) Therefore, when a user whose IP address is not in the 42.120.88.0/24 segment attempts to access the information in the “myphotos” directory, the OSS service notifies the user of having no operation permissions.

Authorization by OSS directory

Authorization by directory is an advanced authorization function.

Background

Assume that you have a photo bucket named “myphotos”. The bucket contains directories that indicate the places where the photos were taken. Each directory contains subdirectories that indicate the years when the photos were taken.

The directory tree is as follows:

```
myphotos[Bucket]
   ├── beijing
   │    ├── 2014
   │    └── 2015
   ├── hangzhou
   │  ├── 2013
   │    ├── 2014
   │  └── 2015 //The read-only permission on this directory must be
assigned.
   └── qingdao
        ├── 2014
        └── 2015
```

Assume that you need to assign the read-only permission on the `myphotos/hangzhou/2015/` directory to a RAM user. The required authorization policy depends on the application scenario. The following describes the authorization policies for three scenarios by policy complexity, from simplest to more complex.

Scenario 1: The RAM user knows all file paths, requires only the permission to read file content, and does not require the permission to list files.

In this scenario, the RAM user knows the complete paths of all files and can directly read the files using the complete paths. A software system requires such permission assignment, because the file paths in the software system comply with a certain rule (for example, files are named after employee IDs) or the file paths have persisted in the database of the software system.

```
{
    "Version": "1",
    "Statement ":[
        {
            "Effect": "Allow",
            "Action": [
                "oss:GetObject"
            ],
            "Resource": [
                "acs:oss:*:*:myphotos/hangzhou/2015/*"
            ]
        }
    ]
}
```

Scenario 2: A RAM user uses the OSS CMD to access the `myphotos/hangzhou/2015/` directory, but does not know what files are available in the directory. Therefore, the files must be listed.

Generally, software developers require such permission assignment. The developers do not know what files are available in a directory and use the OSS CMD or API to directly obtain the directory information.

In this scenario, the ListObjects permission that is not required in `scenario 1` must be added. Because only the files in the `myphotos/hangzhou/2015/` directory are to be listed, the oss:Prefix condition must be added to the ListObjects permission.

```
{
    "Version": "1",
    "Statement ":[
        {
            "Effect": "Allow",
            "Action": [
                "oss:GetObject"
            ],
            "Resource": [
                "acs:oss:*:*:myphotos/hangzhou/2015/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "oss:ListObjects"
            ],
            "Resource": [
                "acs:oss:*:*:myphotos"
            ],
            "Condition":{
```

```
                "StringLike":{
                    "oss:Prefix":"hangzhou/2015/"
                }
            }
        }
    ]
}
```

Scenario 3: A RAM user uses the OSS console to access the `myphotos/hangzhou/2015/` directory.

This is the most easy-to-use scenario. When the RAM user uses the visual OSS client to access the `myphotos/hangzhou/2015/` directory, like Windows File Explorer, the visual OSS client allows the RAM user to access the target directory from the root directory through levels of sub-directories.

Therefore, you need to add the following permissions to implement this type of directory navigation:

1. Permission to list all buckets
2. Permission to list the subdirectories of the “myphotos” directory (In this example, the subdirectories include beijing, hangzhou, and qingdao.)
3. Permission to list the subdirectories under “myphotos/hangzhou” (The subdirectories include 2013, 2014, and 2015.)

```
{
    "Version": "1",
    "Statement ":[
        {
            "Effect": "Allow",
            "Action": [
                "oss:ListBuckets",
                "oss:GetBucketAcl"
            ],
            "Resource": [
                "acs:oss:*:*:*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "oss:GetObject",
                "oss:GetObjectAcl"
            ],
            "Resource": [
                "acs:oss:*:*:myphotos/hangzhou/2015/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "oss:ListObjects"
            ],
```

```
            "Resource": [
                "acs:oss:*:*:myphotos"
            ],
            "Condition": {
                "StringLike": {
                    "oss:Delimiter": "/",
                    "oss:Prefix": [
                        "",
                        "hangzhou/",
                        "hangzhou/2015/*"
                    ]
                }
            }
        }
    ]
}
```

Authorize a RAM user complete management of a bucket

You need to create an authorization policy first. Assume that your bucket is named “myphotos”. Create the authorization policy as follows:

```
{
    "Version": "1",
    "Statement ":[
        {
            "Effect": "Allow",
            "Action": "oss:*",
            "Resource": [
                "acs:oss:*:*:myphotos",
                "acs:oss:*:*:myphotos/*"
            ]
        }
    ]
}
```

Then, add the authorization policy for this user.

RAM user authorized to manage a bucket notified of having no operation permissions when logging on to the OSS console

Assume that you create an authorization policy as follows to authorize a RAM user to read data objects from a bucket (such as “myphotos”):

```
{
  "Version": "1",
  "Statement ":[
    {
      "Effect": "Allow",
      "Action": [
        "oss:ListObjects"
      ],
      "Resource": "acs:oss:*:*:myphotos"
    },
    {
      "Effect": "Allow",
      "Action": [
```

```
        "oss:GetObject"
      ],
      "Resource": "acs:oss:*:*:myphotos/*"
    }
  ]
}
```

However, the RAM user was notified of having no operation permissions when logging on to the OSS console.

The reason is that when the RAM user logs on to the OSS console, the OSS console makes the RAM user access the OSS service as authorized. For a better user interaction experience, the OSS console also calls the ListBuckets, GetBucketAcl, and GetObjectAcl operations. (GetBucketAcl specifies whether a bucket is private or public. GetObjectAcl specifies an object is private or public.)

Therefore, to enable the RAM user to manage a bucket on the OSS console, you need to create the authorization policy as follows:

```
{
  "Version": "1",
  "Statement ":[
    {
      "Effect": "Allow",
      "Action": "oss:ListBuckets",
      "Resource": "acs:oss:*:*:*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "oss:ListObjects",
        "oss:GetBucketAcl"
      ],
      "Resource": "acs:oss:*:*:myphotos"
    },
    {
      "Effect": "Allow",
      "Action": [
        "oss:GetObject",
        "oss:GetObjectAcl"
      ],
      "Resource": "acs:oss:*:*:myphotos/*"
    }
  ]
}
```

3 Authorization for RDS instances

Questions

View RDS permission definitions

See RDS resource authorization.

Assign the RDS read-only permission to a RAM user

Create a RAM user in the RAM console and add the system authorization policy ”AliyunRDSReadOnlyAccess” to the user. For more information about how to add an authorization policy, see *Authorization*.

Assign full RDS service management permissions to a RAM user

Add the system authorization policy “AliyunRDSFullAccess” to the RAM user in the RAM console.

Authorize a RAM user to manage two specified RDS instances

You must use the function of customizing authorization policies. For example, you have two instances and the IDs are i-001 and i-002:

First, you must create a custom authorization policy that includes permissions for managing i-001 and i-002 and viewing all RDS resources:

```
"Statement": [

    "Action": "rds:*",
    "Effect": "Allow",
    "Resource": [
                "acs:rds:*:*:dbinstance/i-001",
                "acs:rds:*:*:dbinstance/i-002"



    "Action": "rds:Describe*",
    "Effect": "Allow",
    "Resource": "*"
```

```
  "Version": "1"
```

Then, add the custom authorization policy for this user.

Access the content of the DMS management database as a RAM user

Access ApsaraDB for RDS through DMS. The corresponding authorization action is “dms: LoginDatabase”.

Authorize the RAM user to log on to the specified RDS instance

Authorization policy example:

```
"Statement": [

    "Action": "dms:LoginDatabase",
    "Effect": "Allow",
    "Resource": "acs:rds:*:*:dbinstance/rds783a0639ks5k7328y"


"Version": "1"
```

Replace `rds783a0639ks5k7328y` with the ID of the RDS instance to be accessed.

Authorize the RAM user to log on to all RDS instances

Authorization policy example:

```
"Statement": [

    "Action": "dms:LoginDatabase",
    "Effect": "Allow",
    "Resource": "acs:rds:*:*:*"


"Version": "1"
```

4 Authorization for SLB instances

Questions

View SLB permission definitions

See *RAM authentication* in the SLB OpenAPI document.

Assign the SLB read-only permission to a RAM user

Create a RAM user in the RAM console and add the system authorization policy “AliyunSLBReadOnlyAccess” to the user. For more information about how to add an authorization policy, see *Authorization*.

Assign the SLB full access permission to a RAM user

Add the system authorization policy “AliyunSLBFullAccess” to the RAM user in the RAM console.

Authorize a RAM user to manage two specified SLB instances

You must use the function of customizing authorization policies. For example, you have two instances and the IDs are i-001 and i-002:

First, you must create a custom authorization policy that includes permissions for managing i-001 and i-002 and viewing all SLB resources:

```
"Statement": [

    "Effect": "Allow",
    "Action": "slb:*",
    "Resource": [
                "acs:slb:*:*:loadbalancer/i-001",
                "acs:slb:*:*:loadbalancer/i-002"



    "Effect": "Allow",
    "Action": "slb:Describe*",
    "Resource": "*"
```

```
  "Version": "1"
```

Then, add the authorization policy for this user.

A RAM user authorized to manage an SLB instance is notified of no operation permission when the user adds or removes ECS servers in the instance or sets weights

In the SLB, ECS server operation interfaces check not only the permissions for SLB resources , but also the permissions for ECS servers. This eliminates the situations in which a RAM user arbitrarily adds servers to an SLB instance after obtaining the permission for the instance.

For example, if you want to add the i-001 ECS server to the slb-001 SLB, you must grant the following permissions to your account:

```
"Statement": [

    "Effect": "Allow",
    "Action": "slb:AddBackendServers",
    "Resource": ["acs:slb:*:*:loadbalancer/slb-001"]


    "Effect": "Allow",
    "Action": "slb:AddBackendServers",
    "Resource": ["acs:ecs:*:*:instance/i-001"]


      "Effect": "Allow",
      "Action": "slb:DescribeLoadBalancers",
      "Resource": "acs:slb:*:*:loadbalancer/*"


"Version": "1"
```

You can make the authorization process more efficient so that you can grant management permissions for one SLB instance. This allows a user to add any servers to the instance and set the weight of any instances. See the following authorization policy. This authorization policy adds permissions for operations on all the SLB instances to the ECS resource.

```
"Statement": [

    "Effect": "Allow",
    "Action": "slb:*",
    "Resource": [
                "acs:slb:*:*:loadbalancer/i-001",
                "acs:slb:*:*:loadbalancer/i-002"



    "Effect": "Allow",
    "Action": "slb:Describe*",
    "Resource": "*"
```

```
    "Effect": "Allow",
    "Action": "slb:*",
    "Resource": "acs:ecs:*:*:*"


"Version": "1"
```

5 Authorization for CDN instances

Questions

Authorize a RAM user to perform the cache refresh and push operations

Authorize a RAM user to perform the cache refresh and push operations

You can create the following authorization policy for the user, which includes the permissions for reading content from CDN, refreshing the cache, and performing the push operation.

```
"Version": "1",
"Statement": [

    "Action": [
      "cdn:Describe*",
      "cdn:PushObjectCache",
      "cdn:RefreshObjectCaches"

    "Resource": "acs:cdn:*:*:*",
    "Effect": "Allow"
```

Then, assign the authorization policy to this user.

6 RAM user FAQ

How do I log on to the Alibaba Cloud console as a RAM user?

You can visit *https://signin-intl.aliyun.com/login.htm* or visit the RAM user logon URL on the right of the overview page in the *RAM console*.

The user name for logon can be in either of the following formats: <$username>@<$AccountAlias> and <$username>@<$AccountAlias>.onaliyun.com. If you have created a domain alias, you can also use the domain alias in <$username>@<$DomainAlias> format for logon.

Note:
When you log on to the Alibaba Cloud console by visiting the RAM user logon URL on the right of the overview page in the RAM console, the system automatically provides a default domain name. You only need to enter the user name.

What are the default domain name, account alias, and domain alias? How do I use and manage them?

For details about the **default domain name**, **account alias**, and **domain alias**, see *Terms*.

To view and manage the default domain name, account alias, and domain alias of your account, log on to the **RAM console** using the account or as a RAM user with the RAM permission, and choose **Identities** > **Settings** > **Advanced** > **Domain Alias**.

What permissions are required for a RAM user to purchase Alibaba Cloud products?

- For Pay-As-You-Go products, permission to create product instances, or similar permissions are required.
- For Subscription products, permission to create product instances and permission to make payments (the AliyunBSSOrderAccess policy) are required.
- For products that must be purchased with the use or creation of some other resources, the permission for reading or creating the corresponding resources is required. The following example describes the permissions required for creating an ECS instance.

  The following policy allows a RAM user to create an ECS instance through the console, the APIs, or the instance launch template.

```
{
  "Version": "1",
  "Statement": [
    {
```

```
      "Action": [
        "ecs:DescribeLaunchTemplates",
        "ecs:CreateInstance",
        "ecs:RunInstances",
        "ecs:DescribeInstances",
        "ecs:DescribeImages",
        "ecs:DescribeSecurityGroups"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "vpc:DescribeVpcs",
        "vpc:DescribeVSwitches"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

To allow a user to use or create other resources when the user is creating an ECS instance, grant the user the following permissions according to the resource types. Log on to the **RAM console** and click **Policies**. On the displayed page, create a custom policy and grant permissions as required to the user.

Operation	Policy action
Use a snapshot to create an ECS instance.	`ecs:DescribeSnapshots`
Create and use a VPC.	`vpc:CreateVpc` `vpc:CreateVSwitch`
Create and use a security group.	`ecs:CreateSecurityGroup` `ecs:AuthorizeSecurityGroup`

Specify the instance RAM role.	`ecs:DescribeInstanceRamRole`
	`ram:ListRoles`
	`ram:PassRole`

Use a key pair.	`ecs:CreateKeyPair` `ecs:DescribeKeyPairs`
Create an ECS instance on a Dedicated Host (DDH).	`ecs:AllocateDedicatedHosts`

After I grant permission to a user, why a message is displayed when the user accesses the system, indicating that the user does not have the permission?

- Check whether the policy attached to the user is correct.

- Check whether `"Effect": "Deny"` has been set in the custom policy (including policies of the user and policies of the user's groups) attached to the user for the corresponding resources or operations.

  For example, a user has both the AliyunECSReadOnlyAccess policy (which contains the read-only permission for accessing ECS) and the following policy:

  ```
  {
    "Statement": [
      {
        "Action": "ecs:*",
        "Effect": "Deny",
        "Resource": "*"
      }
    ],
    "Version": "1"
  }
  ```

  According to the "Deny takes priority" principle in RAM, the user is not allowed to view the ECS resources.

Why can a user perform operations without the corresponding permission?

For example, if a user that does not have the required custom policy or the FullAccess or ReadOnly system policy of ECS and can view ECS instances, perform the following operations:

1. Check whether the group policy of the user contains the permission that allows the user to perform the corresponding operations.
2. Check whether other polices attached to the user contain the corresponding permissions.

For example, the system policy of CloudMonitor is AliyunCloudMonitorFullAccess, which contains the following permissions: `"ecs:DescribeInstances"` (view ECS instances), `"rds:DescribeDBInstances"` (view RDS instances), and `"slb:DescribeLoadBalancer"` (view SLB instances). If you attach the AliyunCloudMonitorFullAccess policy to a user, the user has permission to view the information about the ECS, RDS, and SLB instances.

How do I grant permission to a user for renewal management only?

A unified renewal management policy is not currently available. You must customize a policy according to the specific products. You can grant the user the permission for purchasing the product and the payment permission.

For example, if you want a user to perform ECS renewal management, see *What permissions are required for a RAM user to purchase Alibaba Cloud products?* to grant required permissions and the AliyunBSSOrderAccess policy to the user.

7 STS FAQ

Why does an error occur when I use STS?

If the following error message is displayed, it means that the AliyunSTSAssumeRoleAccess policy is not attached to the authorized user:

```
Error message: You are not authorized to do this action. You should be
 authorized by RAM
```

Attach the AliyunSTSAssumeRoleAccess policy to the authorized user and then continue to use STS.

What permissions does an STS token have?

The permissions of an STS token are the specified role's permissions that are included in the policy set when the AssumeRole API is called.

If you do not set the policy parameter when calling the AssumeRole API, the returned STS token will have all the permissions of the specified role.

What is the validity period of an STS token?

The validity period of an STS token ranges from 900 seconds to 3600 seconds. The default value is 3600 seconds. You can set the DurationSeconds parameter when calling the AssumeRole API to limit the valid period of an STS token.

Is there an upper limit to the number of times that STS API can be called?

STS supports up to 100 Queries Per Second (QPS). If the call requests exceed 100 QPS, an error is reported.

If multiple STS tokens have been obtained at different times, are the old and new tokens valid at the same time?

Both the new and old STS tokens are valid before their expiration time.

8 Use tags to authorize ECS instances by group

This topic describes how to use tags to authorize ECS instances, security groups, disks, snapshots, and images by group.

FAQ

I have multiple ECS instances. If I want different users to see and manage only some of the instances, what should I do?

Solution

Suppose you have 10 ECS instances, and you want the dev team to manage 5 of them and the ops team to manage the other 5. You want each team to see only the authorized instances.

Authorize the ECS instances by group

You can implement this function with RAM. Perform the following operations:

1. Tag the ECS instances by group.

   For example, tag five of them with the key as team and the value as dev. Tag the other five with the key as team and the value as ops.

   To tag an instance, perform the following operations:

   1. In the ECS console, select an instance and choose **More** > **Instance Settings** > **Edit Tag** from the drop-down menu.
   2. Enter the key and value. For example, set the key to team and the value to dev.
2. Create two user groups, for example, named dev and ops. Create users for your employees and add the users to different user groups.
3. Create two custom policies and attach them to different user groups.

   In this example, for the dev user group, the policy content is as follows:

```
{
    "Statement": [
    {
        "Action": "ecs:*",
        "Effect": "Allow",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "ecs:tag/team": "dev"
            }
        }
    },
    {
        "Action": "ecs:DescribeTag*",
```

```
        "Effect": "Allow",
        "Resource": "*"
    }
    ],
    "Version": "1"
}
```

Note:

If your custom tags are different from the ones in the preceding example, the description of the tag conditions in the example must be replaced accordingly.

In the preceding policy, the `"Action": "ecs:*"` part with "Condition" is used to filter the instances tagged as `"team": "dev"`, and `"Action": "ecs:DescribeTag*"` is used to display all tags. When a user performs operations in the ECS console, the system displays all the tags for the user to select, and then filters the instances according to the tag key and value selected by the user.

Display authorized instances

1. Log on to the ECS console as a user of the dev user group.

   Note:

   - The users in the dev user group inherit the permissions of this group.
   - After a user logs on to the ECS console, the system navigates to the ECS overview page by default. In this case, the number of the ECS instances displayed on the page is 0.

2. Go to the instances page and check whether the region displayed in the console is the region where the instances are actually located. If no, select the expected region from the list of regions at the upper part of the console.
3. On the instances page, click **Tags**. The **Tag Key** drop-down list is displayed. Move the pointer over **Tag Key**. The **Tag Value** list is displayed. Select a value, and the system then filters the corresponding instances.

   Note:

   The system can filter the corresponding instances only after you select a value.

Use tags to authorize the security groups, disks, snapshots, and images by group

Follow the preceding method to tag and authorize the security groups, disks, snapshots, and images by group.

Note:

Only custom images can be tagged.

9 Use tags to authorize RDS instances by group

This topic answers common questions about the use of tags to authorize RDS instances. For details, see *Use tags to authorize ECS instances by group*.

How do I write a policy of using tags to authorize RDS instances by group?

A policy contains two parts. The `"Action": "rds:*"` part with "Condition" is used to filter the instances tagged as `"team": "dev"`, and the `"Action": "rds:DescribeTag*"` part is used to display all tags. The condition keyword is `rds:ResourceTag`.

```
{
  "Statement": [
    {
      "Action": "rds:*",
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "rds:ResourceTag/team": "dev"
        }
      }
    },
    {
       "Action": "rds:DescribeTag*",
       "Effect": "Allow",
       "Resource": "*"
     }
  ],
  "Version": "1"
}
```

After I use a tag to authorize an RDS instance, what should I do if an error is reported when a user logs on to the RDS console, indicating that the user does not have the permission?

1. Contact the account owner or the system administrator to confirm that the tag has been correctly attached to the RDS instance, and the policy is using the correct tag key and value in the instance.

   Note:
   The tag key and value of an RDS instance cannot contain uppercase letters. If you enter an uppercase letter, it is automatically converted to a lowercase one when you save it.

2. Contact the account owner or the system administrator to confirm that an expected policy has been attached to the user logging on to the RDS console.

3. After a user with the expected policy logs on to the RDS console, if the console displays a message "You do not have permission to perform this operation", it is because the console displays all resources by default, but the user does not have the permission to view all the resources. In this case, close the message window.
4. On the bar at upper part of the console, select the expected region.
5. On the instances page, click **Tag**. The **Tag Key** drop-down list is displayed. Move the pointer over **Tag Key**. The **Tag Value** list is displayed. Select a value, and the system filters the corresponding instances.

Note:

The system can filter the corresponding instances only after you select a value.