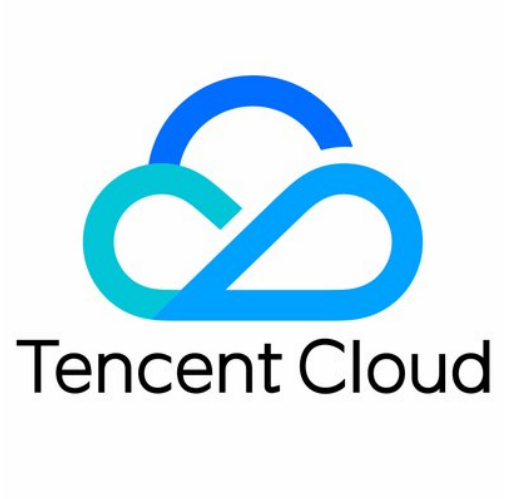
Cloud Access Management

Practical Tutorial

Product Documentation



Contents

Practical Tutorial
Security Practical Tutorial

Last updated：2024-06-28 14:57:30

Basic Principles

1. Enable MFA protection

To strengthen account security, we recommend that you bind MFA for all accounts. We also recommend enabling login and operation protection for root accounts and sub-accounts. For the accounts that support login with e-mail, we strongly recommend enabling MFA secondary verification. This will require a secondary verification for account login and sensitive operations. For related settings, see Setting Security Protection for Collaborators, and Setting Security Protection for Sub-Users.

2. Access Tencent Cloud with a sub-account

Do not use the root account identity credentials to access Tencent Cloud, and **never** share identity credentials with anyone. Create a sub-account for all users that access Tencent Cloud, and grant management permissions as necessary. For information about the related settings, see User Types.

3. Use groups to grant permissions

Define groups according to the job responsibilities, and grant management permissions to the group as necessary. Then, assign the users to the corresponding groups. In this way, when you modify the permissions for the group, the permissions of the users associated with the group will change accordingly. Additionally, when there are organizational changes and people move around, you only need to update the group the user belongs to. For more information, see User Groups.

4. Grant least privilege

Granting least privilege is a standard security principle where you grant only the permissions required to perform a task. Any additional unnecessary permissions should not be granted. For example, if a user only uses CDN Service, access permission for other services (such as COS read and write permissions) should not be granted.

5. Manage users, permissions, and resources with different sub-accounts

We do not recommend managing users, permissions, and resources with the same account. Designate different sub-accounts to manage users, permissions and resources respectively.

6. Rotate credentials regularly

We recommend you or one of your CAM users change the login password or API key regularly. This way, if one of your credentials is compromised, the time it can be used to access your resources is limited.
For information about setting passwords for root accounts, see Account Password.
For more information about setting passwords for sub-accounts, see Resetting Login Passwords for Sub-Users.

7. Delete unnecessary certificates and permissions

Delete certificates that the user does not need, and permissions that the user no longer needs. Minimize the security risks caused by compromised access credentials.

8. Use policy conditions to enhance security

Define the conditions under which your policies will take effect as precisely as possible to limit access and strengthen security. For example, write conditions to specify the server users must perform operations on. The time period can also be specified.
For more information, see Element References - Condition.

Multi-Identity Personnel Permission Management

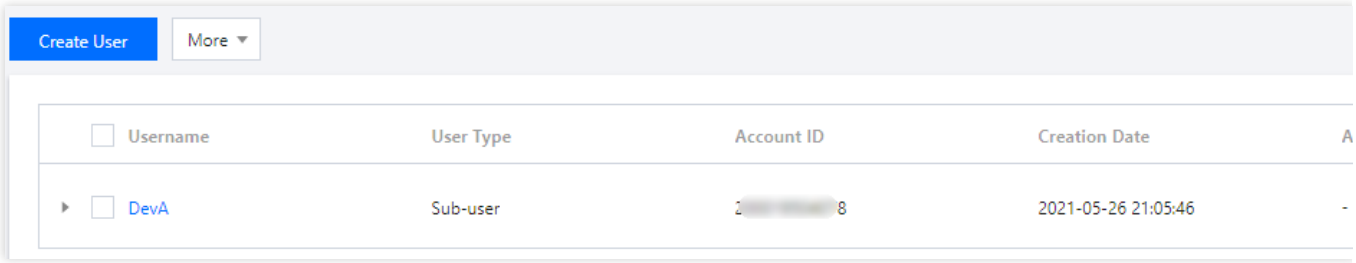
Last updated：2024-01-23 17:59:15

Overview

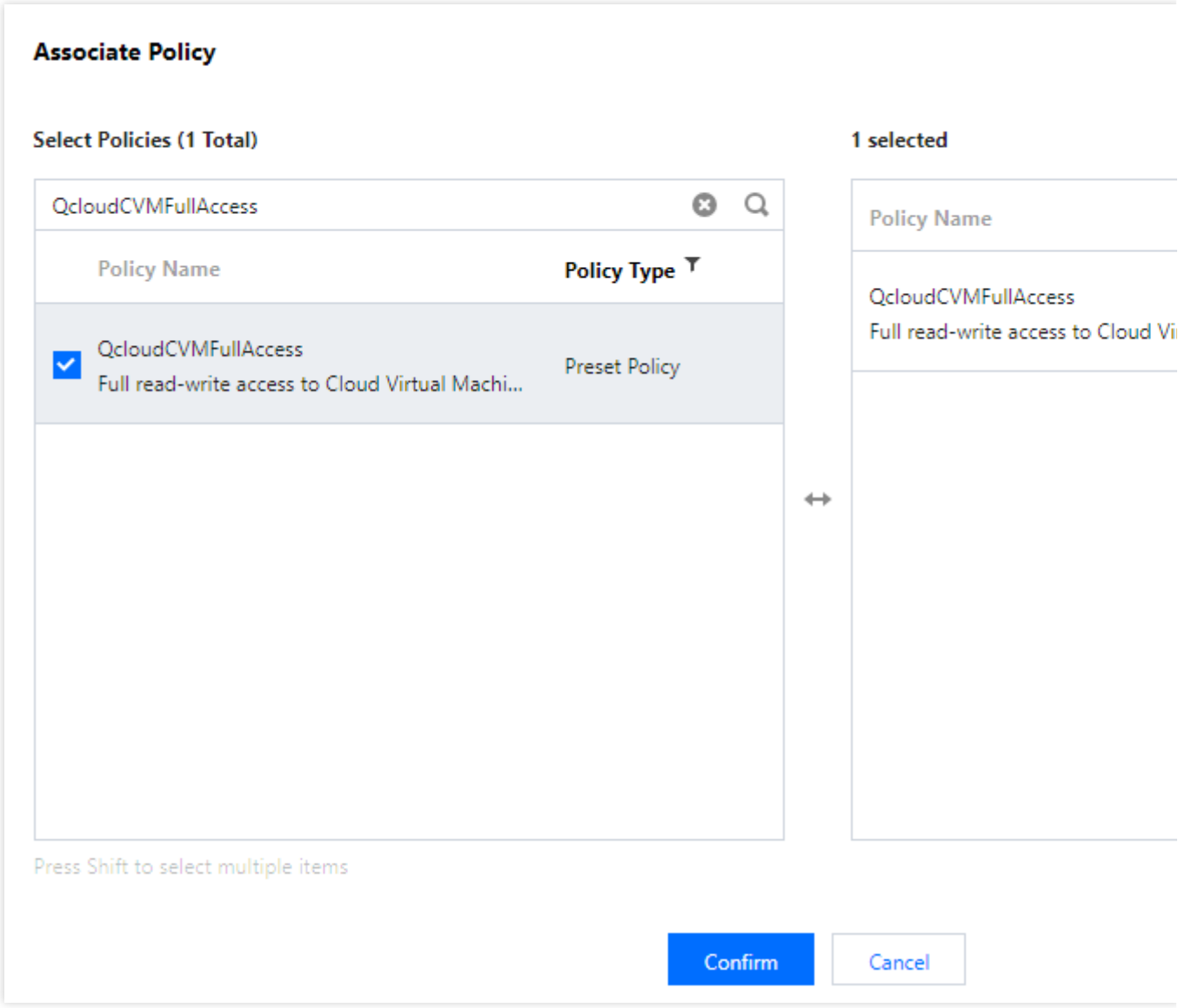
If your company has management personnel with different identities, you can use CAM to divide permissions and grant different permissions to different people to facilitate management and control. This document uses a typical case to describe how to manage the permissions of different identities through sub-accounts.
Suppose that:
The company account `CompanyExample` has two OPS engineers `DevA` and `DevB` .
The OPS engineer `DevA` is responsible for server OPS and has all operation permissions of CVM instances under the company account `CompanyExample` .
The OPS engineer `DevB` is responsible for TencentDB for MySQL OPS and has all operation permissions of TencentDB for MySQL instances under the company account `CompanyExample` .

Directions

1. Log in to the CAM console with the company account `CompanyExample` .
2. Create two sub-accounts with usernames of `DevA` and `DevB` through custom sub-user creation.
3. On the **User List** page, find the just created sub-user `DevA` and click **Authorize** in the **Operation** column on the right as shown below:



4. In the **Associate Policy** window that pops up, search for and select `QcloudCVMFullAccess` and click **OK** as shown below:



5. Associate the `QcloudCDBFullAccess` policy with the sub-account `DevB` as instructed in steps 2 and 3.
6. After the authorization is successful, the sub-account `DevA` has all the operation permissions of CVM instances, while the sub-account `DevB` has all the operation permissions of TencentDB for MySQL instances.

Note:

If you need to configure a CAM user as another role, you can follow the above process and search for and select the corresponding permissions policy name in steps 2 and 3. For specific permissions, please see System Permissions.

System Permissions

Owner	Policy Name	Description
Admin	AdministratorAccess	This policy allows you to manage all users and their permissions, related financial info, and cloud service assets

		under this account.
Financial admin	QCloudFinanceFullAccess	This policy allows you to manage related financial information under the account, such as payment and invoicing.
Database admin	QcloudCynosDBFullAccess	Full access to TDSQL-C
	QcloudMariaDBFullAccess	Full access to TencentDB for MariaDB
	QcloudSQLServerFullAccess	Full access to TencentDB for SQL Server
	QcloudCDWPGFullAccess	Full access to TencentDB for PostgreSQL
Network admin	QcloudCLBFullAccess	Full access to CLB
	QcloudVPCFullAccess	Full access to VPC
	QcloudDCFullAccess	Full access to Direct Connect
Monitoring admin	QcloudMonitorFullAccess	Full access to Cloud Monitor, including the permission to view user groups
	QcloudCATFullAccess	Full access to CAT

Authorizing Certain Operations by Tag

Last updated：2024-01-23 17:59:15

Overview

If you have purchased multiple types of Tencent Cloud resources which are grouped and managed by tag, you can grant employees of different teams permissions to use corresponding APIs by tag on an as-needed basis. This document uses a typical case to describe how to grant sub-accounts certain operation permissions of resources through tags.

Suppose that:

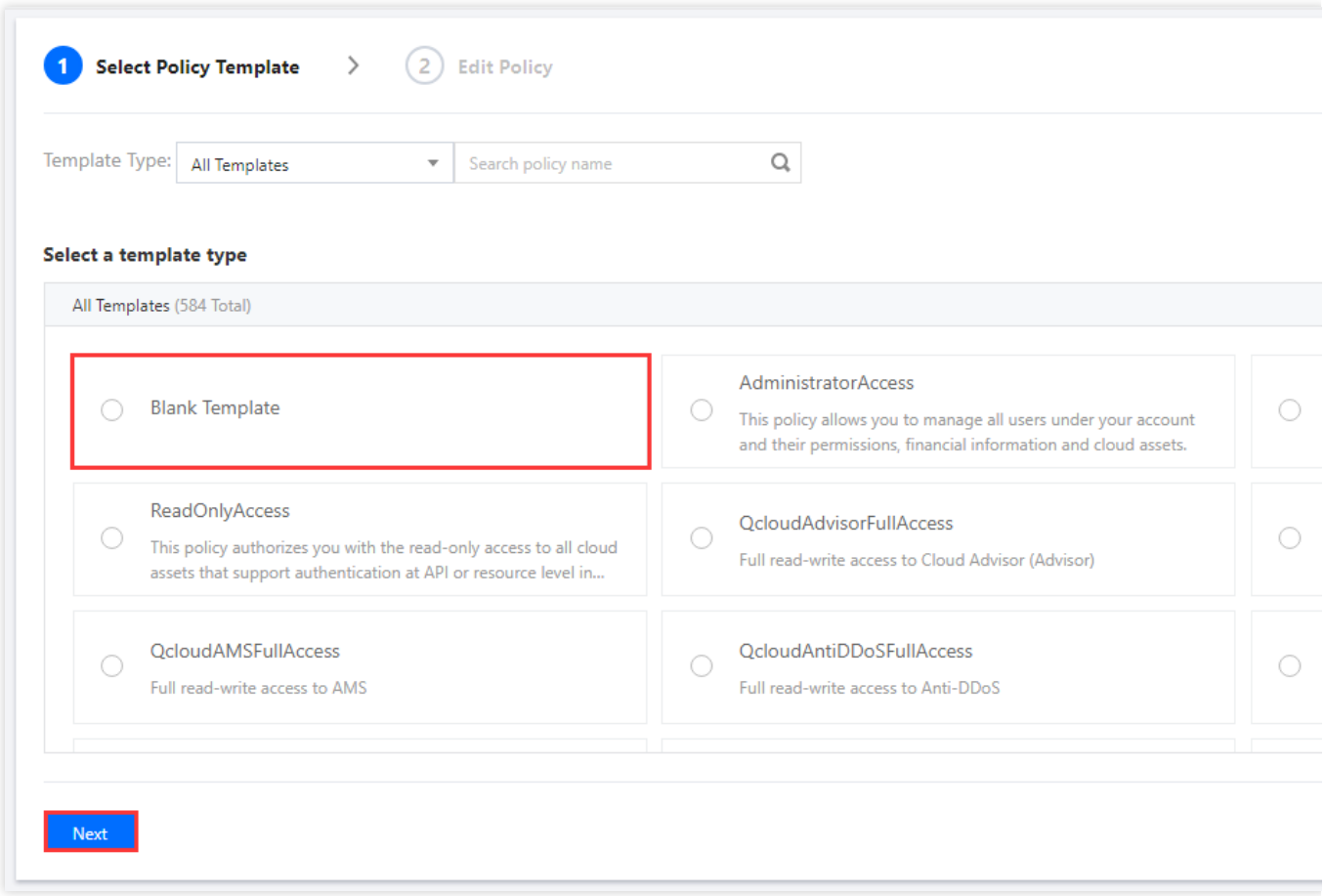
The company account `CompanyExample` has a sub-account `DevA`.

The company account `CompanyExample` has a tag key-value pair `test1&test1`.

The company account `CompanyExample` wants to grant the sub-account `DevA` the permission to restart CVM instances (cvm:RebootInstances) under the tag `test1&test1`.

Directions

1. Log in to the CAM console with the company account `CompanyExample`.
2. On the **Policy** page, click **Create Policy** > **Create by Policy Syntax**.
3. Select **Blank Template** under **Select a template type** and click **Next** to enter the **Edit Policy** page

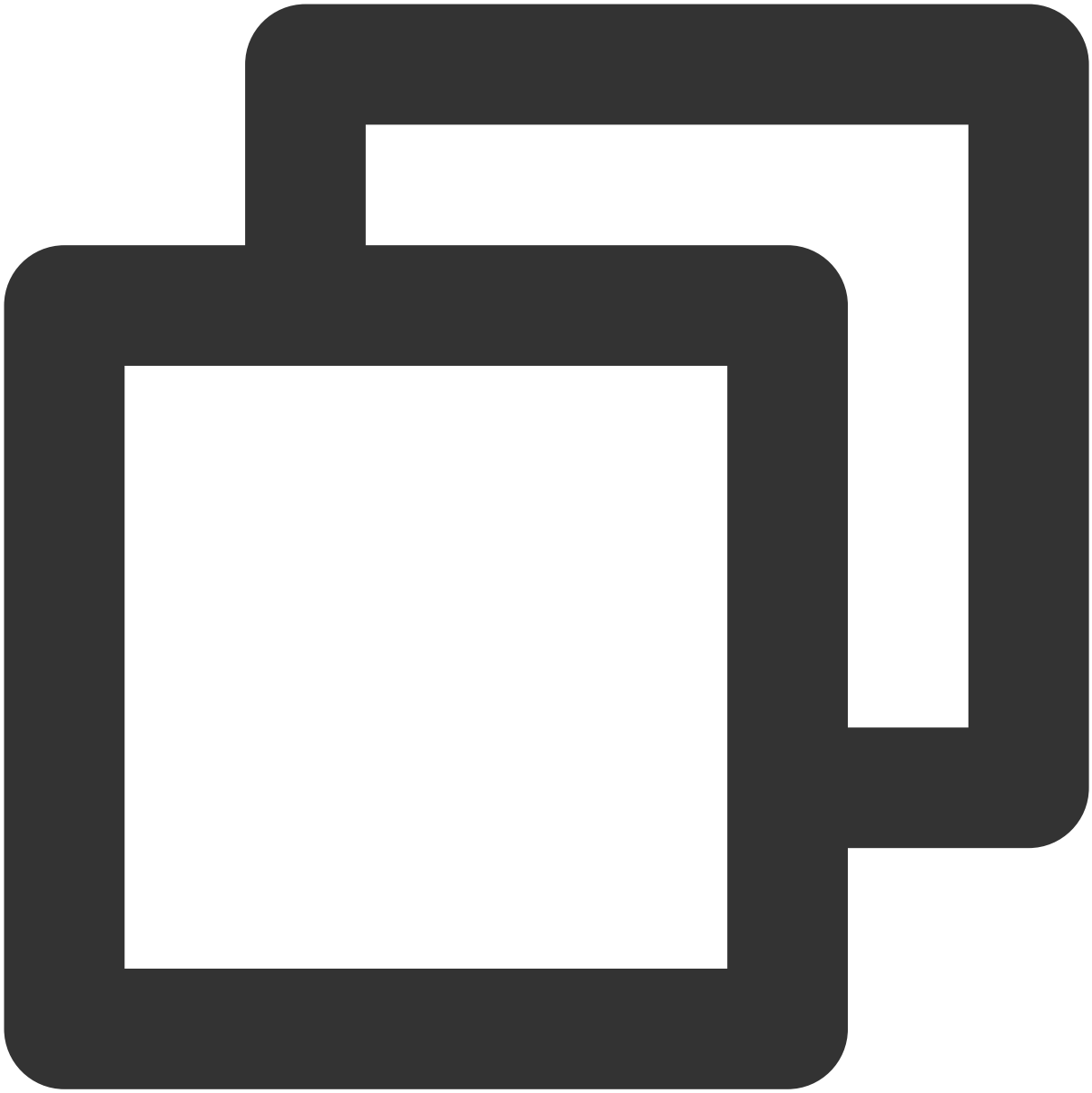


4. On the **Edit Policy** page, fill out the following form:

Policy Name: the default value is `policygen-current date` . We recommend you define a unique and meaningful policy name, such as `cvm-RebootInstances` .

Description: write a description, which is optional.

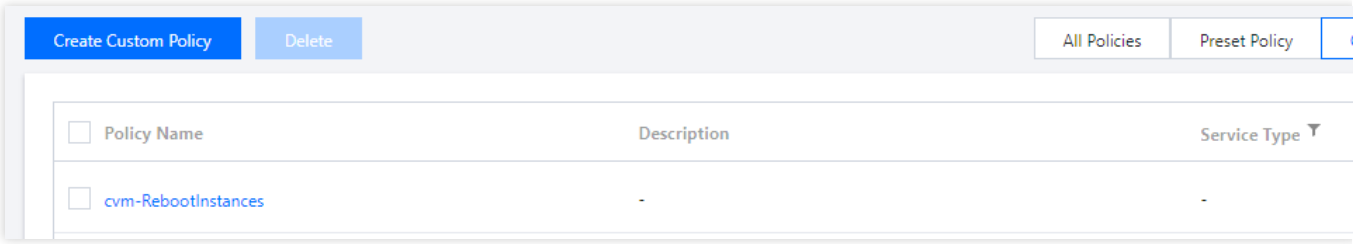
Policy Content: copy and paste the following content. Here, `cvm:RebootInstances` is the name of the API that needs to be authorized, and `test1&test1` is the tag key-value pair that needs to be authorized.



```
{
"version": "2.0",
"statement": [
    {
        "effect": "allow",
        "action": [
            "cvm:RebootInstances"
        ],
        "resource": "*",
        "condition": {
            "for_any_value:string_equal": {
```

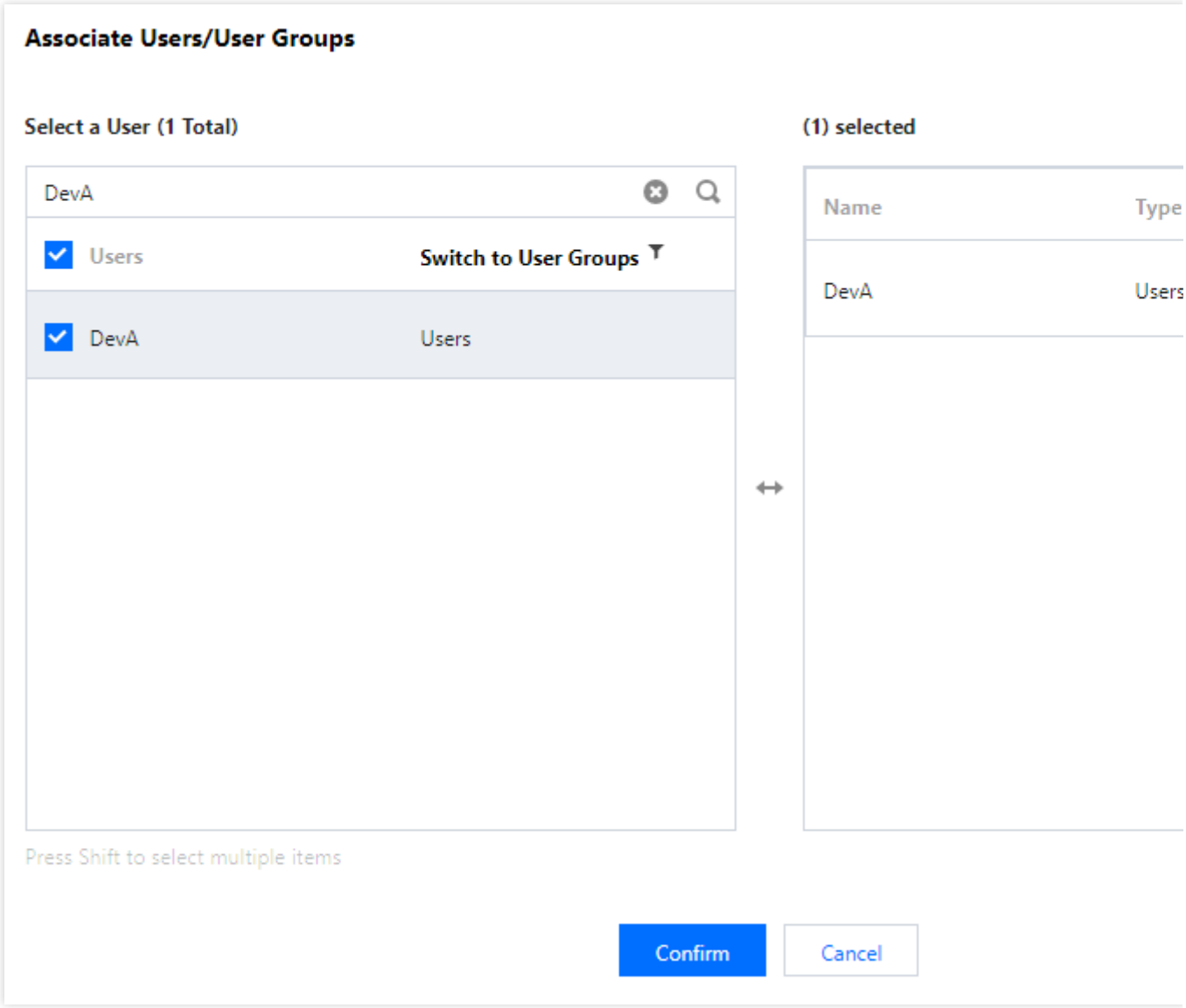
```
                "qcs:tag": [
                    "test1&test1"
                ]
            }
        }
    }
]
}
```

5. Click **Complete** to create the policy, which will be displayed on the **Policy List** page.
6. Find the just created policy in the Policy List and click **Associate** in the **Operation** column on the right.



7. In the **Associate with User/User Group** window that pops up, search for and select the sub-account `DevA` and click **OK** to complete the authorization.

The sub-account `DevA` will have the permission to restart CVM instances under the `test1&test1` tag.



Related Documents

If you want to know how to associate resources with tags, please see Managing Tags.

If you want to know how to grant all operation permissions of the resources under a tag, please see Authorizing Different Sub-accounts Separate Permissions to Manage Tencent Cloud Resources.

Supporting Isolated Resource Access for Employees

Overview

Last updated：2024-01-23 17:59:15

If your root account has multiple businesses and each business has its own resources, you may want employees from different businesses to be able to see and manipulate different resources when logging in with their CAM sub-accounts.
In this case, you can use two permission setting options in CAM to implement isolated resource access: authorization by resource ID or by tag.

Use Case

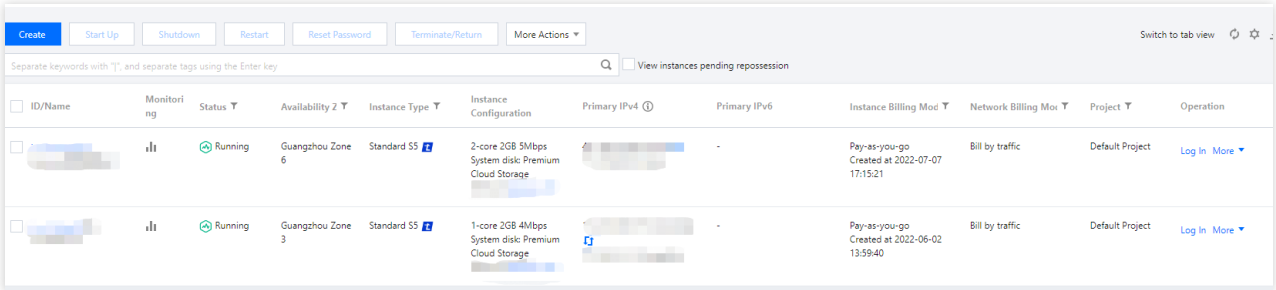
Taking CVM as an example, suppose there are two CVM instances as detailed below:

Resource ID	Image ID	Tag	Project
ins-duglsqg0	img-eb30mz89	game:webpage	webpage
ins-ijp192hy	img-eb30mz89	game:app	app

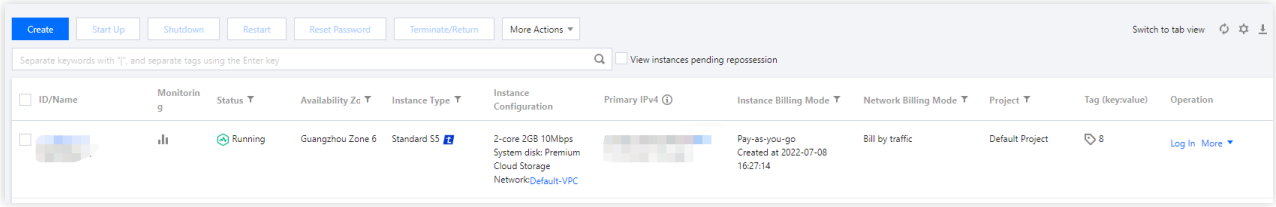
Create a CAM sub-user `cvmtest01` for an employee and use the above two permission setting options to allow `cvmtest01` to only view and access `ins-duglsqg0` .

Expected Result

The list of CVM instances in Guangzhou region viewed by the admin account:



The list of CVM instances in Guangzhou region viewed by `cvmtest01` :



Options

Option 1: Authorization by resource ID

Option 2: Authorization by tag

Authorization by Resource ID

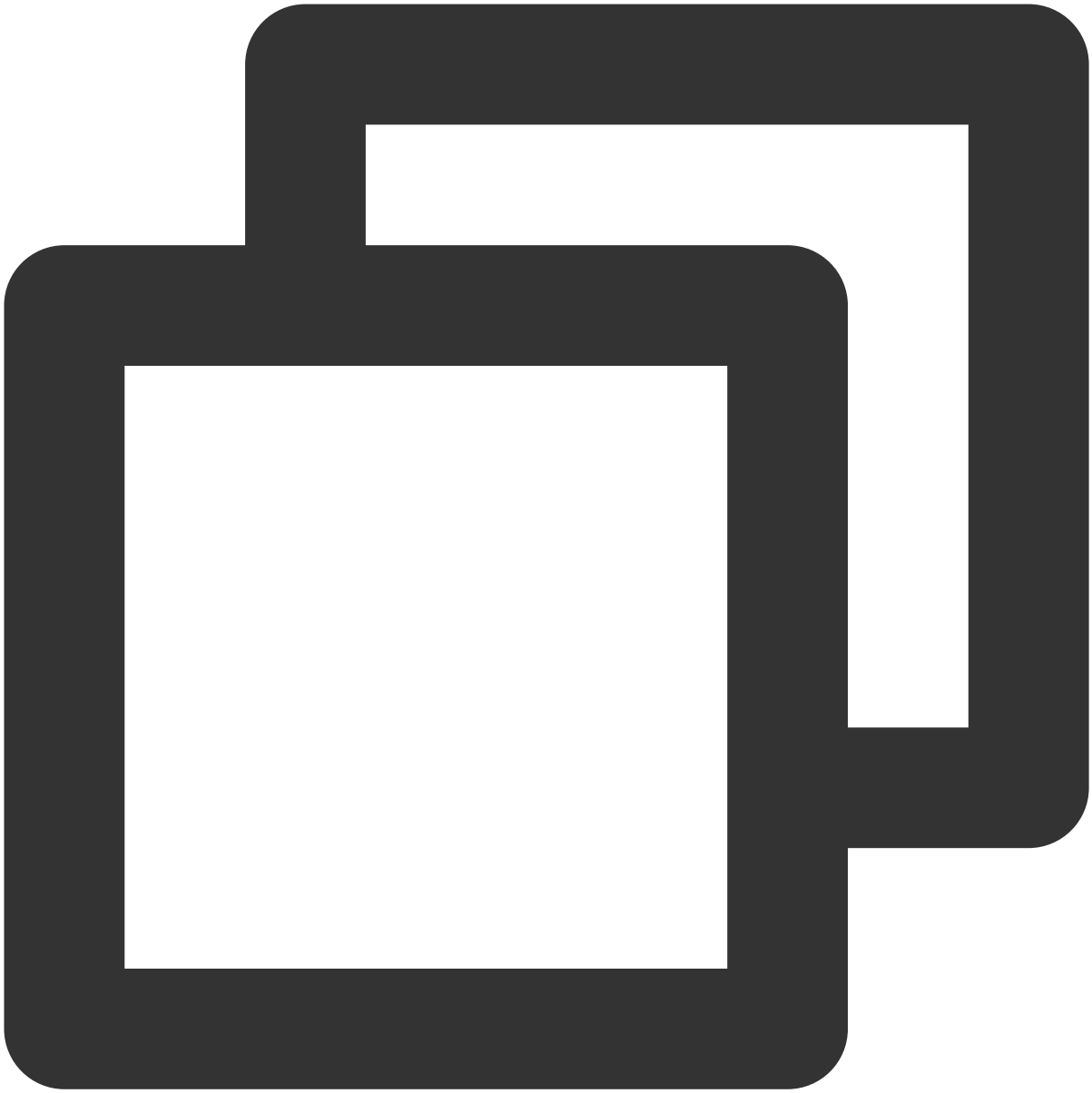
Last updated：2024-01-23 17:59:15

Overview

This document describes how to grant permissions by resource ID to allow the sub-user `cvmtest01` only to manage the resource-level APIs of `ins-duglsqg0` .
For more information, see overview >>

Policy Content

To grant permissions by resource ID to implement the above need, use the following policy content:



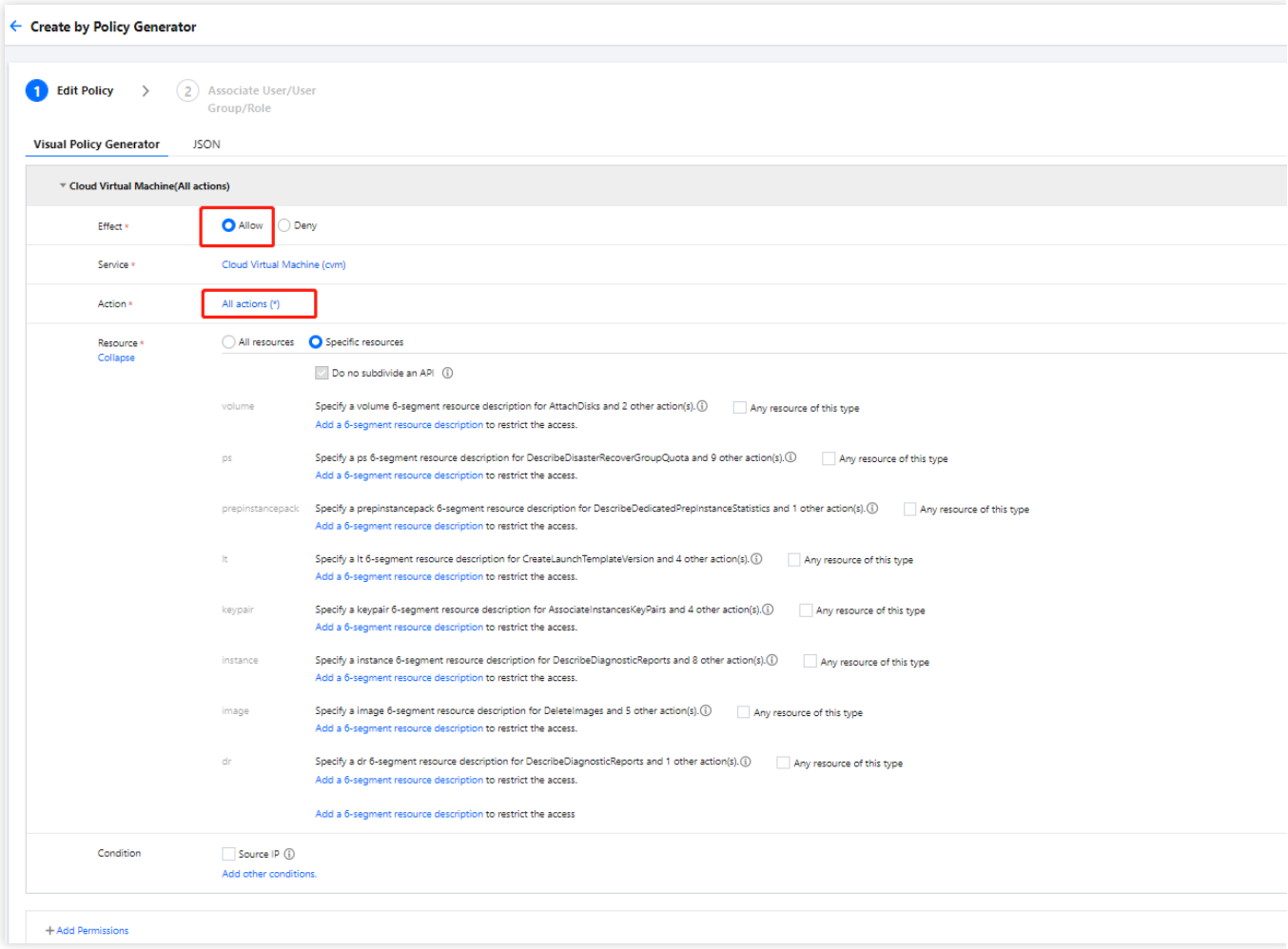
```
{
    "version": "2.0",
    "statement": [
        {
            "effect": "allow",
            "action": [
                "cvm:*"
            ],
            "resource": [
                "qcs::cvm::uin/12345678:instance/ins-duglsqg0",// `12345678` is `UI
                "qcs::cvm::uin/12345678:image/img-eb30mz89"
```

```
            ]
        },
        {
            "effect": "allow",
            "action": [
                "vpc:DescribeVpcEx",
                "vpc:DescribeNetworkInterfaces",
                "cvm:DescribeCbsStorages"
            ],
            "resource": [
                "*"
            ]
        }
    ]
}
```

Directions

Step 1. Use the admin account to create a policy and configure permissions

1. Log in to the CAM console with the admin account. On the Policy page, create a custom policy with the policy generator as instructed in Creating Custom Policy > Creating by policy generator.



Effect: Allowed

Service: CVM

Operation: All

Resource: Specific Resources > Add a custom six-segment resource description

Enter the resource prefixes `instance` and `image` and resource IDs `ins-duglsqg0` and `img-eb30mz89` respectively.

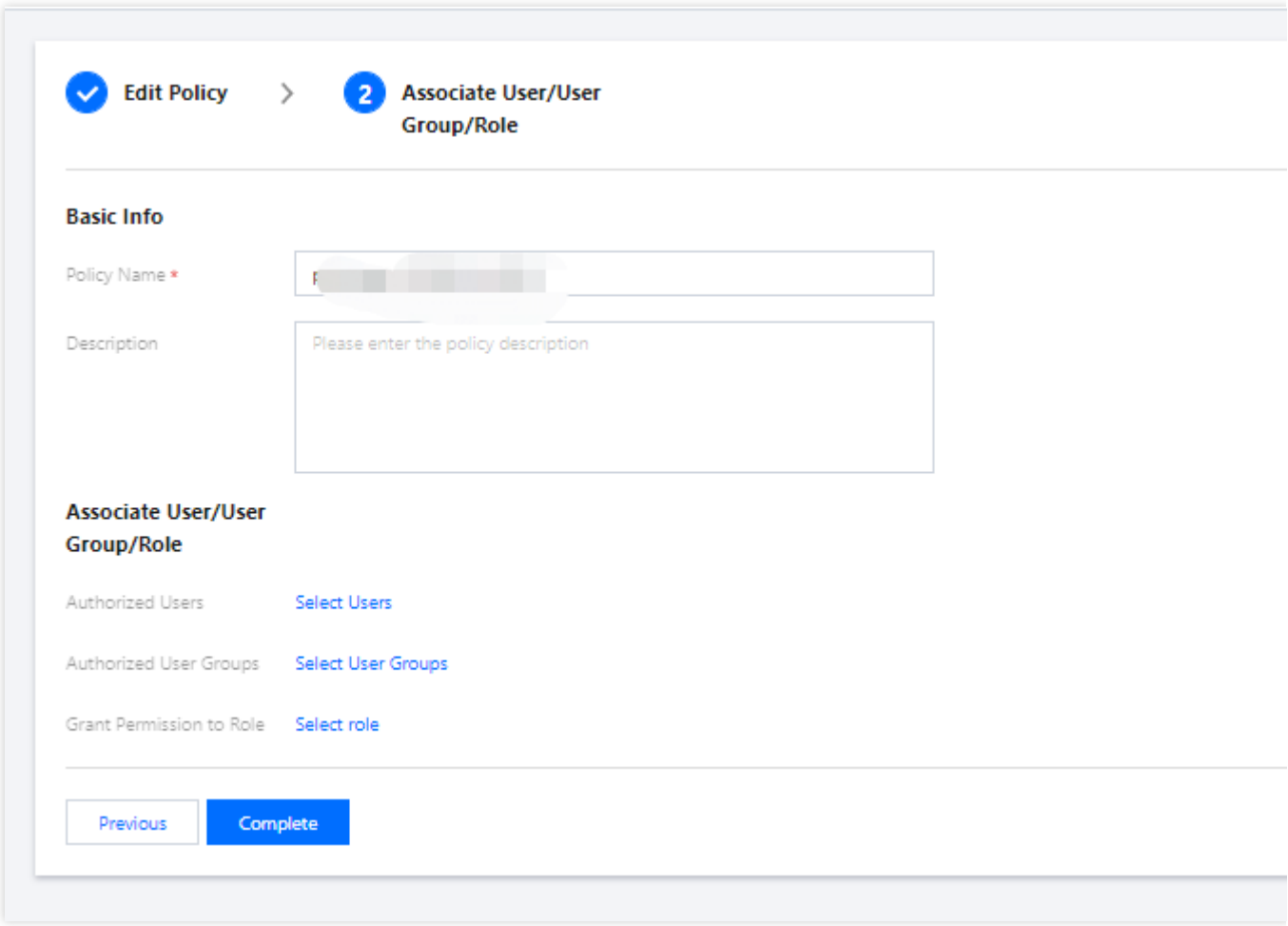
Note:

How to determine the resource prefix: You can view the CVM six-segment resource description in CAM APIs supported by CVM.

In addition to CVM APIs, APIs of other Tencent Cloud products such as VPC will also be used on the CVM product page. In this example, you can skip them and directly generate the policy. However, during actual operations, you need to add such APIs as prompted in CAM.

2. Click **Next**, name the policy `cvm-test01` , and grant it to the sub-account `cvmtest01` .

3. Click **Complete**.

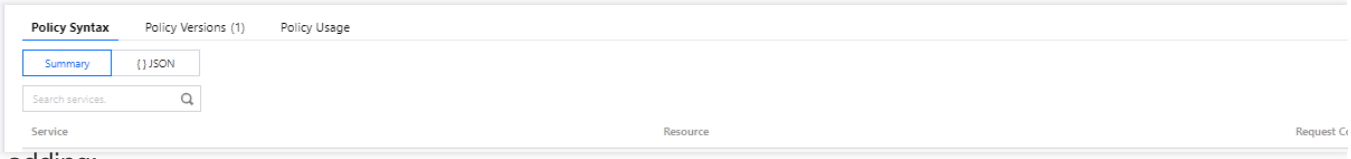


Step 2. Use the sub-account to log in and verify permissions

1. Log in to the CVM console with the sub-user account and enter the instance list page. The page prompts that `DescribeVpcEx` and relevant resource permissions of VPC are missing.
2. Contact the admin account to add such permissions to the policy as prompted.

Step 3. Use the admin account to adjust the policy content

1. Use the root account to find the `DescribeVpcEx` API in the list of CAM APIs supported by VPC and verify that the API is at the operation level.
2. On the Policy page in the CAM console, find the `cvm-test01` policy and click its name to enter the policy details page.
3. In the policy syntax, click **Edit** and add API authorization to the policy details in the format of operation-level API authorization.



Before adding:

```
{
    "version": "2.0",
    "statement": [
        {
            "effect": "allow",
            "action": [
                "cvm:*"
            ],
            "resource": [
                "qcs::cvm::uin/[redacted]:instance/ins-duglsqg0",
                "qcs::cvm::uin/[redacted]:image/img-eb30mz89"
            ]
        }
    ]
}
```

After adding:

```
        {
            "effect": "allow",
            "action": [
                "cvm:*"
            ],
            "resource": [
                "qcs::cvm::uin/[redacted]:instance/ins-duglsqg0",
                "qcs::cvm::uin/[redacted]:image/img-eb30mz89"
            ]
        },
        {
            "effect": "allow",
            "action": [
                "vpc:DescribeVpcEx"
            ],
            "resource": [
                "*"
            ]
        }
    ]
}
```

4. Repeat step 2 to use the sub-account `cvmtest01` to verify permissions again, and you can see that `DescribeNetworkInterfaces` and relevant resource access permissions of VPC are still missing. View the list of CAM APIs supported by VPC and verify that the `DescribeNetworkInterfaces` API is at the operation level.

5. Repeat step 3 to adjust the policy content until the system no longer reports errors.

The eventual policy content is as follows:

```
            "effect": "allow",
            "action": [
                "cvm:*"
            ],
            "resource": [
                "qcs::cvm::uin/[redacted]:instance/ins-duglsqg0",
                "qcs::cvm::uin/[redacted]:image/img-eb30mz89"
            ]
        },
        {
            "effect": "allow",
            "action": [
                "vpc:DescribeVpcEx",
                "vpc:DescribeNetworkInterfaces",
                "cvm:DescribeCbsStorages"
            ],
            "resource": [
                "*"
            ]
        }
    ]
}
```

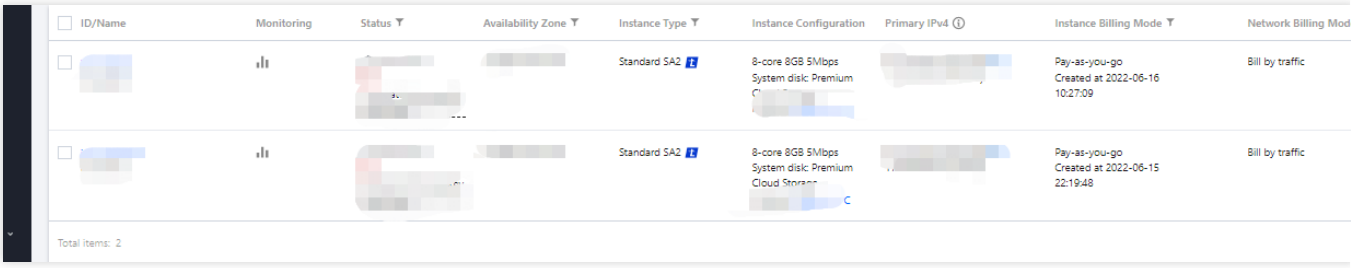
Note:

When writing a CAM policy, if you want to manipulate a specific resource, you need to separate the resource-level API authorization from operation-level API authorization, but you can put multiple operation-level APIs together.

Step 4. Verify the result

Use the sub-user `cvmtest01` to verify the policy again, and the expected effect is achieved.

At this point, the sub-user `cvmtest01` can start, shut down, restart, rename, and reset the password of the CVM instance.



Authorization by Tag

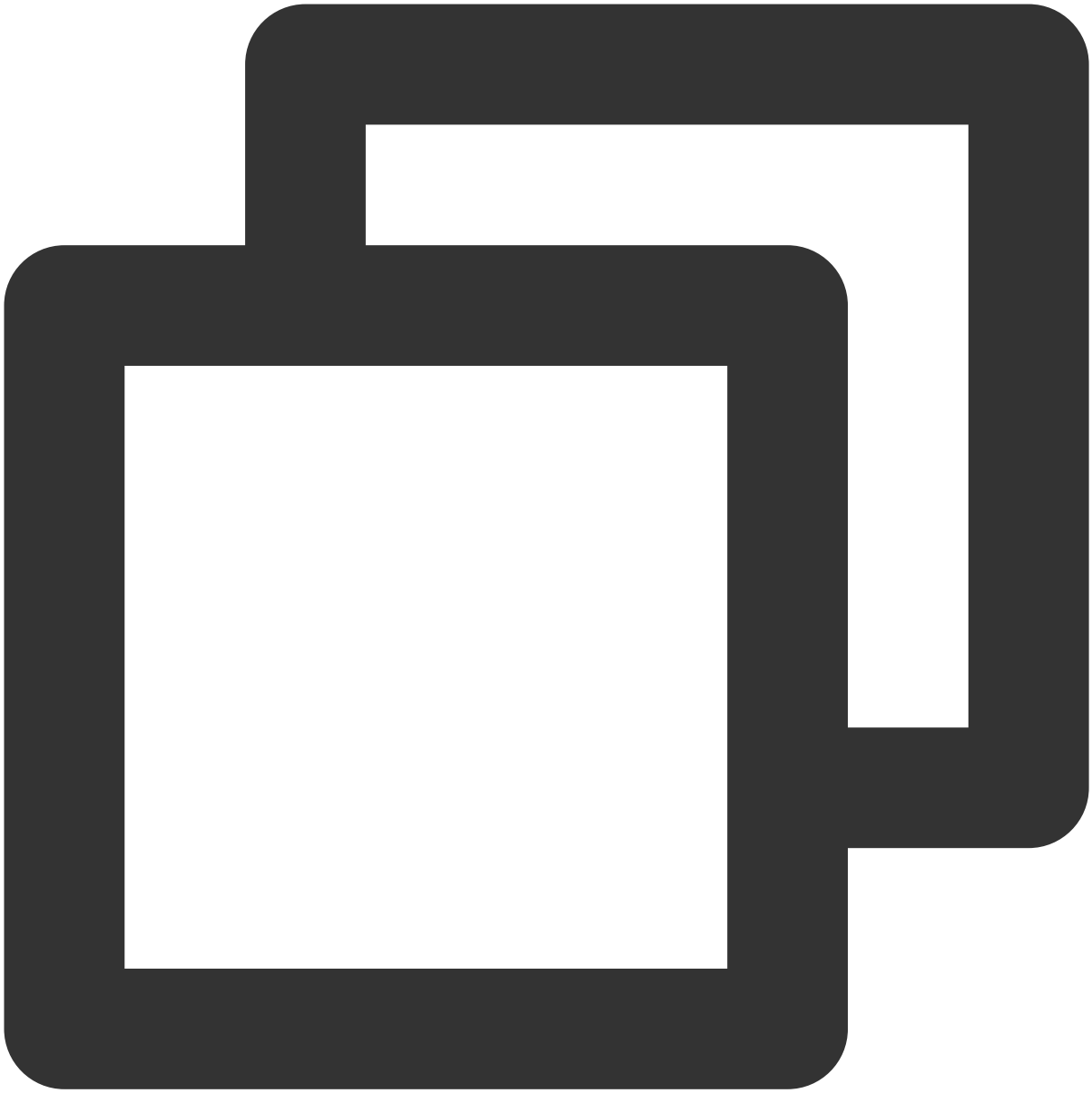
Last updated：2024-01-23 17:59:15

Overview

This document describes how to grant permissions by tag to allow the sub-user `cvmtest01` only to manage the resource-level API permissions of `ins-duglsqg0` .
For details, see Overview.

Policy Content

To grant permissions by tag as needed, you can use the following policy content:



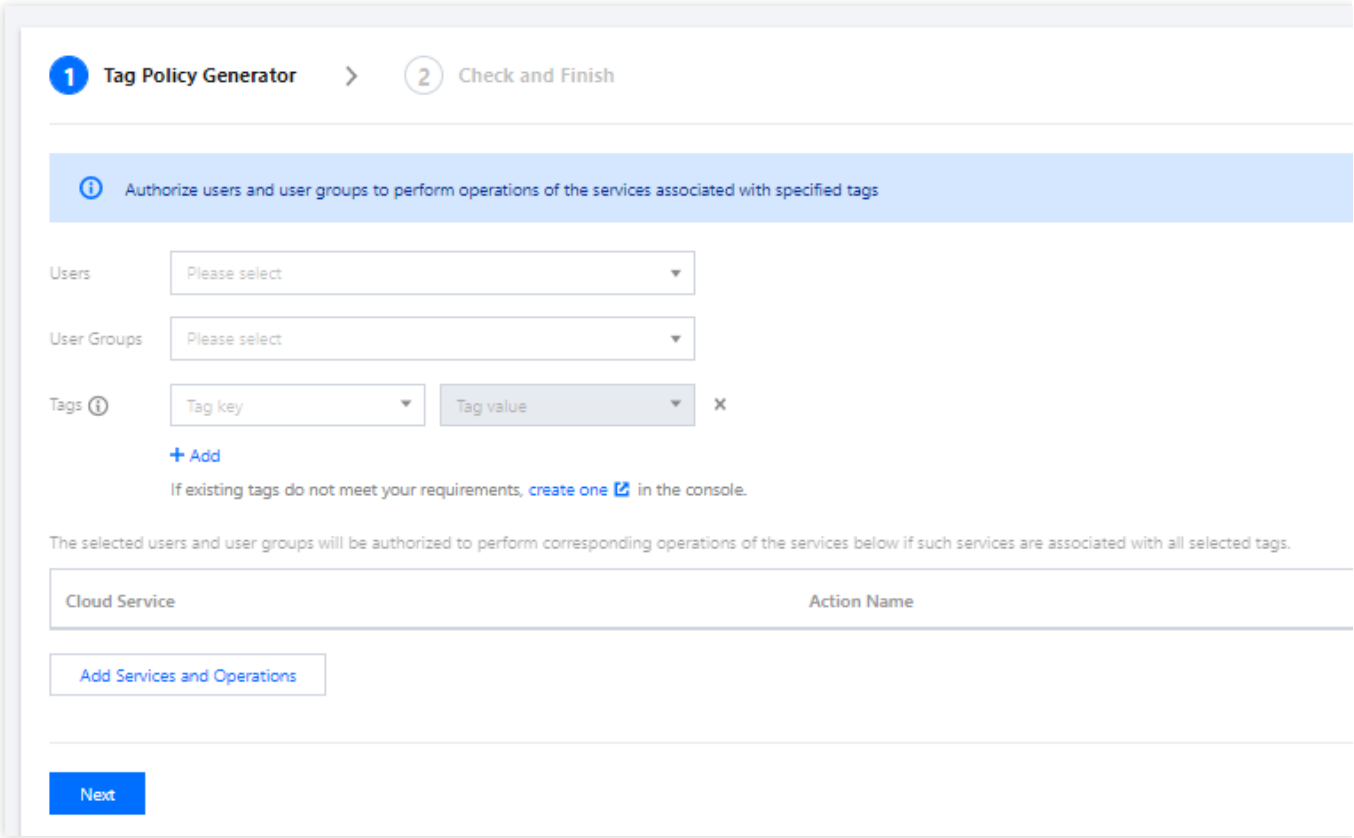
```
{
    "version": "2.0",
    "statement": [
        {
            "effect": "allow",
            "action": [
                "cvm:*",
                "vpc:DescribeVpcEx",
                "vpc:DescribeNetworkInterfaces"
            ],
            "resource": "*",
```

```
            "condition": {
                "for_any_value:string_equal": {
                    "qcs:resource_tag": [
                        "game&webpage"
                    ]
                }
            }
        }
    ]
}
```

Directions

Step 1. Create a policy and configure permissions

1. Log in to the CAM console with the admin account. On the Policies page, create a custom policy by tag as instructed in Creating Custom Policy > Authorizing by tag.



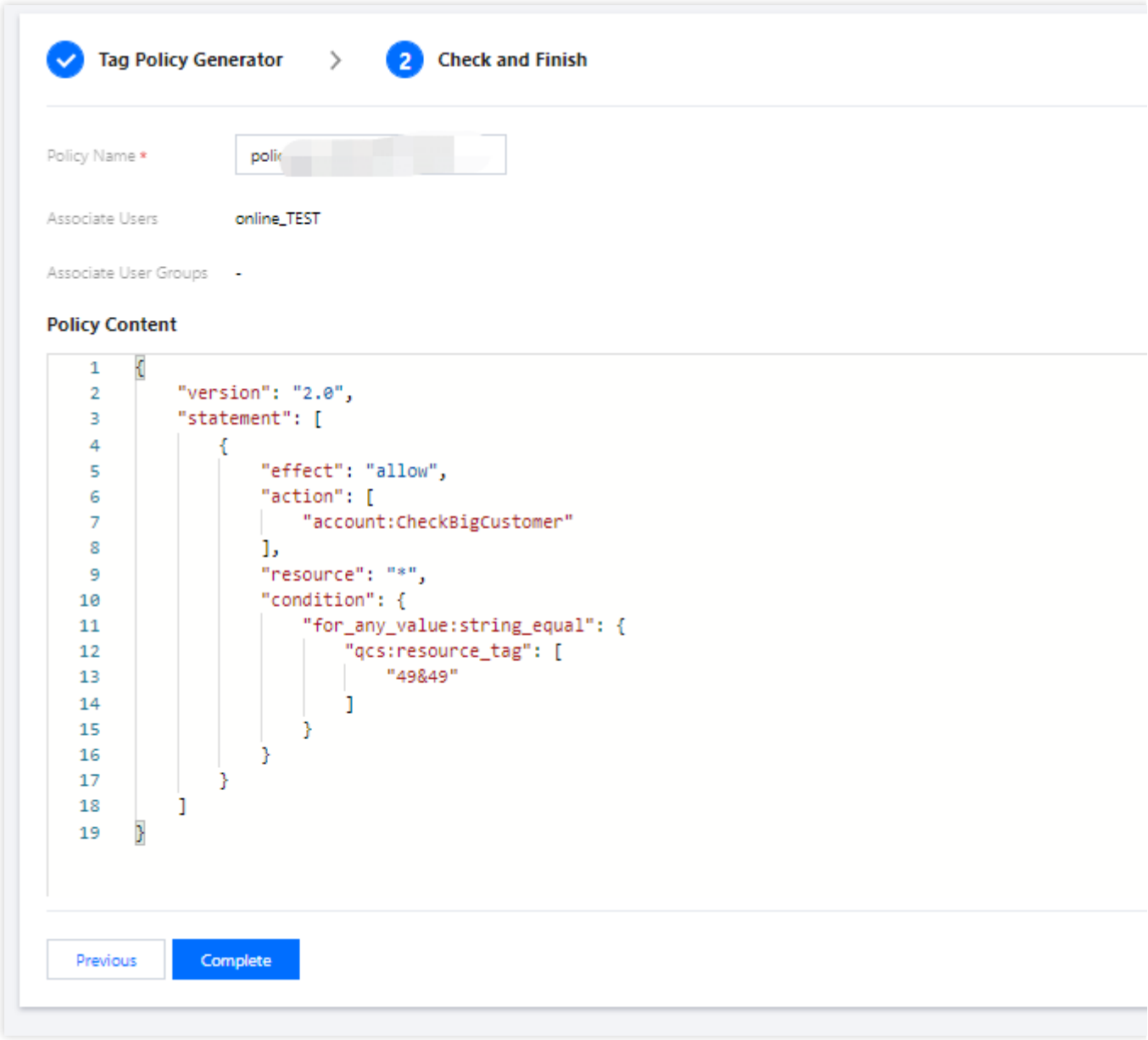
Authorized user: cvmtest01

Bound tag: game:webpage

Operation permissions: All CVM operation permissions and the `DescribeVpcEx` and `DescribeNetworkInterfaces` permissions of VPC. If you are not sure what other APIs are involved, see Authorization by Resource ID > Step 3.

2. Click **Next** and enter a policy name.

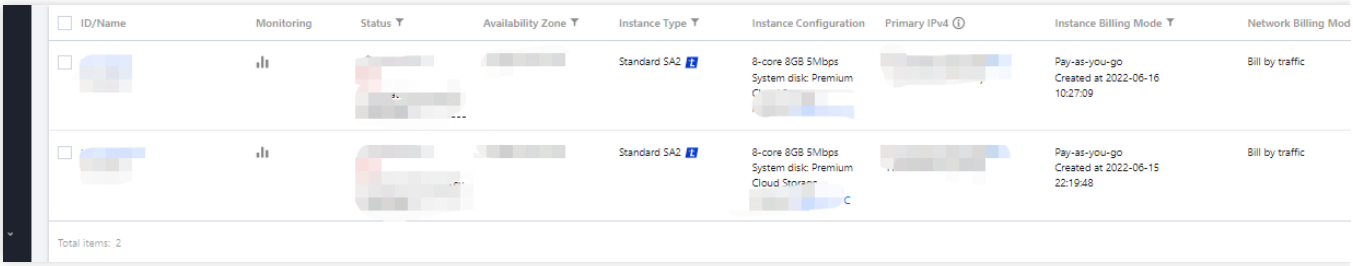
3. Click **Save**.



Step 2: Verify the result

1. Log in to the CVM console as the sub-user `cvmtest01` and access the instance list page.

Then the sub-user `cvmtest01` can start, shut down, restart, rename, and reset the password of the CVM instance.



Enterprise Multi-Account Permissions Management
Overview

Last updated：2024-01-23 17:59:15

Many businesses may have multiple root accounts on Tencent Cloud. The more accounts there are, the more complex the management of accounts and permissions will become. In such circumstances, business administrators hope to manage resource permissions across multiple accounts to reduce the complexity of management. Furthermore, if employees need to access multiple root accounts, they hope to decrease the number of CAM sub-accounts and login frequency.
To address the problems above, Tencent Cloud offers three methods for cross-account access and management including Organization, Role, and Collaborator. The comparison of these three methods is as follows, and you can choose the method suitable for your enterprise based on actual scenarios:

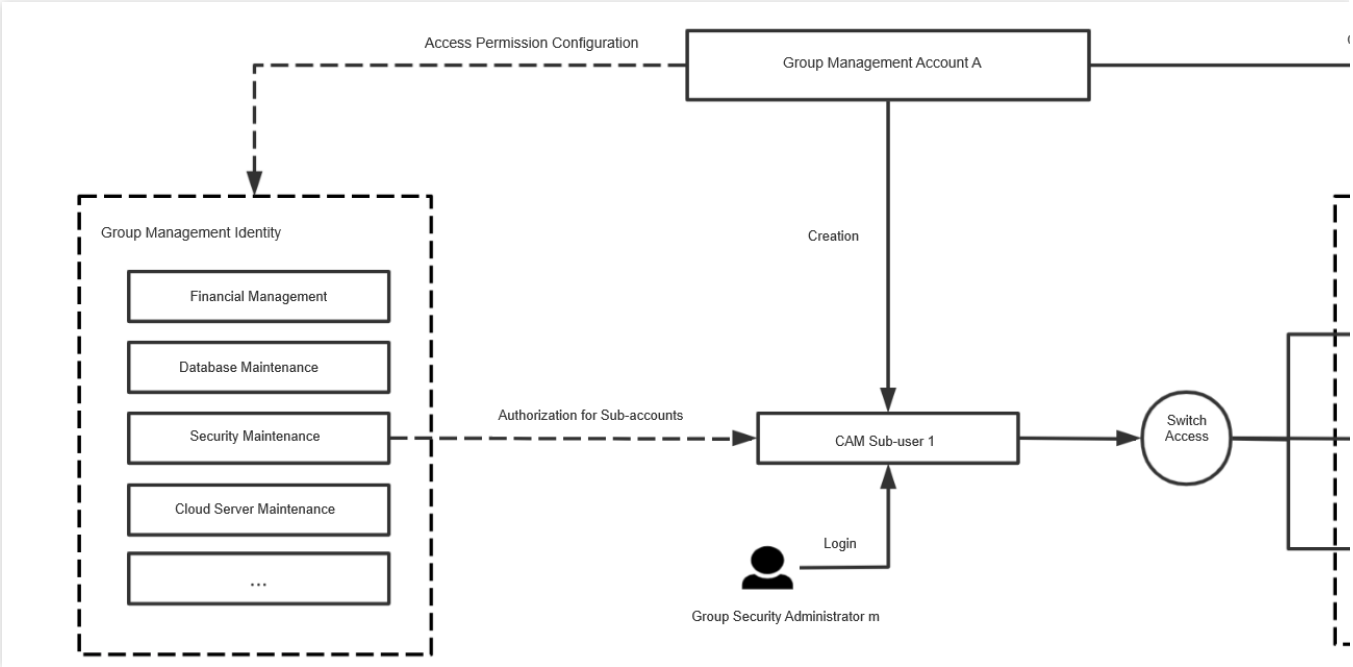
Management Style	Feature Description
Organization	Offers easy-to-use graphical interface and supports usage by company-verified accounts in the same organization.
Role	A role needs to be created under each managed root account. The operation process is relatively long.
Collaborator	Only supports root accounts as collaboration targets.

Tencent Cloud Organization

Last updated：2024-01-23 17:59:15

Organization Overview

The Organization management is a multi-account management product on Tencent Cloud designed specifically for corporate clients. It empowers organization administrators to implement unified management over the Tencent Cloud root accounts of both the group and its subsidiaries, offering management capabilities of accounts, finance and security. For the detailed instructions, please refer to the Tencent Cloud Organization documentation.
Within the Tencent Cloud Organization, the management account can simultaneously grant management permissions of multiple created member accounts to a CAM sub-account. Once authorized, this CAM sub-account allow a single login, and you can select a member and access various member accounts within CAM.



Operation Scenarios

Suppose that a group has Account A and Account B on Tencent Cloud. It choose Account A as the management account to activate the account management product. There are two subsidiaries in the group, possessing Account C and Account D respectively. There is also a security administrator named M in this group, hoping to simultaneously manage the accounts of the group and its subsidiaries.

In such a scenario, the group management account can create a CAM Sub-user 1 for employee M and grant him security operation permissions of Account B, Account C, and Account D. After logging into the Tencent Cloud console through CAM Sub-user 1, the employee can select different member accounts to perform related security operation tasks with no need to create CAM sub-users under each account.

The steps are as follows:

Adding an Authorization

1. Log in to the Organization console using the administrator account and go to the Login permission settings page.
2. Proceed to establish permission models that manage all members, such as network operations, security operations, CVM operations, and financial administrator, etc.
For more information, please refer to Creating Member Login Permission.
3. Navigate to Member Account Management, click **Add Member**, select **Create New** Member and choose the needed permissions to manage.
For further information, kindly refer to Add Organization Members.
4. Upon successful creation of the member, grant sub-users member login permissions via the **Member Account Management** list > **Login Account**.
For more detailed instructions, please refer to Authorizing Sub-Users to Log in to Member Accounts.

Logging in with a Sub-User

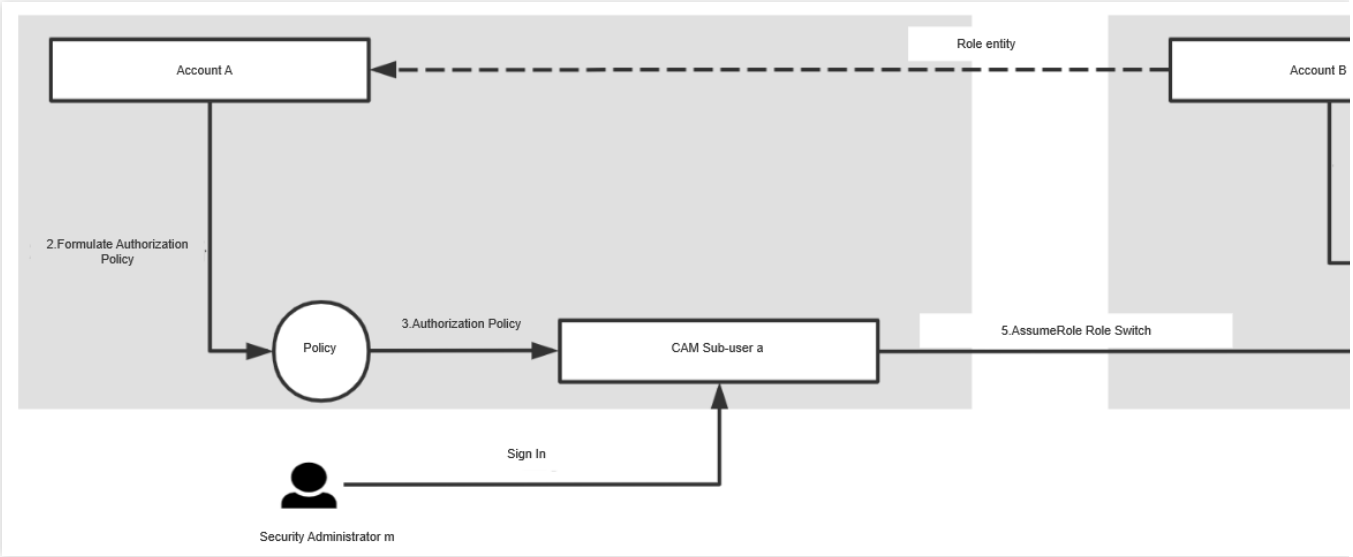
Log into the **Tencent Cloud Organization console** > Member Account Management with a CAM sub-user, choose the member and access permissions you need to manage, and click on the **Login Account** in the operation column. You can log in to the member console with the sub-user and manage operations.

Role

Last updated：2024-01-23 17:59:15

Introduction

A role is a virtual user in CAM, which can be granted a permission policy and has the corresponding permissions of the root account. For more information, see Role Overview.
When creating a role, you can choose to use a Tencent Cloud root account as the role entity, create the role, and bind the authorization policy to it. The root account acting as an entity can grant its CAM sub-accounts the permission to assume this role by creating a permisson policy. Then the CAM sub-accounts can log in to the corresponding root account console by switching roles in the Tencent Cloud console and perform operations within the authorization scope, or they can initiate cross-account requests through API.



Overview

Suppose there are two root accounts in the enterprise, account A and account B, and the security management employee `m` has CAM sub-user `a` under account A. If employee `m` wants to use this sub-account to simultaneously manage the security information under account B, the following steps can be followed:

Directions

1. Create the security operation role `role` under Account B and specify the role entity as root account A.
For more information, see Creating a Role.
2. Under Account A, create a permission policy that supports role assumption of the security operation role `role` through AssumeRole.
3. Assign the policy to CAM sub-user `a` .
For more information, see Authorizing Sub-account with Role Assuming Policy.
4. The employee `m` logs in as CAM sub-user `a` .
5. Employee `m` selects the switch role option on the Tencent Cloud Console and logs in using the security role `role` .
For more information, see Using a Role.
6. Execute security operations-related tasks.
7. If employee `m` needs to carry out security operation-related tasks for multiple root accounts simultaneously, the aforementioned steps can be followed to grant `m` the corresponding security operation permissions for each root account.

Collaborator

Last updated：2024-01-23 17:59:15

Collaborator Overview

Endowed with the identity of the root account, it is added as a collaborator of the existing root account, thus becoming one of the sub-accounts of the current root account, which assists in the management of cloud resources under the root account.

Operation Scenarios

Suppose that an enterprise has multiple accounts on the cloud, such as Account A, Account B, and Account C. It hopes Account B and Account C have the authority to access resources under Account A.

The steps are as follows:

1. Log in to the CAM console with Account A, add Account B and Account C as collaborators and grant them permissions. For detailed procedures, please refer to Creating Collaborator and Setting Collaborator Permissions.
2. Log in to the Tencent Cloud console with Account A or Account C as a collaborator. For detailed procedures, please refer to Logging In to Console with Sub-account.
3. If you want to access a different account, you will need to log out, and log in again. For detailed procedures, please consult Switching Collaborator Identities.

Reviewing Employee Operation Records on Tencent Cloud

Last updated：2024-01-23 17:59:15

Overview

After creating and authorizing CAM sub-users for your employees, they can use CAM sub-users to log in to the Tencent Cloud Console, or use the CAM sub-user keys to access and operate resources associated with your account through API. When many employees need to log in to Tencent Cloud and access resources at the same time, you might want to know the following information:
What resources have been accessed by employees?
Do employees encounter problems with their operations?
Which employee purchased a resource?
How to view modification records of resource configuration?
How to track sensitive operations?
Do employees access Tencent Cloud within your restricted environment?
At this time, you can view and track employee operations through CloudAudit. CloudAudit supports online viewing of Tencent Cloud console and API operation records within 90 days.

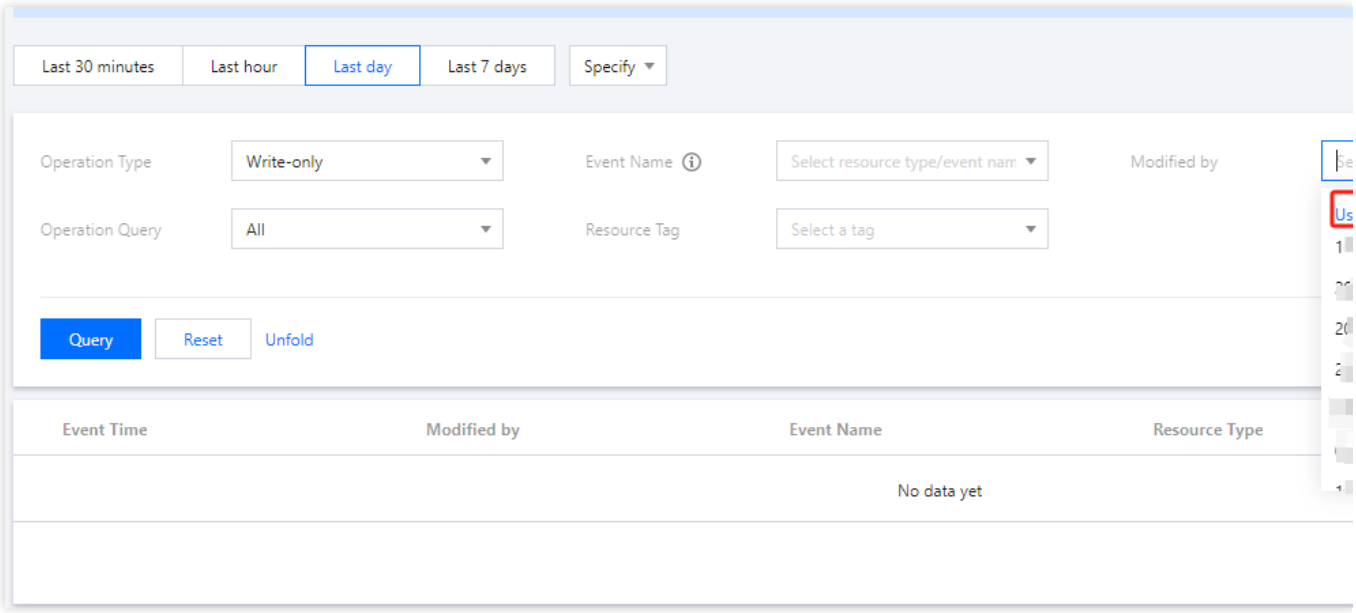
Prerequisites

1. You have created a sub-user.
2. You have logged in to the CloudAudit Console and entered the Operation Records page.

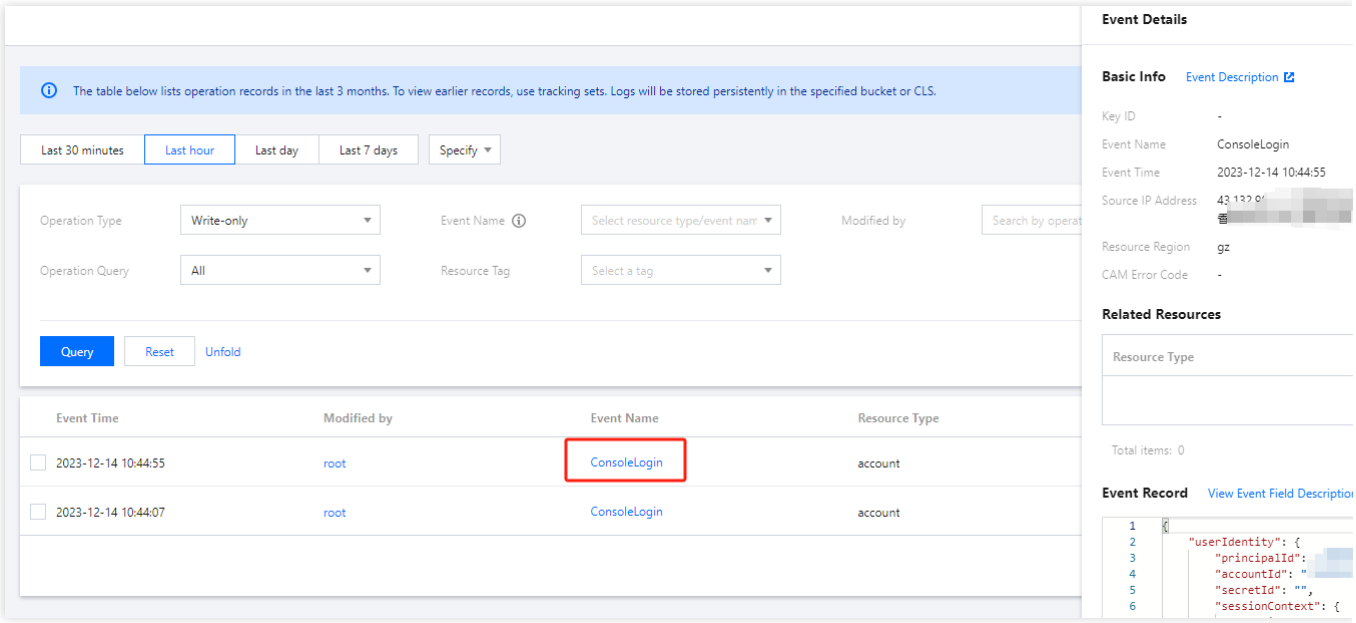
Directions

Viewing Operation Record Event Details

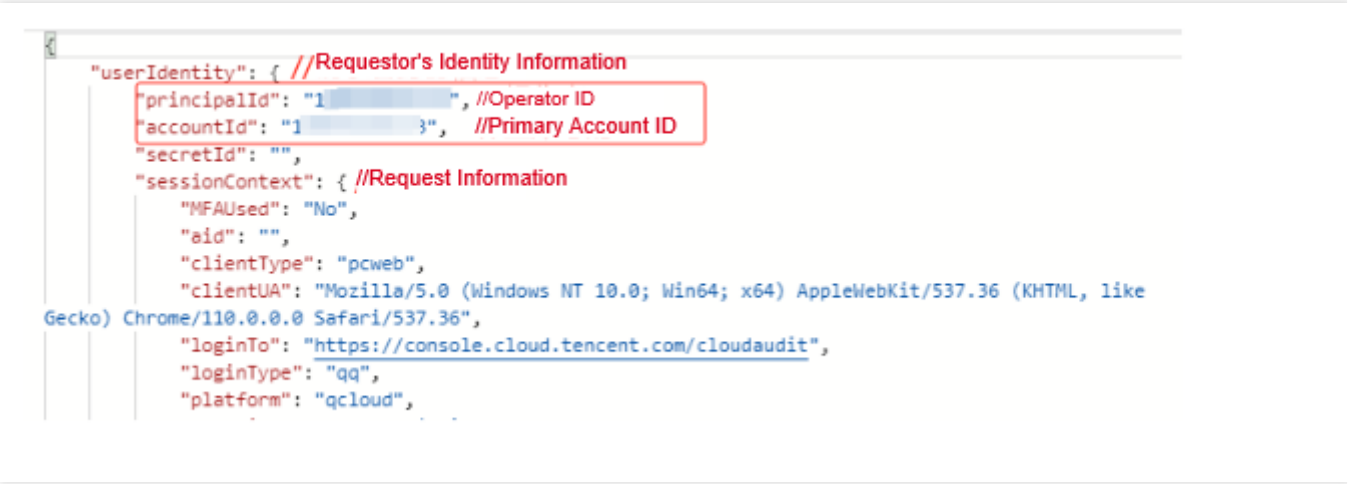
You can filter by the `Operator` condition to search for CAM sub-users/roles, and view the operation records of specified employees.



You can view the event details on the right by clicking **Event Name**. In the specific log summary, identify the account ID and name that performed the actual operation through the operator field, and view the operation source by the source IP address.



In the detailed log information, you can identify the actual operating account ID through the `principalId` .



For detailed operations, please refer to: Viewing Event Details in Operation Record.

Using Tracking Set for Log Delivery

If you need to view employee operation records over a longer period of time, you can use the tracking set feature of CloudAudit to deliver logs to a COS bucket or CLS.

When delivering to CLS, you can select specific operations (such as sensitive operations) for a certain product and configure alarm policies in CLS.

For more detailed operations, please refer to: Shipping Log with Tracking Set.

Implementing Attribute-Based Access Control for Employee Resource Permissions Management
Application Scenarios

Last updated：2024-01-23 17:59:15

Overview

In actual usage within Tencent Cloud, we can define permissions using TAGs through ABAC authorization policies. TAGs are attached to CAM sub-users, roles, and specific cloud resources. Subsequently, permission policies can be defined. These policies use TAG condition keys to grant permissions based on the TAGs of the requesting identity. When you control access to Tencent Cloud resources using TAGs, you can change teams and resources by simply modifying the authorization policies, enjoying greater flexibility.
This document describes how to create a CAM role with a TAG in CAM for employees, along with a policy that grants permissions based on the attributes of the role to access resources matching their TAG. When the employee initiates a request to Tencent Cloud through this role, permissions are granted based on whether the TAG of the role matches the resource TAG. In this case, employees are authorized to view or operate resources needed for their work only.

Usage Examples

Assume in gaming company A, there are two projects, webpage and app, where employee M is a developer for the webpage project and employee N is a developer for the app project. When creating the authorization policy, it is essential to ensure that employees within different teams can access the resources imperative for their work, while also considering the scalability for the company's future growth.

You can create an authorization policy for products that support ABAC policy through the use of resource TAGs and CAM role TAGs. When your employees wish to access Tencent Cloud through combined identities, their attributes will be applied to the role TAGs within Tencent Cloud. Subsequently, ABAC can be used to either approve or reject the access based on these attributes.

Description

For products that support TAG-based authorization, please refer to Tagging-enabled Services.
For the marker condition keys that are supported in the authorization policy, please refer to Conditions.
Based on the above projects and teams, we define the following TAGs:
game-project = web (Corresponding to the webpage project)
game-project = app (Corresponding to the app project)

web = dev (Corresponding to the webpage project developers)
app = dev (Corresponding to the app project developers)
game=dev (Corresponding to the webpage/app project developers)

How It Works

1. Employees log in using the CAM user credentials and then assume the CAM role for their respective teams and projects.
2. Attach the same policy to roles of similar positions, employing TAGs to approve or reject operations.

Verification Scenario

Assume there are two CVMs, ins-78qewdr8(TAG game-project:app) and ins-7txjj4a6(TAG game-project:web), that belong to the app and webpage projects, respectively.
Verification point 1: How to ensure that different employees can only access the CVMs under their own projects after logging in with different CAM sub-users.
Verification point 2: Assume there is a position change and employee n also requires the permissions of the webpage project. How to adjust permissions quickly.
Verification Point 3: Suppose the company has added a new H5 type project. How to quickly grant permissions for the new project to the employees.

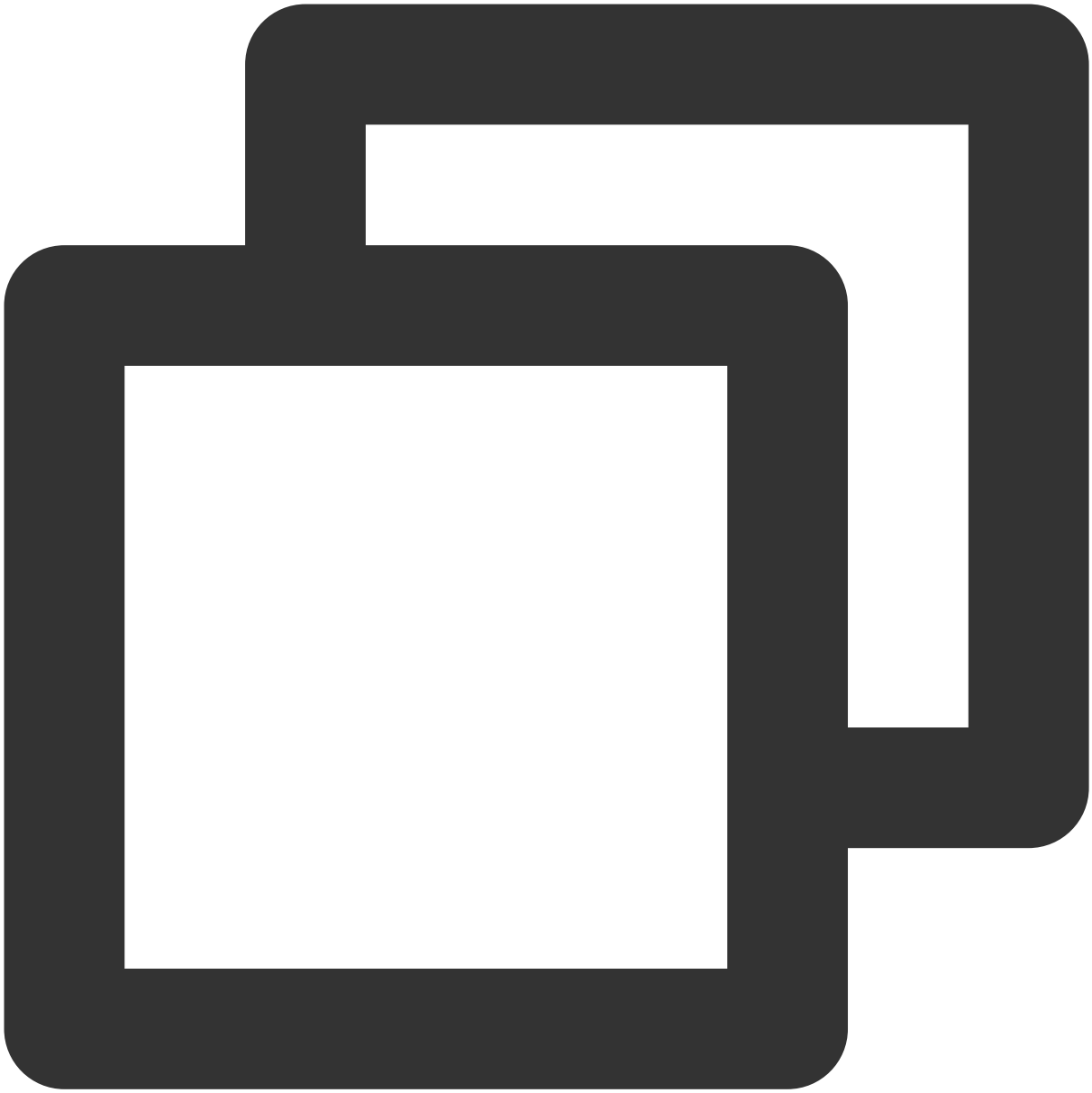
Directions

Step 1: Create a TEST-IMAGE CAM sub-user.

1. Create a customized policy named "access-assume-role". The policy content is "Allow assuming ABAC role when the identity's TAG matches role's TAG".

Description

For detailed steps on how to create a CAM policy, please refer to Creating Role.



```
{
 "version": "2.0",
 "statement": [
     {
         "effect": "allow",
         "action": [
             "sts:AssumeRole"
         ],
         "resource": "*",
         "condition": {
             "for_any_value:string_equal": {
```

```
                "qcs:resource_tag": [
                    "game&${qcs:principal_tag_value}"
                ]
            }
        }
    },
    {
        "effect": "allow",
        "action": [
            "cam:ListUserTags",
            "cam:ListLoginRoles"
        ],
        "resource": [
            "*"
        ]
    }
 ]
}
```

2. Create the CAM sub-users m-developer and n-developer, bind them with the "access-assume-role" authorization policy, and associate the following TAGs with these sub-users.

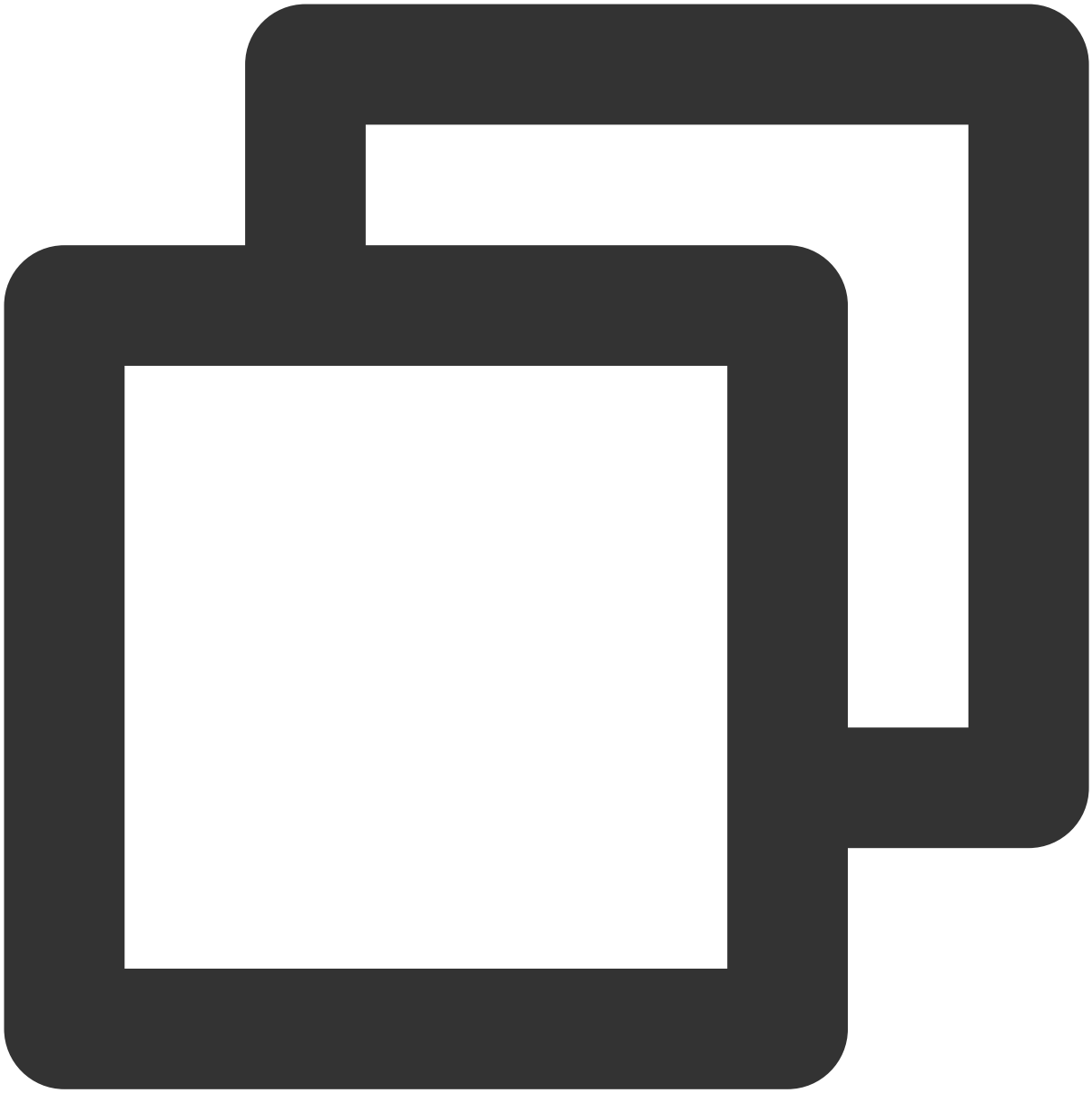
Description

For a detailed guide on how to create CAM sub-users, please refer to Creating Sub-User.

Sub-user Name	Associated TAG
m-developer	web=dev
n-developer	app=dev

Step 2: Create an ABAC Policy

1. Customize a policy named 'access-resource-project' (using the cvm product as an example). The policy content is as follows:



```
{
 "version": "2.0",
 "statement": [
     {
         "effect": "allow",
         "action": "cvm:*",
         "resource": "*",
         "condition": {
             "for_any_value:string_equal": {
                 "qcs:request_tag": [
                    "game-project&${qcs:principal_tag_key}"
```

```
                ]
            }
        }
    },
    {
        "effect": "allow",
        "action": "cvm:*",
        "resource": "*",
        "condition": {
            "for_any_value:string_equal": {
                "qcs:resource_tag": [
                    "game-project&${qcs:principal_tag_key}"
                ]
            }
        }
    },
     {
        "effect": "allow",
        "action": [
            "vpc:DescribeVpcEx",
            "vpc:DescribeSubnetEx",
            "vpc:DescribeNetworkInterfaces",
            "cvm:DescribeDiskSecurityConfigurations",
            "cvm:DescribeCbsStorages",
            "tag:DescribeTagKeys",
            "tag:DescribeTagValues"
        ],
        "resource": [
            "*"
        ]
    }

 ]
}
```

The 'game-project' is associated with the key and value of TAG bound to ${qcs:principal_tag_key}, identifying the values related to a specific TAG key within the project.

2. Create the role 'access-developer-role', associate the above policy and bind the following TAG.

Description

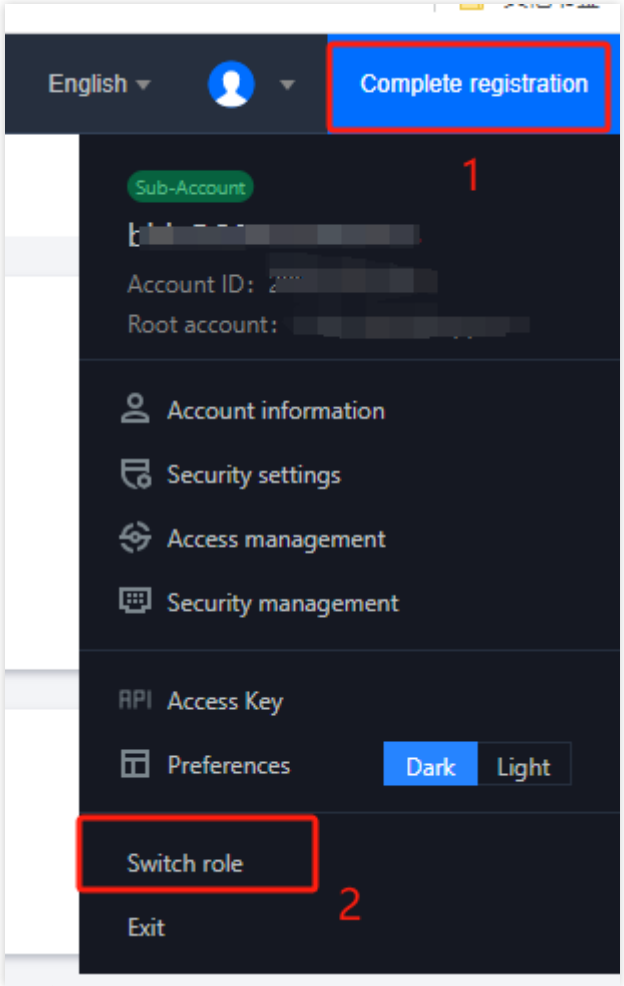
For detailed steps on how to create a CAM policy, please refer to Creating Roles.

CAM Role Name	Associated TAG
access-developer-role	game=dev

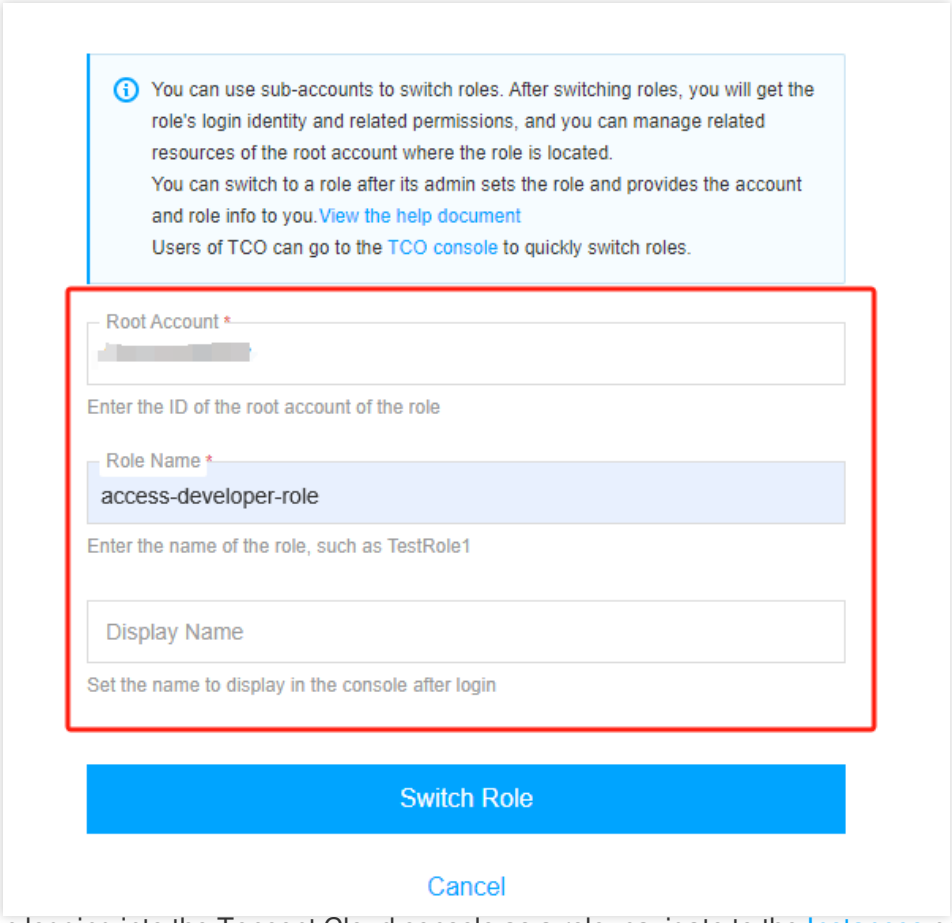
Step 3: Verify the scenario.

Verification Point 1: After the login with various sub-users, only the CVMs under the corresponding project can be accessed.

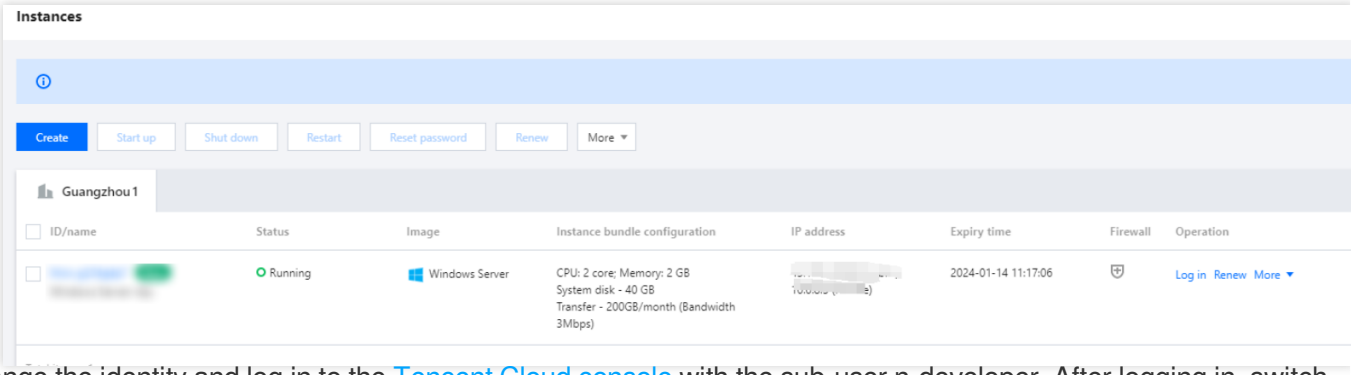
1. Log in to the Tencent Cloud console with the sub-user m-developer. In the upper-right corner of the console, click **Switch Role**.



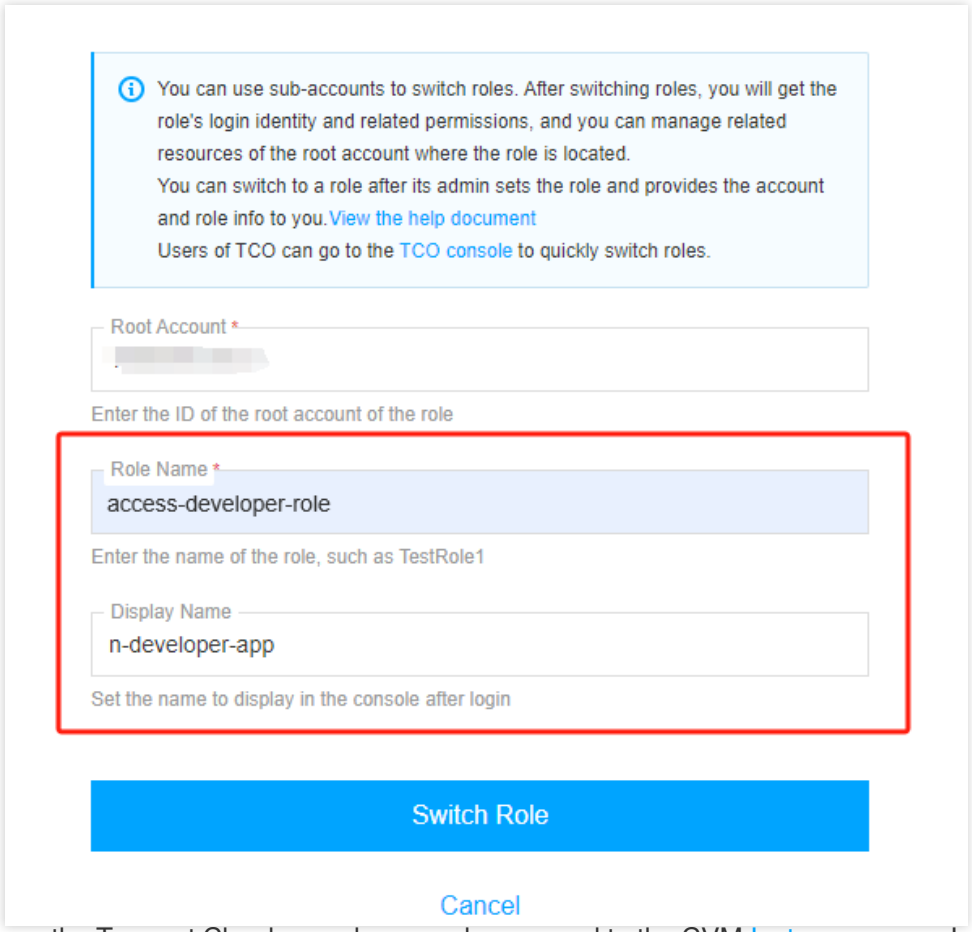
2. On the Switch Role page, select access-developer-role for the Role Name, then click **Switch Role**.



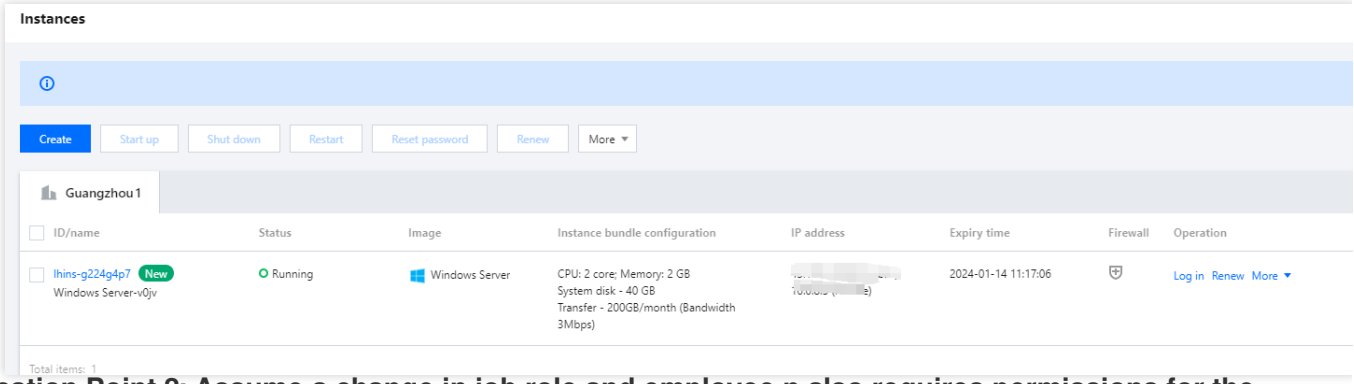
3. Upon logging into the Tencent Cloud console as a role, navigate to the Instances page in CVM. If you are only able to view lhins-g224g4p7 in the CVM Product Console, then it meets the expectation.



4. Change the identity and log in to the Tencent Cloud console with the sub-user n-developer. After logging in, switch roles, and select access-developer-role for the Role Name. The name is displayed as n-developer-app. Then click **Switch Role**.

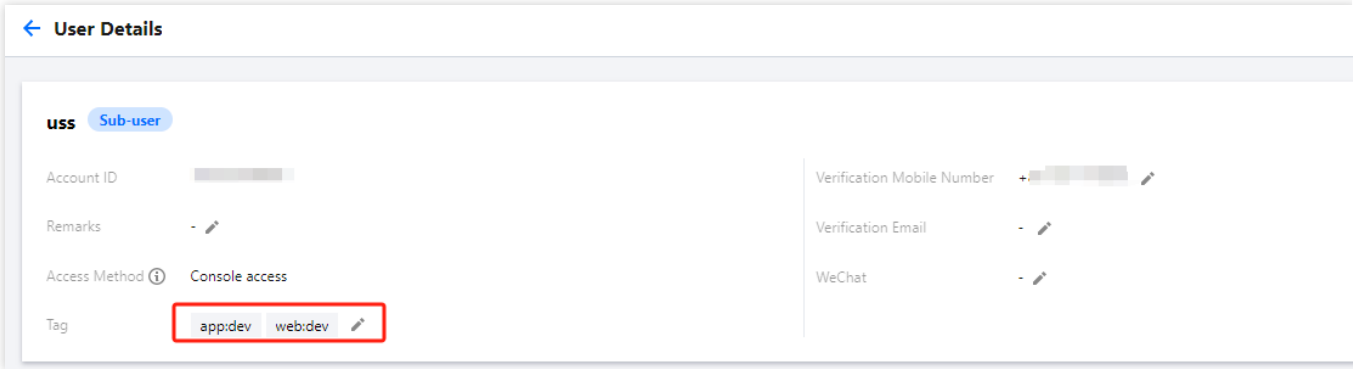


5. Access the Tencent Cloud console as a role, proceed to the CVM Instances page. In the CVM product console, if you are only able to view the cloud server ins-78qewdr8 (tagged as game-project:app), then it meets the expectation.



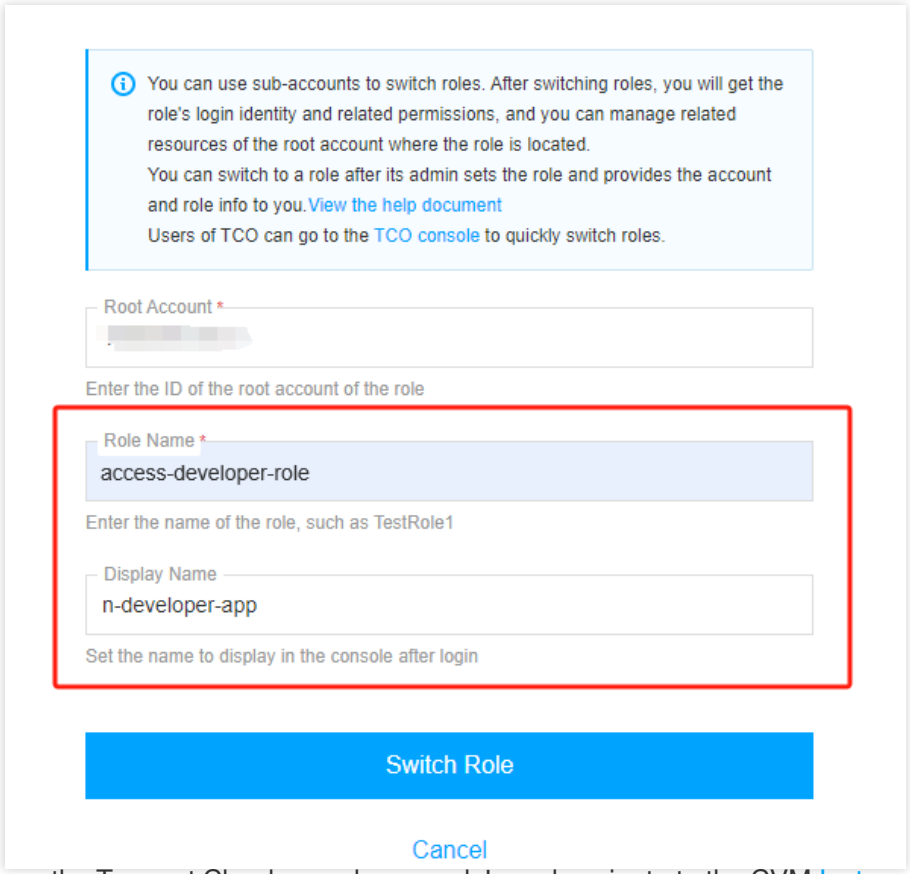
Verification Point 2: Assume a change in job role and employee n also requires permissions for the webpage project, how should this be set up?

In the current scenario, we can simply add the TAG app:web to the CAM sub-user n-developer corresponding to employee n in the user details of the CAM Console.

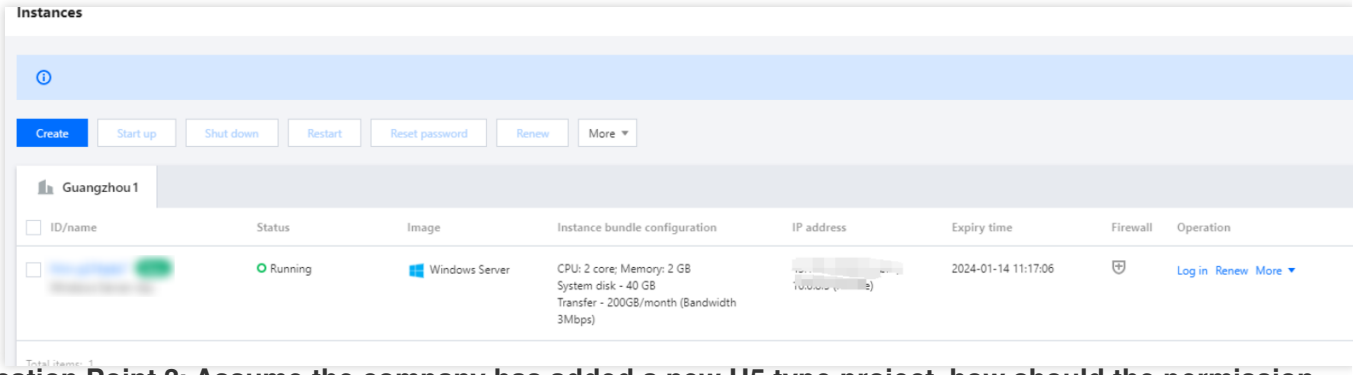


1. Log in to the Tencent Cloud console as the sub-user n-developer, and in the upper-right corner of the console, click **Switch Role** under the account.

2. On the switch role page, select 'web' for the application, 'access-developer-role' for the role, and 'n-developer-web' for the alias. Then click **Switch Role**.



3. Access the Tencent Cloud console as a role's and navigate to the CVM Instances page. If you are only able to view the CVM lhins-g224g4p7 in the CVM product console, then it meets the expectation.



Verification Point 3: Assume the company has added a new H5 type project, how should the permission policy be adjusted to fit this?

After the H5 project is added, if there is a need to grant development permissions for the H5 project, it does not require changes to the existing policy itself; all required includes:

1. Create new sub-users for colleagues engaged in developing the H5 project.
2. Bind the sub-user with the corresponding TAG for the H5 project and associate it to the access-assume-role policy.

During tag-based authentication, only tag key matching is supported

Last updated：2024-01-23 17:59:15

This document describes how to grant your sub-account permission to all resources under a tag and how to grant your sub-account permission to bind only a tag key.

Note:

The `resource_tag` grants permission to all resources under a tag, while `request_tag` grants a sub-account permission to only bind a tag key. However, this does not take effect on the console lists and related APIs.

Granting permission to all resources under a tag key (resource_tag)

Overview

If your organization has purchased multiple Tencent Cloud resources, and the resources are managed by tag groups, you may want to grant permission to all resources associated with a tag key (resource_tag).

Suppose that:

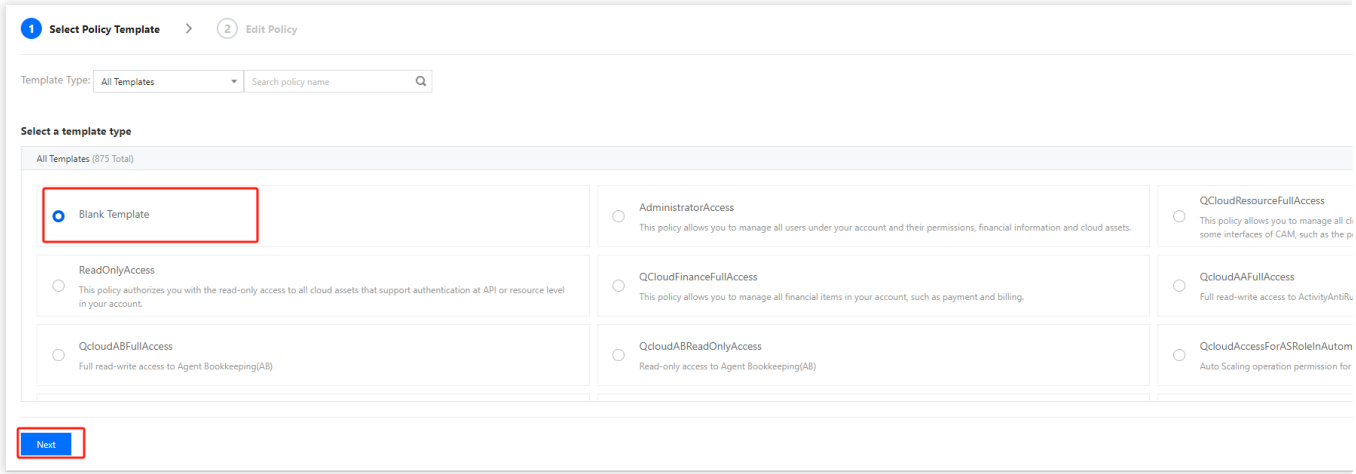
There is a sub-account `Operator` under the enterprise account `CompanyExample` .

There is a tag key **Operation** under the enterprise account `CompanyExample` .

The enterprise account `CompanyExample` wants to grant the sub-account `Operator` permission to all resources under the tag key **Operation**.

Directions

1. Log in to the CAM console with the enterprise account `CompanyExample` .
2. On the **Policies** page, click **Create Custom Policy** and then Create by Policy Syntax.
3. Select Blank Template under the Select a template type, then click **Next** to proceed to the editing policy page.

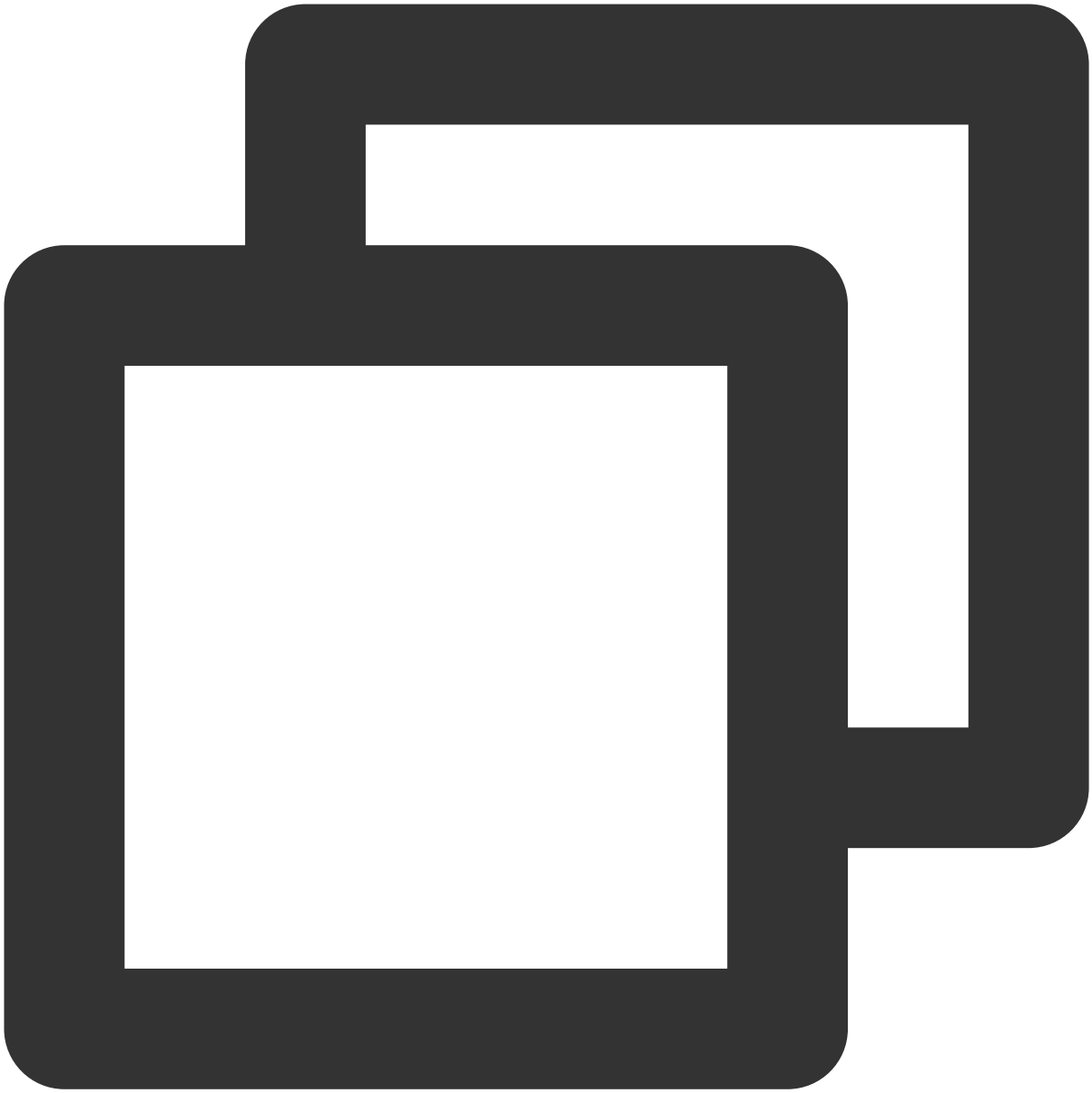


4. On the editing policy page, fill in the following form:

Policy Name: It defaults to `policygen-current date`. It is recommended to define a unique and meaningful policy name, such as `Operator-resource_tag`.

Description: Optional, write it yourself.

Policy Content: Copy the following content and fill it out. Among them, `operation` is the tag key name which can be Chinese or English and `false` is a fixed tag value.

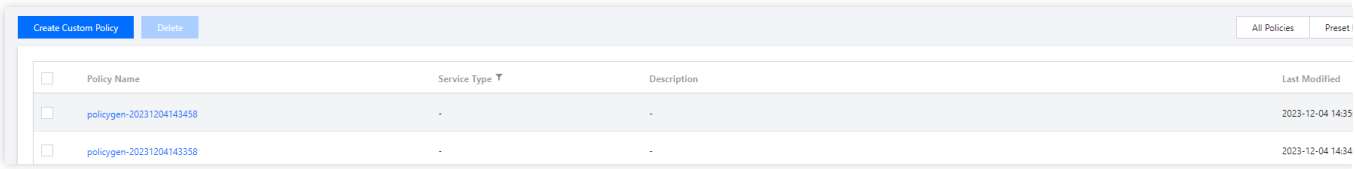


```
{
"version": "2.0",
 "statement": [
   {
        "effect": "allow",
        "action": "*",
        "resource": "*",
        "condition": {
            "null_equal": {
                "qcs:resource_tag/operation": "false"
            }
```

```
                }
            }
        ]
    }
```

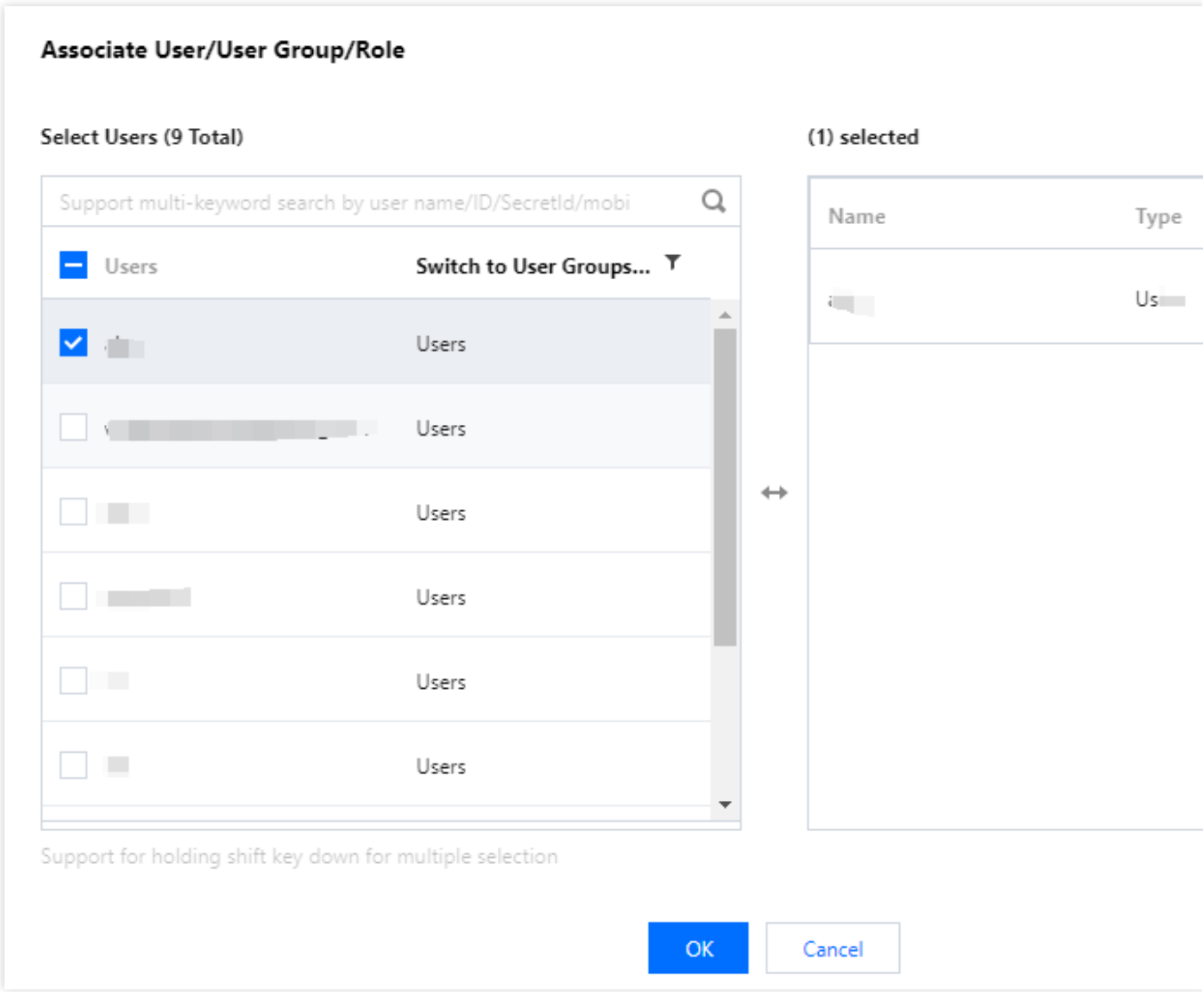
5. Click **Complete** to create the policy. The newly created policy will be displayed on the policy list page.

6. In the Policies List, search for the policy you just created, and then click **Associate User/Group/Role** in the operation column on the right.



7. In the pop-up **Associate User/Group/Role** window, search for and select the sub-account `Operator` , then click **OK** to complete the permission.

The `Operator` sub-account will possess all the permission under the **Operation** tag.



Granting a sub-account permission to bind a tag Key (request_tag)

Overview

If your organization has purchased multiple Tencent Cloud resources, and the resources are managed by tag groups, you may want to grant permission to all resources associated with a tag key (request_tag).

Suppose that:

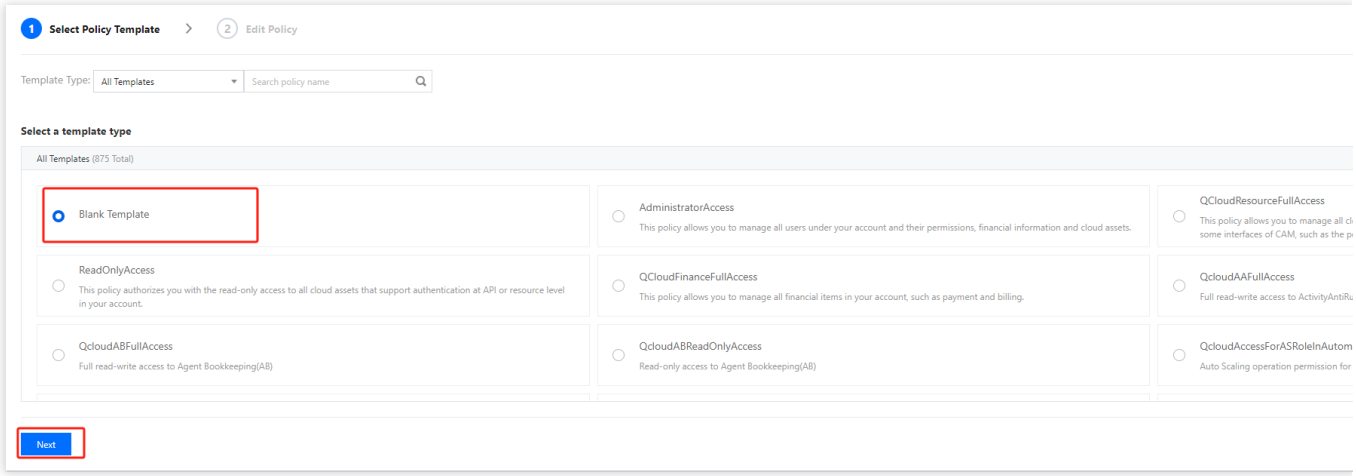
There is a sub-account `Developer` under the enterprise account `CompanyExample`.

There is a tag key **Development** under the enterprise account `CompanyExample`.

The enterprise account `CompanyExample` wants to grant the sub-account `Developer` permission to all resources under the tag key **Development** (request_tag).

Directions

1. Log in to the CAM console with the enterprise account `CompanyExample` .
2. On the **Policies** page, click **Create Custom Policy** and then Create by Policy Syntax.
3. Select Blank Template under the Select a template type, then click **Next** to proceed to the editing policy page.

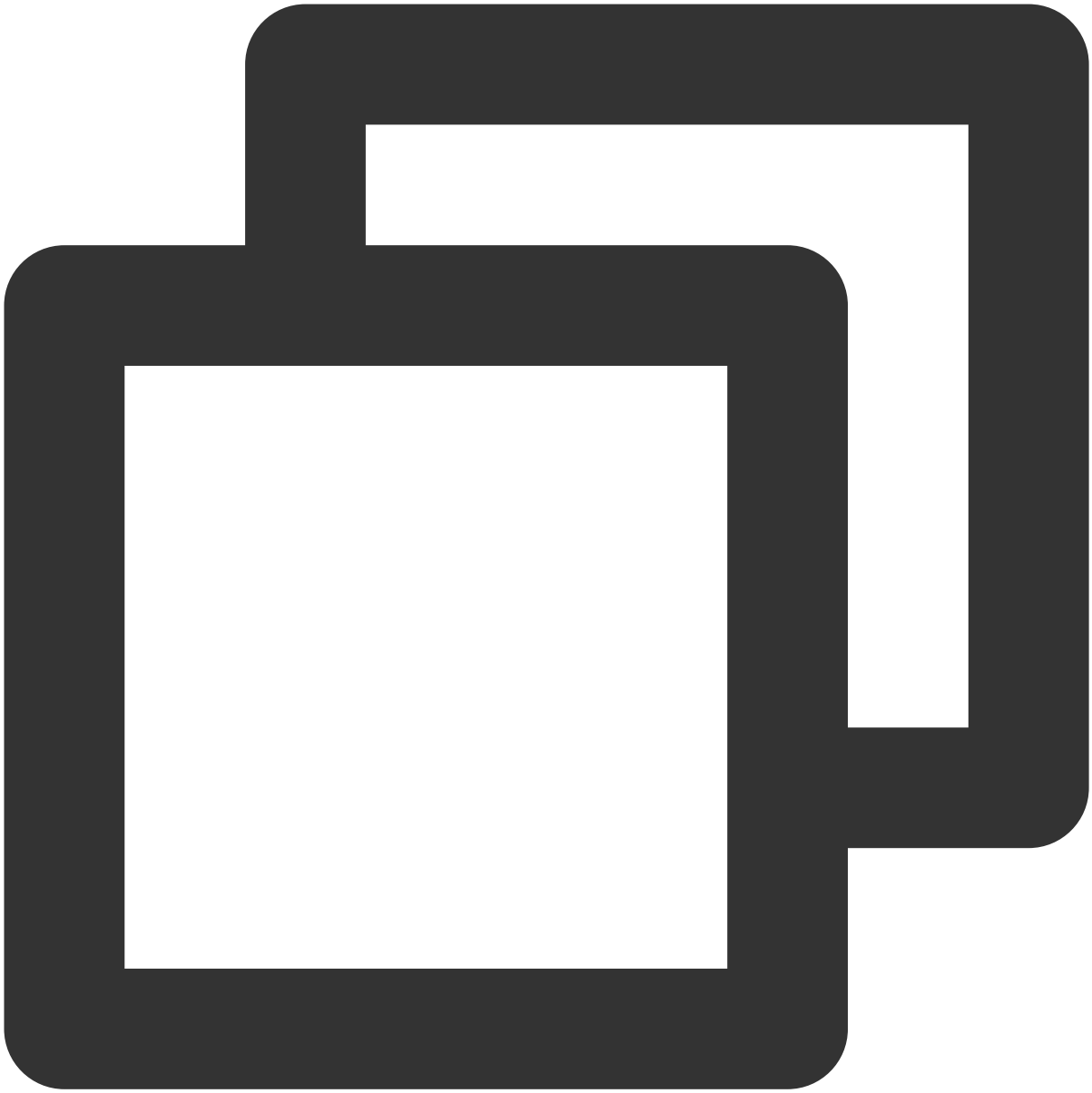


4. On the editing policy page, fill in the following form:

Policy Name: It defaults to `policygen-current date` . It is recommended to define a unique and meaningful policy name, such as `Developer-request_tag` .

Description: Optional, write it yourself.

Policy Content: Copy the following content and fill it out. Among them, `develop` is the tag key name which can be Chinese or English and `false` is the fixed tag value.

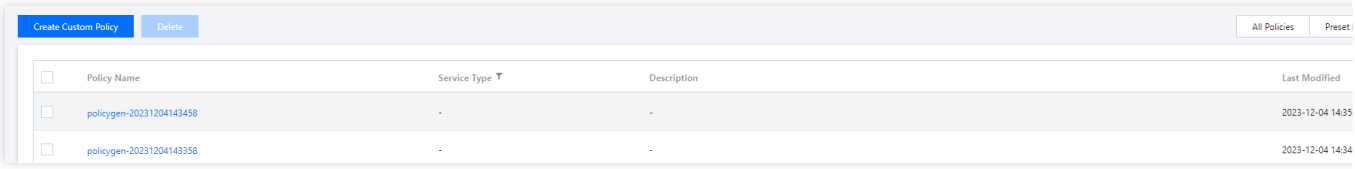


```
{
"version": "2.0",
 "statement": [
   {
        "effect": "allow",
        "action": "*",
        "resource": "*",
        "condition": {
            "null_equal": {
                "qcs:request_tag/develop": "false"
            }
```

```
                }
            }
        ]
    }
```

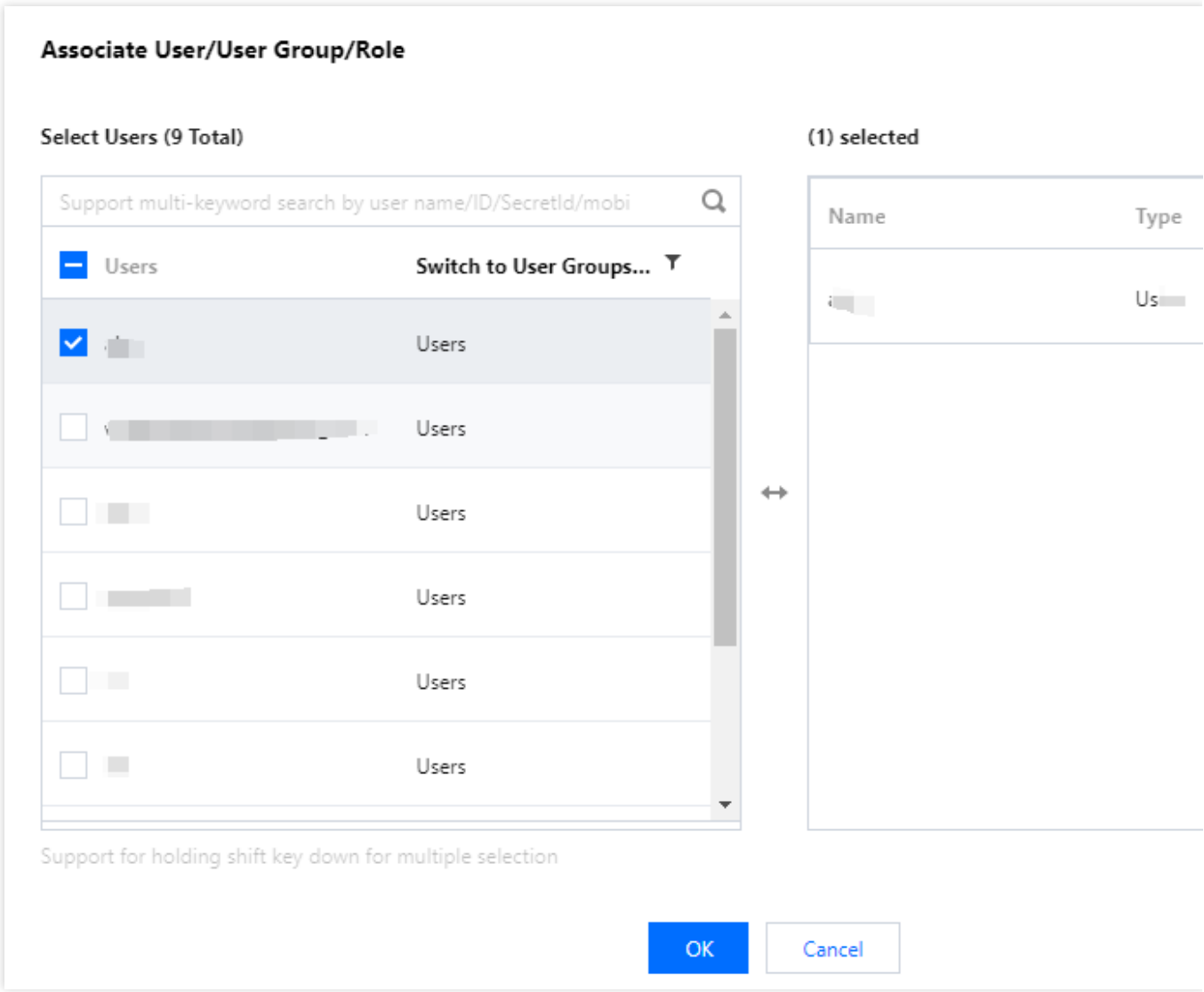
5. Click **Complete** to create the policy. The newly created policy will be displayed on the policy list page.

6. In the Policies List, search for the policy you just created, and then click **Associate User/Group/Role** in the operation column on the right.



7. In the pop-up **Associate User/Group/Role** window, search for and select the sub-account `Developer`, and then click **OK** to complete the permission.

The `Developer` sub-account will possess all the permission to bind the **develop** tag key.



Associated documents

If you want to understand how to associate resources with tags, please refer to Querying Resources by Tag.