Cloud Access Management

Business Use Cases

Service Notice

This document provides an overview of the as–is details of Tencent Cloud's products and services in their entirety or part. The descriptions of certain products and services may be subject to adjustments from time to time.

The commercial contract concluded by you and Tencent Cloud will provide the specific types of Tencent Cloud products and services you purchase and the service standards. Unless otherwise agreed upon by both parties, Tencent Cloud does not make any explicit or implied commitments or warranties regarding the content of this document.

Contact Us

We are committed to providing personalized pre–sales consultation and technical after–sale support. Don't hesitate to contact us at 4009100100 or 95716 for any inquiries or concerns.

Contents

Business Use Cases

TencentDB for MySQL

Allowing Account to View TencentDB for MySQL Instances Under Specified Tag

Last updated：2024-02-01 21:02:59

Under the corporate account CompanyExample (ownerUin is 12345678), there is a sub-account named Developer. This sub-account requires view permissions for two MySQL instances (instance ID cdb-1 tagged as game&webpage and instance ID cdb-2 tagged as game&app) owned by the corporate account CompanyExample.

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "effect": "allow",
         "action": [
            "cdb:Describe*"
         ],
         "resource": "*",
         "condition": {
            "for_any_value:string_equal": {
               "qcs:resource_tag": [
                  "game&webpage",
                  "game&app"
               ]
            }
         }
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

> **Note**
> The sub-account “Developer” can only view the resources of instances with the IDs being “cdb-1” and “cdb-2” in the TencentDB for MySQL query list.

Granting a sub-account the operational permissions for CDB in a specific region

Last updated：2024-02-01 21:04:00

Under the corporate account CompanyExample (with ownerUin as 12345678), there is a sub-account named cdb. This sub-account has been granted full operation permissions (*) for all CDB instances within a specific region (gz). Specifically, this sub-account has the permission to execute any operation on these CDB instances.

1. Create a policy through policy syntax.

```
{
 "version": "2.0"
     "statement": [
         {
            "action": "cdb:*",
            "resource": "qcs::cdb:gz::*",
            "effect": "allow"
         }
     ]
}
```

   - action: Indicates the permitted operation. `cdb:*` signifies that all CDB operations are allowed.
   - effect: Indicates whether the permission described in the statement is allowed or denied. `allow` signifies that the permission is allowed.
   - resource: Indicates the scope of resources applicable to this statement. `qcs::cdb:gz::*` signifies that the sub-account is granted full operation permissions for all CDB instances within a specific region (gz). `*` represents a wildcard, indicating permissions for all instances.

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

CLB

Authorizing Sub-account Full Access to CLB（Includes payment permission）

Last updated：2024-02-01 21:04:51

Under the corporate account CompanyExample (with ownerUin 12345678), there is a sub-account named Developer. This sub-account requires full management permissions for the CLB service of the corporate account CompanyExample, including all operations such as creation, management, and CLB order payment.

Scenario A:

The corporate account CompanyExample directly grants the preset policies QcloudCLBFullAccess and QcloudCLBFinanceAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":[
      {
        "effect": "allow",
        "action": "clb:*",
        "resource": "*"
      },
      {
          "effect": "allow",
          "action": "finance:*",
          "resource": "qcs::clb:::*"
      }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Full CLB Access other than the Payment Permission

Last updated：2024-02-01 21:08:00

Under the corporate account CompanyExample (ownerUin is 12345678), there is a sub-account named Developer. This sub-account needs to have full management permissions (including creation, management, and all other operations) for the CLB service of the corporate account CompanyExample, excluding payment permissions. That is, it can place orders but cannot make payments.

Scenario A:

The corporate account CompanyExample directly grants the preset policy QcloudCLBFullAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":[
     {
        "effect": "allow",
        "action": "clb:*",
        "resource": "*"
     }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

CMQ
Authorizing a Sub-account Full Permissions to Use Messaging Services

Last updated：2024-02-01 21:09:22

Under the corporate account CompanyExample, there is a sub-account named Developer. This sub-account requires full permissions for all message queues under the corporate account CompanyExample. Regardless of whether the message queue is based on a topic model or a queue model, it should be readable and writable.

Scenario A:

The corporate account CompanyExample directly grants the preset policies QCloudCmqQueueFullAccess and QCloudCmqTopicFullAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
 {
   "version": "2.0",
   "statement":[
    {
       "effect": "allow",
       "action": ["cmqtopic:","cmqqueue:"],
       "resource": "*"
    }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing a Sub-account Full Permissions to Access the Message Queue It Created

Last updated: 2024-02-01 21:10:12

Under the corporate account CompanyExample, there is a sub-account Developer which requires to access the message queue the sub-account has created.

Scenario A:

The corporate account CompanyExample directly grants the preset policies QCloudCmqQueueCreaterFullAccess and QCloudCmqTopicCreaterFullAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":
  [
    {
      "effect": "allow",
      "action": "cmqtopic:*",
      "resource": "qcs::cmqtopic:::topicName/uin/${uin}/*"
    },
    {
      "effect": "allow",
      "action": "cmqqueue:*",
      "resource": "qcs::cmqqueue:::queueName/uin/${uin}/*"
    }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing a Sub-account Permission to Read a Topic-based Message Queue

Last updated: 2024-02-01 21:10:48

The corporate account CompanyExample (with an ownerUin of 12345678) possesses a theme-model-based message queue. In addition, it has a sub-account Developer, which requires to access this message queue.

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement": [
   {
     "action": "cmqqueue:SendMessage",
     "resource":"qcs::cmqqueue:::queueName/uin/12345678/test-caten",
     "effect": "allow"
   }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

COS

Authorizing Sub-account Full Access to Specific Directory

Last updated：2024-02-01 21:11:35

Under the corporate account CompanyExample (with ownerUin as 12345678 and appId as 1250000000), there is a sub-account named Developer. This sub-account requires full access permissions to the dir1 directory of the Bucket1 storage bucket in the Shanghai region of CompanyExample's COS service.

Solution A:

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement":[
    {
       "effect": "allow",
       "action": "cos:*",
       "resource": ["qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1/*",
                "qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1"]
    }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Solution B:

Setting up Policy and ACL through the COS console. For specifics, please see COS Product Documentation.

Authorizing Sub-account Read-only Access to Files in Specific Directory

Last updated：2024-02-01 21:12:08

Under the corporate account CompanyExample (with ownerUin as 12345678 and appId as 1250000000), there is a sub-account named Developer. This sub-account requires read permissions for the files in the dir1 directory of the Bucket1 storage bucket in the Shanghai region of CompanyExample's COS service.

Solution A:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":[
   {
      "effect": "allow",
      "action":  [
              "cos:List*",
              "cos:Get*",
              "cos:Head*",
              "cos:OptionsObject"
           ],
      "resource": "qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1/*"
   }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Solution B:

Setting up Policy and ACL through the COS console. For specifics, please see COS Product Documentation.

Authorizing Sub-account Read/Write Access to Specific File

Last updated：2024-02-01 21:12:39

Under the corporate account CompanyExample (with ownerUin as 12345678 and appId as 1250000000), there is a sub-account named Developer. This sub-account requires read/write permissions for the object Object1 located in the dir1 directory of the Bucket1 storage bucket in the Shanghai region of CompanyExample's COS service.

Solution A:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":[
   {
     "effect": "allow",
     "action": "cos:*",
     "resource": "qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1/object1"
   }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Solution B:

Setting up Policy and ACL through the COS console. For specifics, please see COS Product Documentation.

Authorizing Sub-account Read-only Access to COS Resources

Last updated：2024-02-01 21:13:03

Under the corporate account CompanyExample (with ownerUin 12345678), there is a sub-account named Developer. This sub-account requires read-only access to the COS service of the corporate account CompanyExample (access to COS buckets, objects, object lists, and so on).

Scenario A:

The corporate account CompanyExample directly grants the preset policy QcloudCOSReadOnlyAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":[
   {
      "effect": "allow",
      "action":  [
              "cos:List*",
              "cos:Get*",
              "cos:Head*",
              "cos:OptionsObject"
           ],
      "resource": "*"
   }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Grant the sub-account read/write permissions for all files in a specific directory, excluding a specified file

Last updated：2024-02-01 21:13:38

Under the corporate account CompanyExample (with ownerUin as 12345678 and appId as 1250000000), there is a sub-account named Developer. This sub-account requires read/write permissions for all objects in the dir1 directory of the Bucket1 storage bucket in the Shanghai region of the CompanyExample's COS service. However, it does not have read/write permissions for the Object1 object under the same directory.

Solution A:

1. Create a policy through policy syntax.

```
 {
  "version": "2.0",
  "statement":
  [
   {
     "effect": "allow",
     "action": "cos:*",
     "resource": "qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1/*"
   },
   {
     "effect": "deny",
     "action": "cos:*",
     "resource": "qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1/Object1"
   }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Solution B:

Configure Policy and ACL through the COS Console. For details, please see ACL Access Control Practice.

Authorizing Sub-account Read/Write Access to Files with Specified Prefix

Last updated：2024-02-01 21:14:40

Under the corporate account CompanyExample (with ownerUin as 12345678 and appId as 1250000000), there is a sub-account named Developer. This sub-account requires read/write permissions for objects prefixed with test in the dir1 directory of the Bucket1 storage bucket in the Shanghai region of CompanyExample's COS service.

Solution A:

1. Create a policy through policy syntax.

```
 {
   "version": "2.0",
   "statement":[
    {
       "effect": "allow",
       "action": "cos:*",
       "resource": "qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1/test*"
    }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Solution B:

Configure Policy and ACL through the COS Console. For details, please see the COS Documentation.

Cross-account access: Granting the primary account permission to access specific files of another primary account

Last updated：2024-02-01 21:15:19

The corporate account CompanyGranter (with ownerUin as 12345678 and appId as 1250000000) possesses an object named Object1, located in the dir1 directory of the Bucket1 storage bucket in the Guangzhou region. Another corporate account CompanyGrantee (with ownerUin as 87654321) requires read/write permissions for the aforementioned object.
Configure Policy and ACL through the COS Console. For more information, see ACL Access Control Practice.

Cross-account access: Granting sub-account permission to access specific files of another root account

Last updated：2024-02-01 21:16:04

The corporate account CompanyGranter (with ownerUin as 12345678 and appId as 1250000000) possesses an object, Object1, located in the dir1 directory of the Bucket1 storage bucket in the Guangzhou region. Another corporate account CompanyGrantee (with ownerUin as 87654321) has a sub-account which requires the read/write permissions for the aforementioned object.
This involves permission propagation.

1. The corporate account CompanyGrantee creates a policy using policy syntax.

```
 {
   "version": "2.0",
   "statement":[
    {
      "effect": "allow",
      "action": "cos:*",
      "resource": "qcs::cos:ap-shanghai:uid/1250000000:Bucket1-1250000000/dir1/Object1"
    }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.
3. The corporate account CompanyGranter grants the object Object1 to the corporate account CompanyGrantee by configuring Policy and ACL through the COS Console. For more information, please see COS Product Documentation.

CVM

Authorizing Sub-account Full Access to CVMs

Last updated：2024-02-01 21:16:35

Under the corporate account CompanyExample (with ownerUin 12345678), there is a sub-account named Developer. This sub-account requires full management permissions for the CVM services of the corporate account CompanyExample, including all operation permissions such as creation, management, and cloud server order payment.

Scenario A:

The corporate account CompanyExample directly grants the preset policies QcloudCVMFullAccess and QcloudCVMFinanceAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":[
      {
        "effect": "allow",
        "action": "cvm:*",
        "resource": "*"
      },
      {
          "effect": "allow",
          "action": "finance:*",
          "resource": "qcs::cvm:::*"
      }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Read-only Access to CVMs

Last updated：2024-02-01 21:17:19

Under the corporate account CompanyExample (ownerUin is 12345678), there is a sub-account named Developer. This sub-account requires the permission to query CVM instances of the CompanyExample's CVM service, but does not have the permissions of creation, deletion, or powering-on/off.

Scenario A:

The corporate account CompanyExample directly grants the preset policy QcloudCVMInnerReadOnlyAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
 {
   "version": "2.0",
   "statement":[
    {
      "effect": "allow",
      "action": [
           "cvm:Describe*",
           "cvm:Inquiry*"
           ],
      "resource": "*"
    }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Read-only Access to CVM-related Resources

Last updated：2024-02-01 21:18:00

Under the corporate account CompanyExample (ownerUin 12345678), there is a sub-account named Developer. This sub-account requires the permission to query CVM services and related resources (VPC, CLB) of the corporate account CompanyExample, but does not possess the permissions of creation, deletion, or powering-on/off.

Scenario A:

The corporate account CompanyExample directly authorizes the preset policy QcloudCVMReadOnlyAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax,

```
{
  "version": "2.0",
  "statement": [
    {
      "action": [
        "cvm:Describe*",
        "cvm:Inquiry*"
      ],
      "resource": "*",
      "effect": "allow"
    },
    {
      "action": [
        "vpc:Describe*",
        "vpc:Inquiry*",
        "vpc:Get*"
      ],
      "resource": "*",
      "effect": "allow"
    },
    {
      "action": [
        "clb:Describe*"
      ],
      "resource": "*",
      "effect": "allow"
    },
    {
      "effect": "allow",
      "action": "monitor:*",
      "resource": "*"
    }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Access to Perform Operations on CBSs

Last updated：2024-02-01 21:18:39

Under the corporate account CompanyExample (ownerUin 12345678), there is a sub-account Developer. This sub-account requires permissions to view cloud disk information, create cloud disks, and use cloud disks in the CVM Console under the CVM service of the corporate account CompanyExample.

Scenario A:

The corporate account CompanyExample directly grants the preset policy QcloudCBSFullAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement": [
    {
      "action": [
        "cvm:CreateCbsStorages",
        "cvm:AttachCbsStorages",
        "cvm:DetachCbsStorages",
        "cvm:ModifyCbsStorageAttributes",
        "cvm:DescribeCbsStorages",
        "cvm:DescribeInstancesCbsNum",
        "cvm:RenewCbsStorage",
        "cvm:ResizeCbsStorage"
        ],
      "resource": "*",
      "effect": "allow"
    }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

> **Note**
> If the sub-account is not permitted to modify cloud disk attributes, remove cvm:ModifyCbsStorageAttributes from the above policy syntax.

Authorizing Sub-account Access to Perform Operations on Security Groups

Last updated：2024-02-01 21:19:17

Under the corporate account CompanyExample (ownerUin 12345678), there is a sub-account named Developer. This sub-account requires the permission to view the security group in the CVM console of the corporate account CompanyExample, and to utilize the security group's permissions.
The following policy gives the sub-account permission to create and delete security groups in the CVM Console.

1. Create the following policy using policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "action": [
            "cvm:DeleteSecurityGroup",
            "cvm:CreateSecurityGroup"
         ],
         "resource": "*",
         "effect": "allow"
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

The following policy grants the sub-account permission to create, delete, and modify security group policies in the CVM Console.

1. Create the following policy using policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "action": [
            "cvm:ModifySecurityGroupPolicy",
            "cvm:CreateSecurityGroupPolicy",
            "cvm:DeleteSecurityGroupPolicy"
         ],
         "resource": "*",
         "effect": "allow"
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Access to Perform Operations on EIPs

Last updated: 2024-02-01 21:20:41

Under the corporate account CompanyExample (ownerUin 12345678), there is a sub-account named Developer. This sub-account requires the permission to view the elastic IP addresses in the CVM console, and to use the elastic IP addresses under the CVM service of the corporate account CompanyExample.

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "action": [
            "cvm:AllocateAddresses",
            "cvm:AssociateAddress",
            "cvm:DescribeAddresses",
            "cvm:DisassociateAddress",
            "cvm:ModifyAddressAttribute",
            "cvm:ReleaseAddresses"
         ],
         "resource": "*",
         "effect": "allow"
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

The following policy allows the sub-account to view elastic IP addresses, assign it to instances, and associate it with them. The sub-account can modify the properties of elastic IP addresses, disassociate elastic IP addresses, or release elastic IP addresses.

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "action": [
            "cvm:DescribeAddresses",
            "cvm:AllocateAddresses",
            "cvm:AssociateAddress"
         ],
         "resource": "*",
         "effect": "allow"
      }
```

```
    ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Access to Perform Operations on Specific CVM

Last updated：2024-02-01 21:21:19

Under the corporate account CompanyExample (with ownerUin as 12345678), there is a sub-account named Developer. This sub-account requires operation permissions for specific CVMs under the corporate account CompanyExample. These CVMs are all tagged game&webpage.

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "effect": "allow",
         "action": [
            "cvm:*",
            "vpc:DescribeVpcEx",
            "vpc:DescribeNetworkInterfaces"
         ],
         "resource": "*",
         "condition": {
            "for_any_value:string_equal": {
               "qcs:resource_tag": [
                  "game&webpage"
               ]
            }
         }
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Access to Perform Operations on CVMs in Specific Region

Last updated：2024-02-01 21:21:53

Under the corporate account CompanyExample (with ownerUin as 12345678), there is a sub-account named Developer. This sub-account requires operation permissions for all machines in the Guangzhou region under the corporate account CompanyExample.

Solution A:

The corporate account CompanyExample directly authorizes the preset policy QcloudCVMReadOnlyAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Solution B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement": [
    {
      "action": "cvm:*",
      "resource": "qcs::cvm:gz::*",
      "effect": "allow"
    }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Full Access to CVMs Except Payment

Last updated：2024-02-01 21:22:18

Under the corporate account CompanyExample (with ownerUin 12345678), there is a sub-account named Developer. This sub-account requires full management permissions (including creation, management, and all other operations) for the CVM services of the corporate account CompanyExample, but payment permissions are not included. That is, it can place orders but cannot make payments.

Scenario A:

The corporate account CompanyExample directly grants the preset policy QcloudCVMFullAccess to the sub-account Developer. For the granting method, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

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

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Grant sub-account the permission to manage projects

Last updated：2024-02-01 21:22:39

Under the corporate account CompanyExample (with a primary account ID of 12345678), there is a sub-account named Developer. This sub-account requires project-based permission to manage resources via the console.

1. Create a custom policy for project management based on business permissions, as detailed in Create Project Policy.
2. Associate the created custom policy with the sub-account, as detailed in Grant Project Policy.

If the sub-account encounters a lack of permissions while managing a project, such as when attempting to view snapshots, images, VPCs, or Elastic Public IPs, you can grant the sub-account the preset policies `QcloudCVMAccessForNullProject`, `QcloudCVMOrderAccess`, and `QcloudCVMLaunchToVPC`. For the method of authorization, please refer to Authorization Management.

VPC

Granting operational permissions for a specific VPC to the sub-account

Last updated：2024-02-01 21:23:08

Under the corporate account CompanyExample (with ownerUin 12345678), there is a sub-account named Developer. This sub-account requires operation permissions for a specific VPC (with ID vpc-id1) under the VPC service of the corporate account CompanyExample, as well as the network resources under this VPC (such as subnets, routing tables, and so on, excluding CVM, databases, and so on).

1. Create a policy through policy syntax.

```
{
    "version": "2.0",
    "statement": [
        {
            "action": "vpc:*",
            "resource": "*",
            "effect": "allow",
            "condition": {
                "string_equal_if_exist": {
                    "vpc:vpc": [
                    "vpc-id1"
                    ],
                    "vpc:accepter_vpc": [
                     "vpc-id1"
                    ],
                     "vpc:requester_vpc": [
                     "vpc-id1"
                    ]
                }
            }
        }
    ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Access to Perform Operations on VPC Except on Routing Table

Last updated：2024-02-01 21:23:29

Under the corporate account CompanyExample (ownerUin 12345678), there is a sub-account named Developer. This sub-account requires read/write permissions for the VPC service and its related resources of the corporate account CompanyExample, but it is not permitted to perform operations related to the routing table.

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "action": [
            "vpc:*"
         ],
         "resource": "*",
         "effect": "allow"
      },
      {
         "action": [
            "vpc:AssociateRouteTable",
            "vpc:CreateRoute",
            "vpc:CreateRouteTable",
            "vpc:DeleteRoute",
            "vpc:DeleteRouteTable",
            "vpc:ModifyRouteTableAttribute"
         ],
         "resource": "*",
         "effect": "deny"
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Access to Perform Operations on VPN

Last updated: 2024-02-01 21:23:55

Under the corporate account CompanyExample (ownerUin 12345678), there is a sub-account named Developer. This sub-account requires the permission to view all VPC resources and only to perform adding, deleting, modifying, and querying operations on VPN of the corporate account CompanyExample.

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "action": [
            "vpc:Describe*",
            "vpc:Inquiry*",
            "vpc:Get*"
         ],
         "resource": "*",
         "effect": "allow"
      },
      {
         "action": [
            "vpc:Vpn",
            "vpc:UserGw"
         ],
         "resource": "*",
         "effect": "allow"
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing Sub-account Full Access to VPCs

Last updated：2024-02-01 21:24:15

Under the corporate account CompanyExample (with ownerUin as 12345678), there is a sub-account named Developer. This sub-account requires full management permissions (including all operations such as creation, management, and VPC order payment) for the VPC service of the corporate account CompanyExample.

Scenario A:

The corporate account CompanyExample directly grants the preset policies QcloudVPCFullAccess and QcloudVPCFinanceAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
  "version": "2.0",
  "statement":[
     {
       "effect": "allow",
       "action": "vpc:*",
       "resource": "*"
     },
     {
         "effect": "allow",
         "action": "finance:*",
         "resource": "qcs::vpc:::*"
     }
  ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

Authorizing a Sub-account Full Access to VPCs Except Payment

Last updated: 2024-02-01 21:24:38

Under the corporate account CompanyExample (with ownerUin as 12345678), there is a sub-account named Developer. This sub-account requires full administrative permissions (including creation, management, and all other operations) for the VPC services of the corporate account CompanyExample, excluding payment permissions. That is, it can place orders but cannot process payments.

Scenario A:

The corporate account CompanyExample directly grants the preset policy QcloudVPCFullAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

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

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

VOD

Authorizing a Sub-account with Full Permissions to Manage VOD Services

Last updated：2024-02-01 21:25:04

Under the corporate account CompanyExample (with ownerUin 12345678), there is a sub-account named Developer. This sub-account requires full management permissions for the VOD service of the corporate account CompanyExample.

Scenario A:

The corporate account CompanyExample directly grants the preset policy QcloudVODFullAccess to the sub-account Developer. For the method of authorization, please see Authorization Management.

Scenario B:

1. Create a policy through policy syntax.

```
{
   "version": "2.0",
   "statement": [
      {
         "action": [
            "vod:*"
         ],
         "resource": "*",
         "effect": "allow"
      },
      {
         "action": "cos:*",
         "resource": "qcs::cos::uid/10022853:*",
         "effect": "allow"
      }
   ]
}
```

2. Grant this policy to the sub-account. For the authorization method, please see Authorization Management.

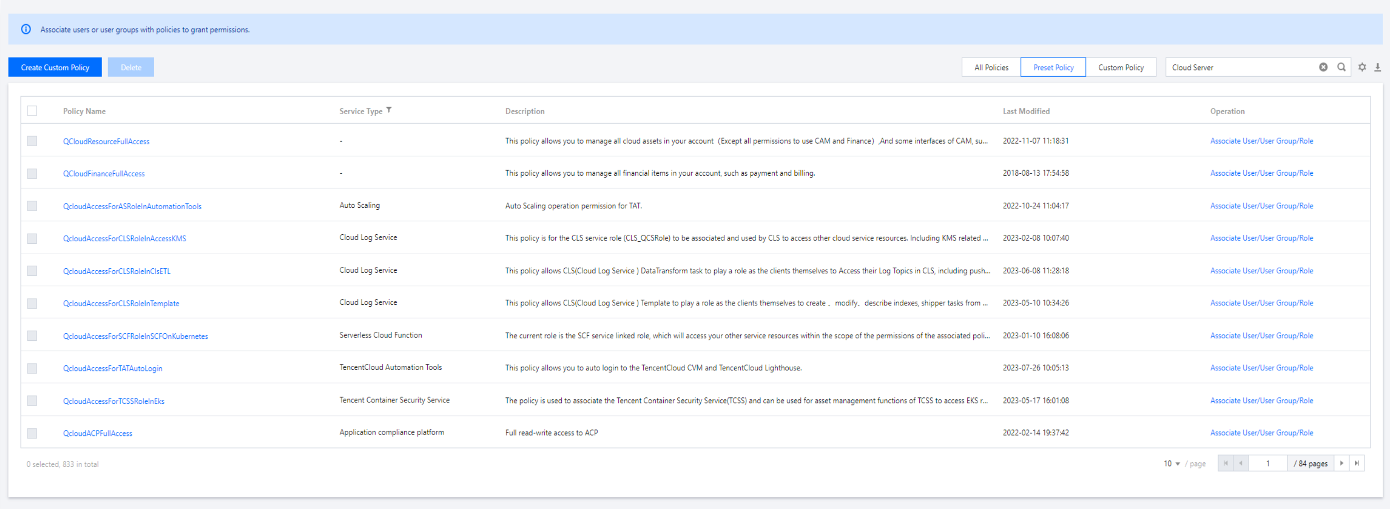
Others

Granting Management or Read-Only Permissions for Specified Product

Last updated：2024-02-01 21:25:33

To streamline the configuration of user permissions, Tencent Cloud products provide default permission granting policies when integrating with Cloud Access Management (CAM). These policies can be directly associated with CAM sub-accounts or user groups to control access to corresponding product services. These policies fall under the category of preset CAM policies, with each type of product service typically offering at least management and read-only policies.

1. Navigate to the **Access Management** > Policies console. Enter the product name (such as Cloud Server) in the search box to view the preset policy list for the corresponding product.



	Policy Name	Service Type	Description	Last Modified	Operation
□	QCloudResourceFullAccess	-	This policy allows you to manage all cloud assets in your account（Except all permissions to use CAM and Finance）,And some interfaces of CAM, su...	2022-11-07 11:18:31	Associate User/User Group/Role
□	QCloudFinanceFullAccess	-	This policy allows you to manage all financial items in your account, such as payment and billing.	2018-08-13 17:54:58	Associate User/User Group/Role
□	QcloudAccessForASRoleInAutomationTools	Auto Scaling	Auto Scaling operation permission for TAT.	2022-10-24 11:04:17	Associate User/User Group/Role
□	QcloudAccessForCLSRoleInAccessKMS	Cloud Log Service	This policy is for the CLS service role (CLS_QCSRole) to be associated and used by CLS to access other cloud service resources. Including KMS related ...	2023-02-08 10:07:40	Associate User/User Group/Role
□	QcloudAccessForCLSRoleInClsETL	Cloud Log Service	This policy allows CLS(Cloud Log Service) DataTransform task to play a role as the clients themselves to Access their Log Topics in CLS, including push...	2023-06-08 11:28:18	Associate User/User Group/Role
□	QcloudAccessForCLSRoleInTemplate	Cloud Log Service	This policy allows CLS(Cloud Log Service) Template to play a role as the clients themselves to create 、modify、describe indexes, shipper tasks from ...	2023-05-10 10:34:26	Associate User/User Group/Role
□	QcloudAccessForSCFRoleInSCFOnKubernetes	Serverless Cloud Function	The current role is the SCF service linked role, which will access your other service resources within the scope of the permissions of the associated poli...	2023-01-10 16:08:06	Associate User/User Group/Role
□	QcloudAccessForTATAutoLogin	TencentCloud Automation Tools	This policy allows you to auto login to the TencentCloud CVM and TencentCloud Lighthouse.	2023-07-26 10:05:13	Associate User/User Group/Role
□	QcloudAccessForTCSSRoleInEks	Tencent Container Security Service	The policy is used to associate the Tencent Container Security Service(TCSS) and can be used for asset management functions of TCSS to access EKS r...	2023-05-17 16:01:08	Associate User/User Group/Role
□	QcloudACPFullAccess	Application compliance platform	Full read-write access to ACP	2022-02-14 19:37:42	Associate User/User Group/Role

0 selected, 833 in total

10 / page 1 / 84 pages

Among them, QcloudCVMFullAccess is the management policy, and QcloudCVMInnerReadOnlyAccess is the read-only policy.

> **Note**
> The management policies of some services do not include payment permissions. You can associate a default payment management policy (such as `QcloudCVMFullAccess`) with the CAM sub-user/user group you want to authorize.

2. Grant the aforementioned policies to the CAM sub-account/user group. For the method of authorization, please see Authorization Management.
   - If you need to grant a sub-user full management permissions for a Tencent Cloud account, you can use the preset policy `AdministratorAccess`.
     `AdministratorAccess`: This policy allows you to manage all users within the account, their permissions, financial-related information, and cloud service assets.
   - If you need to grant read-only access to a Tencent Cloud account to a sub-user, you can use the preset policy `ReadOnlyAccess`.

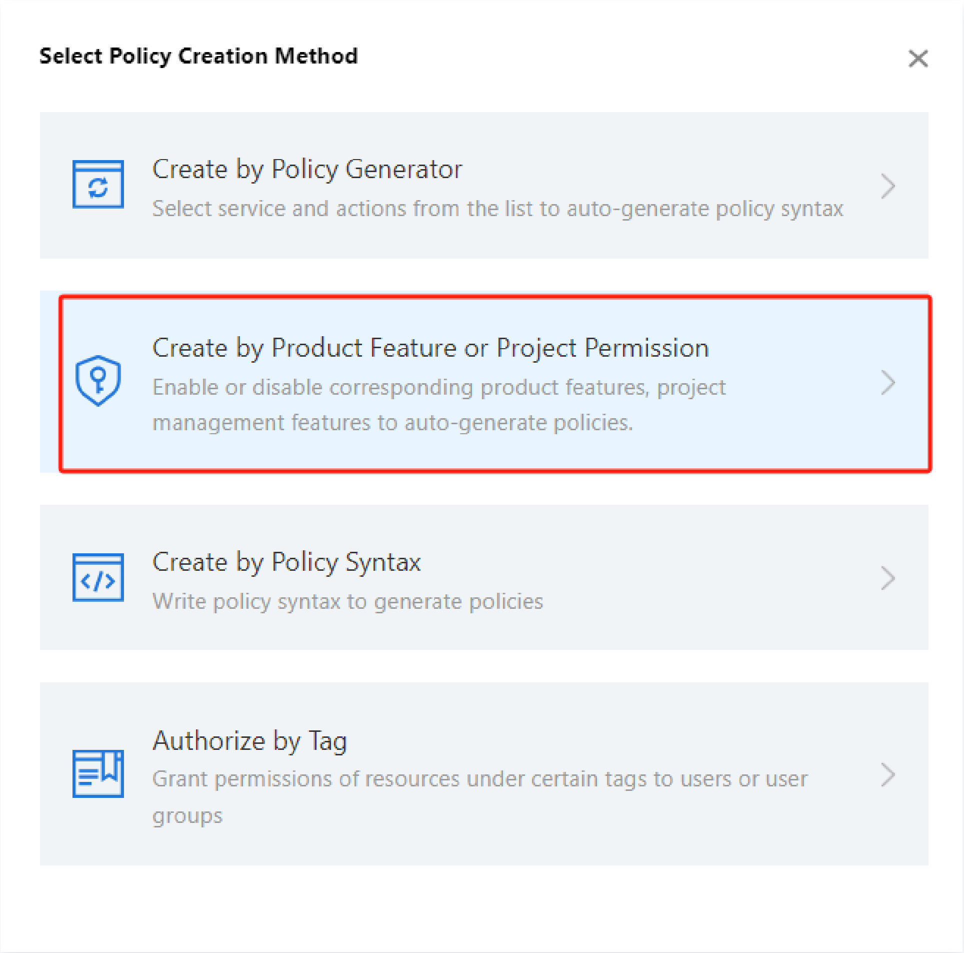
`ReadOnlyAccess` : This policy allows you to have read-only access to all cloud service assets within the account that support interface-level or resource-level authentication.

Authorizing Sub-Accounts to Manage Project

Last updated: 2024-02-01 21:44:37

Under the corporate account CompanyExample, there is a sub-account Developer, which requires full access permissions to a specified project under the corporate account CompanyExample.

1. Navigate to the Access Management Console. In the left sidebar, click **Policies** to navigate to the policy management page.
2. On the policy management page, click **Create Custom Policy**.
3. In the **Select Policy Creation Method** pop-up window, choose **Create by Product Features or Project Permissions** as shown in the image below:



Select Policy Creation Method

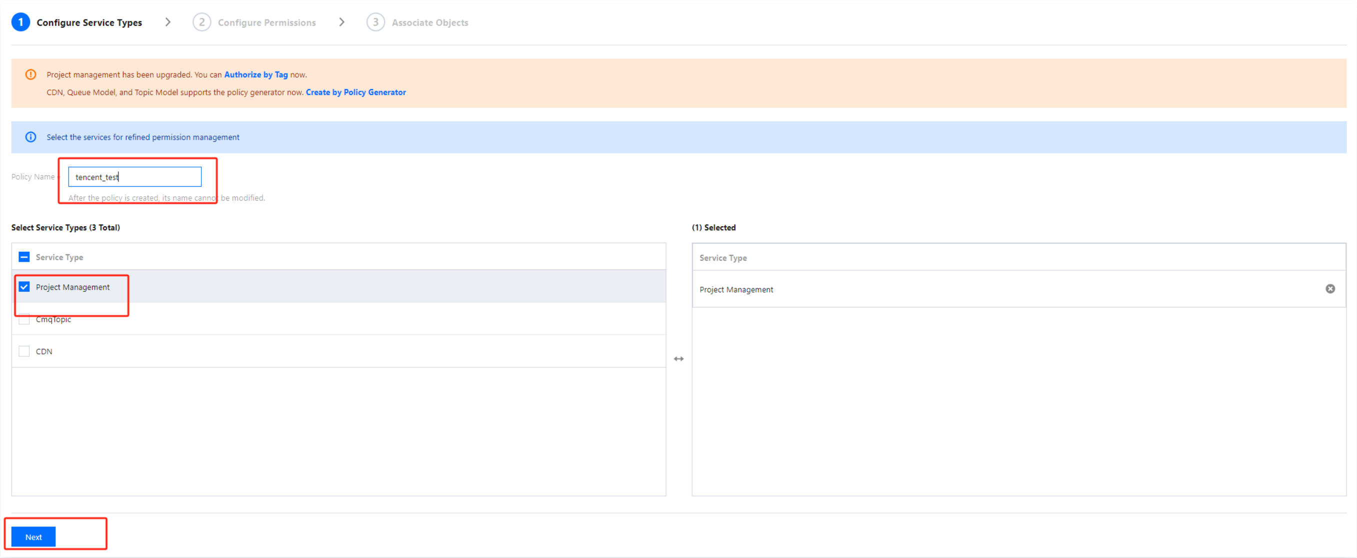
Create by Policy Generator
Select service and actions from the list to auto-generate policy syntax

Create by Product Feature or Project Permission
Enable or disable corresponding product features, project management features to auto-generate policies.

Create by Policy Syntax
Write policy syntax to generate policies

Authorize by Tag
Grant permissions of resources under certain tags to users or user groups

4. On the **Create by Product Features or Project Permissions** page, enter the policy name, and in the **Select Service Type**, check **Project Management**, then click **Next**.



1 Configure Service Types > 2 Configure Permissions > 3 Associate Objects

Project management has been upgraded. You can **Authorize by Tag** now.
CDN, Queue Model, and Topic Model supports the policy generator now. **Create by Policy Generator**

Select the services for refined permission management

Policy Name: tencent_test
After the policy is created, its name cannot be modified.

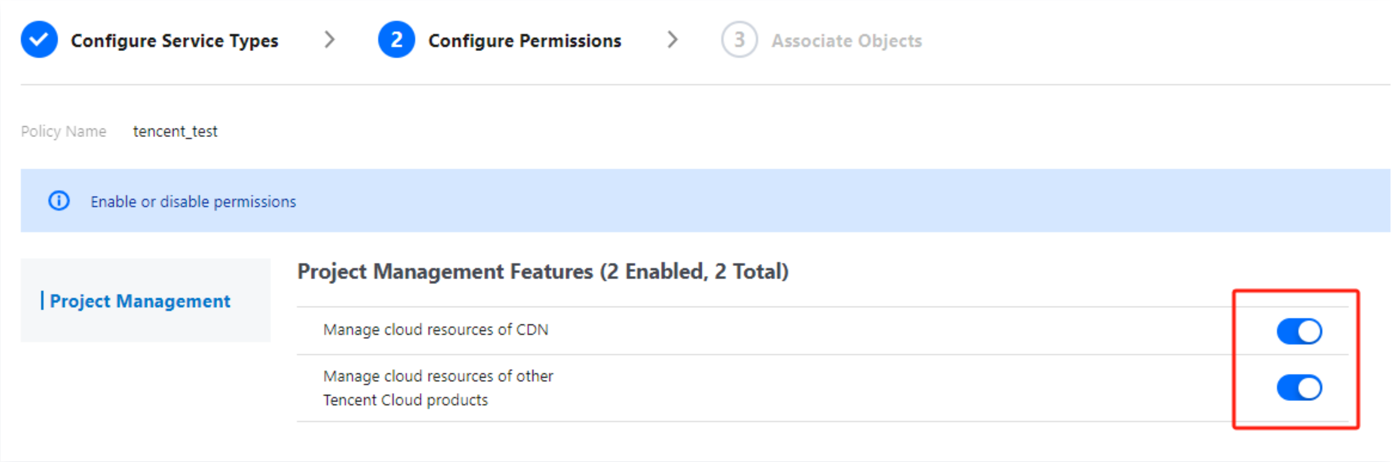
Select Service Types (3 Total)

- ☐ Service Type
- ☑ Project Management
- ☐ CmqTopic
- ☐ CDN

(1) Selected

Service Type
Project Management

5. Configure the Project Management feature according to actual requirements, as depicted in the following illustration:



✓ Configure Service Types > 2 Configure Permissions > 3 Associate Objects

Policy Name: tencent_test

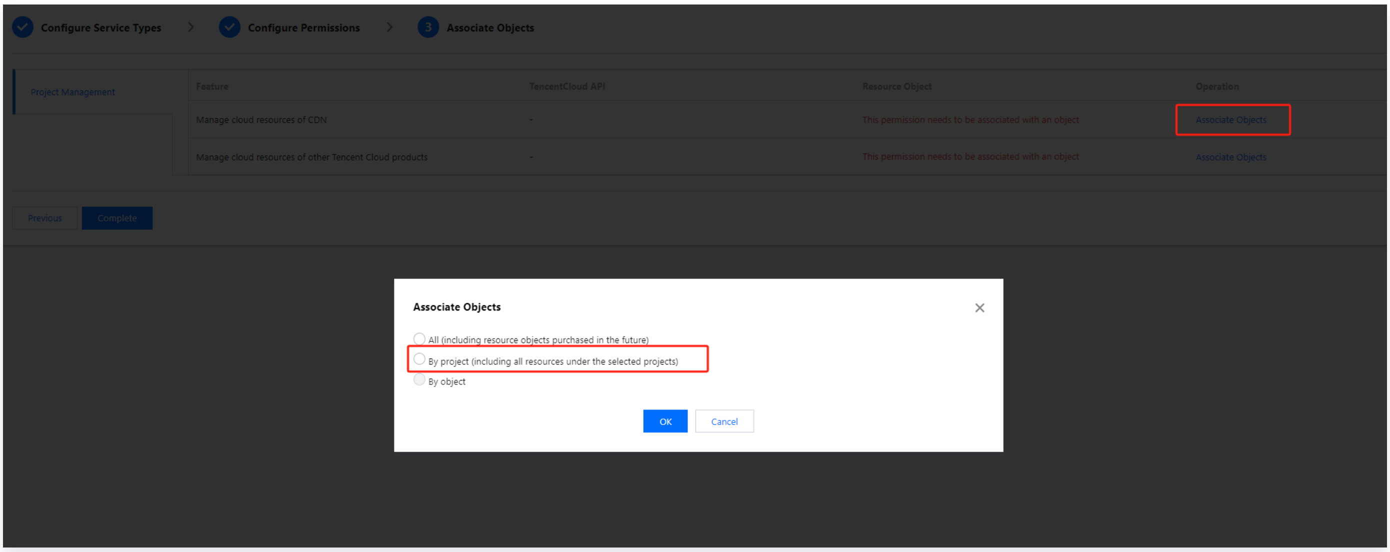
Enable or disable permissions

Project Management Features (2 Enabled, 2 Total)

Manage cloud resources of CDN	On
Manage cloud resources of other Tencent Cloud products	On

   - If management is required for CDN-related project cloud resources, set the CDN permission switch to **On**.
   - If management is required for other service-related project cloud resources, set other permission switches to **On**.

6. Click **Next**.
7. Click on **Associate Object**, then select **Select by Project** as shown in the image below:



Feature	TencentCloud API	Resource Object	Operation
Manage cloud resources of CDN	-	This permission needs to be associated with an object	Associate Objects
Manage cloud resources of other Tencent Cloud products	-	This permission needs to be associated with an object	Associate Objects

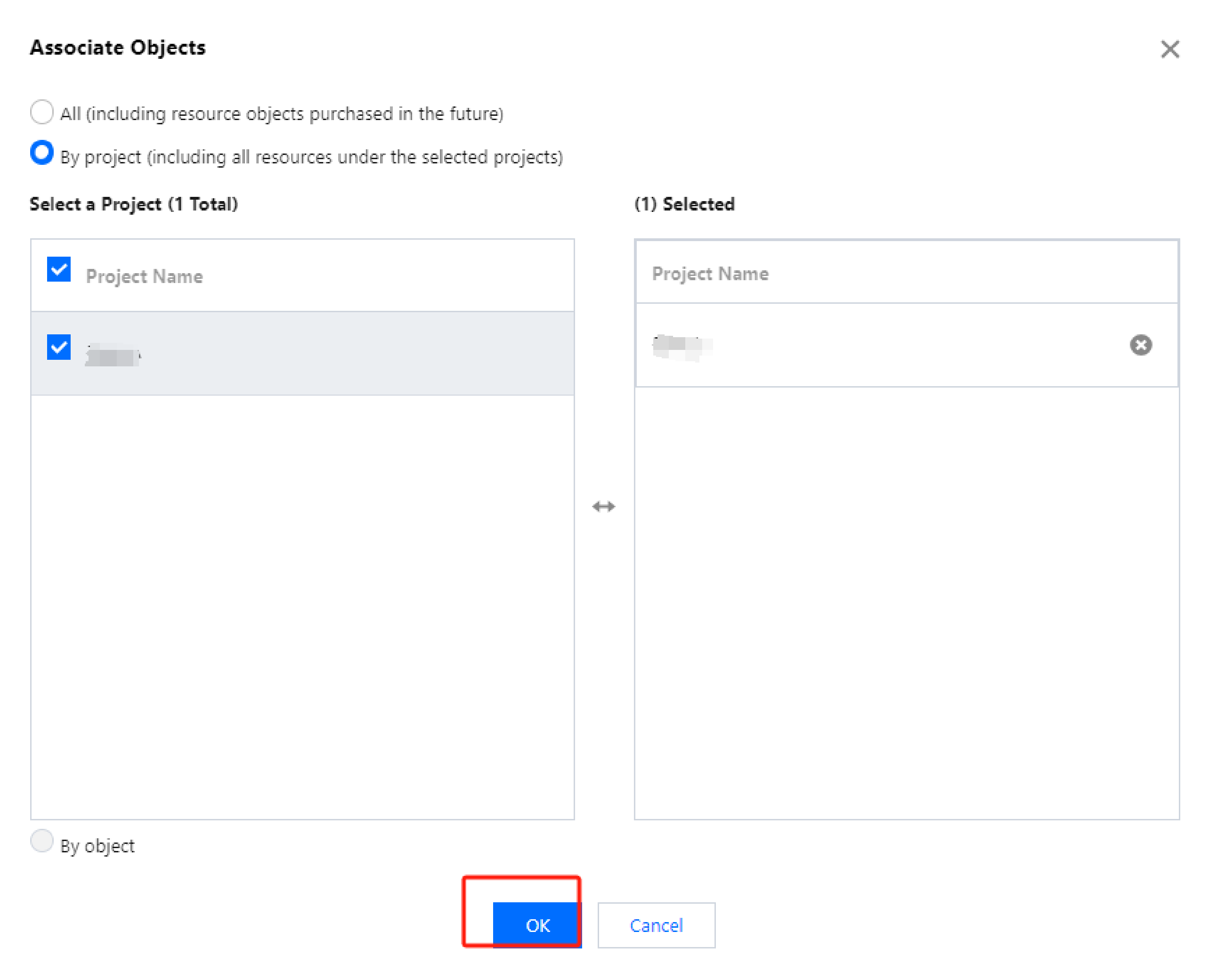
Associate Objects

○ All (including resource objects purchased in the future)

○ By project (including all resources under the selected projects)

○ By object

8. Select the project you want to associate, then click OK. As shown in the image below:



Associate Objects

○ All (including resource objects purchased in the future)

● By project (including all resources under the selected projects)

Select a Project (1 Total)

☑ Project Name
☑ [redacted]

(1) Selected

Project Name
[redacted]

○ By object

9. Click **Finish**.

Note

> Currently, it is not possible to implement refined permission management for projects. If differentiated permission management for resources within a project is required, it is recommended to grant permissions to resources individually through policy syntax. In the future, tag-based methods will be used for resource permission management.