ALIBABA CLOUD

Alibaba Cloud

物联网平台

Authorization

Document Version: 20220526

Alibaba Cloud

Legal disclaimer

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

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

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

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

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

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

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

Document conventions

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

Table of Contents

1.Log on to the console by using an Alibaba Cloud account

An Alibaba Cloud account has full permissions on all resources that belong to this account and allows you to modify account information.

Log on to the IoT Platform console by using an Alibaba Cloud account

Before you use an Alibaba Cloud account to log on to the IoT Platform console, you must complete real-name verification to obtain full permissions on IoT Platform.

1. Go to the Alibaba Cloud international site (alibabacloud.com).
2. Click **Console**.
3. Log on to the console by using your Alibaba Cloud account and password.

   > **Note** To retrieve an Alibaba Cloud account or password, click **Forgot Password** on the logon page to start the retrieval process.

4. Move the pointer over the upper-left corner in the console and click **Products and Services**. Then, all Alibaba Cloud services are displayed.
5. Enter IoT Platform in the search box and click **IoT Platform** in the search results to go to the **IoT Platform console**.

> **Note** If you have not activated the IoT Platform service, the **IoT Platform console** prompts you to activate this service on the homepage. You can click **Activate Now** to activate this service.

After you log on to the **IoT Platform console**, you can manage products, devices, and rules.

Create RAM users by using an Alibaba Cloud account

The leak of your Alibaba Cloud account information gives rise to serious security risks because the Alibaba Cloud account has full permissions. Therefore, we recommend that you do not disclose your Alibaba Cloud account and password information when you authorize others to access your Alibaba Cloud resources. You can create RAM users and grant them only the required access permissions. Users who are not Alibaba Cloud account owners or administrators can access resources as RAM users. For more information about RAM users, see RAM user access and Custom permissions.

2.Resource Access Management (RAM)

This chapter describes IoT Platform access control.

2.1. RAM and STS

Resource Access Management (RAM) and Security Token Service (STS) are access control systems that are provided by Alibaba Cloud.

For more information about RAM and STS, see the RAM documentation.

RAM is used to control the permissions of accounts. You can use RAM to create and manage RAM users. You can also grant permissions to RAM users to control what resources the RAM users can access.

STS is a security token management system. It is used to manage short-term permissions that are granted to RAM users. You can use STS to grant permissions to temporary users.

Background

RAM and STS enable you to securely grant permissions without the need to expose the AccessKey information of your Alibaba Cloud account. The leak of the AccessKey pair of an Alibaba Cloud account gives rise to serious security risks. Users who obtain the AccessKey pair of an Alibaba Cloud account can manage all resources of the account and steal important information.

RAM is an access control service that is used to manage long-term permissions. The owner of an Alibaba Cloud account can create RAM users and grant different permissions to the RAM users. The AccessKey pairs of RAM users must be kept safe. However, if the AccessKey pair of a RAM user is leaked, only limited information is potentially exposed. RAM users are valid for a long term.

RAM enables you to grant long-term permissions to users, whereas STS enables you to grant short-term permissions to users. You can use STS to obtain temporary AccessKey pairs and tokens. The temporary AccessKey pairs and tokens can be sent to temporary users so that the temporary users can access specific resources. Permissions that are obtained from STS are strictly restricted and have validity periods. This reduces the effects of information leak.

For more information about how to use RAM and STS, see Examples.

Terms

Before you use RAM and STS, we recommend that you have a basic understanding of the following terms:

- RAM user: a user that is created in the RAM console. An independent AccessKey pair is generated for a RAM user during or after the creation of the RAM user. After you create a RAM user, you must configure the password and permissions for the RAM user. Then, the RAM user can perform the authorized operations. A RAM user can be considered a user with specific operation permissions.
- RAM role: an identity that has a set of permissions. RAM roles do not have independent logon passwords and AccessKey pairs. RAM users can assume RAM roles. After a RAM role is assigned to a RAM user, the RAM user has the permissions of the RAM role.
- Policy: a set of permissions. For example, a policy defines the permissions that allow a RAM user to read or write specific resources.
- Resource: cloud resources that are accessible to RAM users, such as all Tablestore instances, a

Tablestore instance, or a table in a Tablestore instance.

The relationship between RAM users and RAM roles is similar to the relationship between individuals and their identities. For example, a person might be an employee at work and a father at home. A person has different identities in different scenarios. When a person uses an identity, the person has the permissions of the identity. A RAM role is not an entity that can perform operations. To perform the operations that a RAM role allows, you must assign the RAM role to a user. A RAM role can be assumed by multiple users.

Examples

To prevent the security risks that are caused by the leak of the AccessKey pair of an Alibaba Cloud account, you, an Alibaba Cloud account administrator, create two RAM users. One of them is named A and the other is named B. An independent AccessKey pair is generated for each of them. A has the read permission and B has the write permission. You can revoke the permissions from the RAM users at any time in the RAM console.

To meet business requirements, you want to grant users short-term permissions to access the IoT Platform API. In this case, we recommend that you do not disclose the AccessKey pair of A. We recommend that you create a RAM role C and grant C the permission to access the IoT Platform API. Note that C cannot be directly used because no AccessKey pair is configured for C. C is only a virtual entity that has the permission to access the IoT Platform API.

You must call the AssumeRole operation of STS to obtain temporary identity credentials that are required to access the IoT Platform API. When you call the AssumeRole operation, you must set the RoleArn parameter to the Alibaba Cloud Resource Name (ARN) of C. If the call is successful, STS returns the temporary AccessKey ID, AccessKey secret, and token as temporary identity credentials. The validity period of these credentials can be specified when you call the AssumeRole operation. You can deliver these credentials to the users who need to access the IoT Platform API. This access permission is temporary.

Why is it complicated to use RAM and STS?

The terms and use of RAM and STS are complicated. They deliver high account security and flexible access control at the cost of ease of use.

RAM allows you to create RAM users and RAM roles to separate the entities that perform operations from the virtual entities that define a set of permissions. A user who needs multiple permissions, such as the read and write permissions, may use only one permission at a time. In this case, you can create two RAM roles and grant one of them the read permission and the other the write permission. Then, you can create a RAM user and assign the two roles to the RAM user. When the RAM user needs the read permission, the RAM user assumes the RAM role that has the read permission. When the RAM user needs the write permission, the RAM user assumes the RAM role that has the write permission. This reduces the risks that are caused by unauthorized permissions in each operation. In addition, you can assign a RAM role to other Alibaba Cloud accounts and RAM users to grant them the permissions of the RAM role. This facilitates collaboration.

STS enables more flexible access control. For example, you can configure the validity period for credentials. If long-term credentials are required, you do not need to use STS. In this case, you can use only RAM to manage RAM users.

The following articles provide guidelines and examples to describe how to use RAM and STS. For more information about the code of RAM and STS, see the API reference of RAM API and STS API.

2.2. Custom permissions

You can define permissions to allow or deny operations on resources in specified conditions.

Procedure

Permissions are defined in Resource Access Management (RAM) policies. You can define custom permissions by creating custom policies.

1. Log on to the RAM console.
2. In the left-side navigation pane, choose **Permissions > Policies**.
3. On the **Policies** page, click **Create Policy**.
4. On the **Create Policy** page, click the **JSON** tab.
5. Enter the policy content and click **Next Step**.

   Configure the policy in the JSON format. The following parameters are required:

   - Action: the actions that you want to authorize. IoT Platform actions start with `iot:`. For more information about actions and examples, see the "Define actions" section of this article.
   - Effect: the authorization type. Valid values: *Allow* and *Deny*.
   - Resource: the resources that you want to authorize.

     If you want to authorize a RAM user to access all resources of your IoT Platform service, set this parameter to `*`.
   - Condition: the condition. For more information, see the "Define conditions" section of this article.
6. Specify the **Name** and **Note** parameters, and then click **OK**.

Define actions

To define actions for a policy, you must specify API operations in the Action parameter. When you create a policy to grant permissions on IoT Platform, specify IoT Platform actions in the Action parameter. Each IoT Platform action must start with `iot:`. Multiple actions must be separated by commas (,). You can set the Action parameter to an asterisk (*), which indicates a wildcard. For information about the API operations of IoT Platform, see Mapping between IoT Platform API operations and RAM policies.

The following examples show how to define actions.

- Specify a single API operation to define an action.

```
"Action": "iot:CreateProduct"
```

- Specify multiple API operations to define actions.

```
"Action": [
"iot:UpdateProduct",
"iot:QueryProduct"
]
```

- Specify all read-only API operations to define actions, including the actions that are performed when the rules engine forwards data of a product.

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

```
        "iot:Query*",
        "iot:List*",
        "iot:Get*",
        "iot:BatchGet*",
        "iot:Check*"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "rds:DescribeDBInstances",
        "rds:DescribeDatabases",
        "rds:DescribeAccounts",
        "rds:DescribeDBInstanceNetInfo"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": "ram:ListRoles",
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "mns:ListTopic",
        "mns:GetTopicRef"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "ots:ListInstance",
        "ots:GetInstance",
        "ots:ListTable",
        "ots:DescribeTable"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "fc:ListServices",
        "fc:GetService",
        "fc:GetFunction",
        "fc:ListFunctions"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
```

```
        "log:ListShards",
        "log:ListLogStores",
        "log:ListProject"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "cms:QueryMetricList"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

- Specify all read/write API operations to define actions, including the actions that are performed when the rules engine forwards data of a product.

```
{
  "Version": "1",
  "Statement": [
    {
      "Action": "iot:*",
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "rds:DescribeDBInstances",
        "rds:DescribeDatabases",
        "rds:DescribeAccounts",
        "rds:DescribeDBInstanceNetInfo",
        "rds:ModifySecurityIps"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": "ram:ListRoles",
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "mns:ListTopic",
        "mns:GetTopicRef"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "ots:ListInstance",
```

```
        "ots:ListTable",
        "ots:DescribeTable",
        "ots:GetInstance"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "fc:ListServices",
        "fc:GetService",
        "fc:GetFunction",
        "fc:ListFunctions"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "log:ListShards",
        "log:ListLogStores",
        "log:ListProject"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": "ram:PassRole",
      "Resource": "*",
      "Effect": "Allow",
      "Condition": {
        "StringEquals": {
          "acs:Service": "iot.aliyuncs.com"
        }
      }
    },
    {
      "Action": [
        "cms:QueryMetricList"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

Define conditions

RAM policies support multiple conditions. For example, you can set limits on the access IP addresses and access time. You can also specify whether HTTPS-based access is allowed, and whether multi-factor authentication (MFA) is required. All API operations of IoT Platform support these conditions.

- IP address-based access control

RAM allows you to specify the source IP addresses from which requests are allowed. You can also use Classless Inter-Domain Routing (CIDR) blocks to specify source IP addresses. The following examples show how to set limits on access IP addresses.

- Specify a single IP address or CIDR block. In this example, only requests from the IP address 192.0.2.1 or CIDR block 198.51.100.0/24 are allowed.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*",
      "Condition": {
        "IpAddress": {
          "acs:SourceIp": [
            "192.0.2.1",
            "198.51.100.0/24"
          ]
        }
      }
    }
  ],
  "Version": "1"
}
```

- Specify multiple IP addresses. In this example, only requests from IP addresses 192.0.2.1 and 198.51.100.1 are allowed.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*",
      "Condition": {
        "IpAddress": {
          "acs:SourceIp": [
            "192.0.2.1",
            "198.51.100.1"
          ]
        }
      }
    }
  ],
  "Version": "1"
}
```

- HTTPS-based access control

  RAM allows you to specify whether resources must be requested over HTTPS.

  In this example, only requests over HTTPS are allowed.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*",
      "Condition": {
        "Bool": {
          "acs:SecureTransport": "true"
        }
      }
    }
  ],
  "Version": "1"
}
```

- MFA-based access control

  RAM allows you to specify whether to enable MFA for requests. MFA applies to console logon and is not required for API requests.

  In this example, only requests that pass MFA are allowed.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*",
      "Condition": {
        "Bool": {
          "acs:MFAPresent ": "true"
        }
      }
    }
  ],
  "Version": "1"
}
```

- Time-based access control

  RAM allows you to specify the access time. Requests that are sent earlier than the specified time are allowed or denied.

  In this example, only requests that are sent earlier than 00:00:00 on January 1, 2019 (UTC+8) are allowed.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*",
      "Condition": {
        "DateLessThan": {
          "acs:CurrentTime": "2019-01-01T00:00:00+08:00"
        }
      }
    }
  ],
  "Version": "1"
}
```

Scenarios

Based on the Action, Resource, and Condition parameters that are described in the preceding sections, this section describes the scenarios of custom policies.

- A custom policy that allows specific requests

  Scenario: Only requests that are sent over HTTPS, from the CIDR block 10.101.168.111/24, and earlier than 00:00:00 on January 1, 2019 (UTC+8) are allowed.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*",
      "Condition": {
        "IpAddress": {
          "acs:SourceIp": [
            "192.0.2.1/24"
          ]
        },
        "DateLessThan": {
          "acs:CurrentTime": "2019-01-01T00:00:00+08:00"
        },
        "Bool": {
          "acs:SecureTransport": "true"
        }
      }
    }
  ],
  "Version": "1"
}
```

- A custom policy that denies specific requests

  Scenario: Read requests from the IP address 10.101.169.111 are denied.

```
{
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "iot:Query*",
        "iot:List*",
        "iot:Get*",
        "iot:BatchGet*"
      ],
      "Resource": "*",
      "Condition": {
        "IpAddress": {
          "acs:SourceIp": [
            "198.51.100.1"
          ]
        }
      }
    }
  ],
  "Version": "1"
}
```

After a policy is created, attach the policy to RAM users. Then, the RAM users can perform the operations that are defined in the policy. For more information about how to create and authorize RAM users, see RAM user access.

2.3. Mapping between IoT Platform API operations and RAM policies

You can create a custom policy for IoT Platform and attach the policy to a RAM user. This policy grants the RAM user the permissions to call the specified API operation of IoT Platform.

For more information about how to grant permissions to a RAM user, see Custom permissions.

The following table describes the valid values of the Action element that you must configure when you create a custom Resource Access Management (RAM) policy for IoT Platform.

> **Note** The following table describes some API operations that you can specify in RAM policies. The API operations must be specified in the `iot:${API operation name}` format. `${API operation name}` specifies the name of an API operation that you want to specify.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
CreateProduct	iot:CreateProduct	*	Creates a product.
UpdateProduct	iot:UpdateProduct	*	Modifies the details of a product.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
QueryProduct	iot:QueryProduct	*	Queries the details of a product.
QueryProductList	iot:QueryProductList	*	Queries products.
DeleteProduct	iot:DeleteProduct	*	Deletes a product.
CreateProductTags	iot:CreateProductTags	*	Creates product tags.
UpdateProductTags	iot:UpdateProductTags	*	Modifies the tags of a product.
DeleteProductTags	iot:DeleteProductTags	*	Deletes product tags.
ListProductTags	iot:ListProductTags	*	Queries product tags.
ListProductByTags	iot:ListProductByTags	*	Queries products by tag.
RegisterDevice	iot:RegisterDevice	*	Registers a device.
QueryDevice	iot:QueryDevice	*	Queries the devices of a specified product.
DeleteDevice	iot:DeleteDevice	*	Deletes a device.
QueryPageByApplyId	iot:QueryPageByApplyId	*	Queries the details of the multiple devices that are registered at the same time.
BatchGetDeviceState	iot:BatchGetDeviceState	*	Queries the statuses of devices.
BatchRegisterDeviceWithApplyId	iot:BatchRegisterDeviceWithApplyId	*	Creates multiple devices by application ID.
BatchRegisterDevice	iot:BatchRegisterDevice	*	Registers multiple devices. Device names are randomly generated.
QueryBatchRegisterDeviceStatus	iot:QueryBatchRegisterDeviceStatus	*	Queries the status of the multiple devices that are registered at the same time.
BatchCheckDeviceNames	iot:BatchCheckDeviceNames	*	Specifies names for multiple devices.
QueryDeviceStatistics	iot:QueryDeviceStatistics	*	Queries device statistics.
QueryDeviceEventData	iot:QueryDeviceEventData	*	Queries the historical events of a device.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
QueryDeviceServiceData	iot:QueryDeviceServiceData	*	Queries the service records of a device.
SetDeviceProperty	iot:SetDeviceProperty	*	Configures properties for a device.
SetDevicesProperty	iot:SetDevicesProperty	*	Configures properties for multiple devices.
InvokeThingService	iot:InvokeThingService	*	Calls a service on a device.
InvokeThingsService	iot:InvokeThingsService	*	Calls a service on multiple devices.
QueryDevicePropertyStatus	iot:QueryDevicePropertyStatus	*	Queries the property snapshot of a device.
QueryDeviceDetail	iot:QueryDeviceDetail	*	Queries the details of a device.
DisableThing	iot:DisableThing	*	Disables a device.
EnableThing	iot:EnableThing	*	Enables a device.
ResetThing	iot:ResetThing	*	Resets a device.
GetThingTopo	iot:GetThingTopo	*	Queries the topological relationships of a device.
RemoveThingTopo	iot:RemoveThingTopo	*	Removes the topological relationships of a device.
NotifyAddThingTopo	iot:NotifyAddThingTopo	*	Adds a topological relationship to IoT Platform.
QueryDevicePropertyData	iot:QueryDevicePropertyData	*	Queries the historical properties of a device.
QueryDevicePropertiesData	iot:QueryDevicePropertiesData	*	Queries the property data of a device.
GetGatewayBySubDevice	iot:GetGatewayBySubDevice	*	Queries the information about a gateway device based on sub-device information.
SaveDeviceProp	iot:SaveDeviceProp	*	Specifies tags for a device.
QueryDeviceProp	iot:QueryDeviceProp	*	Queries the tags of a device.
DeleteDeviceProp	iot:DeleteDeviceProp	*	Deletes the tags of a device.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
QueryDeviceByTags	iot:QueryDeviceByTags	*	Queries devices by tag.
CreateDeviceGroup	iot:CreateDeviceGroup	*	Creates a device group.
UpdateDeviceGroup	iot:UpdateDeviceGroup	*	Modifies the information about a device group.
DeleteDeviceGroup	iot:DeleteDeviceGroup	*	Deletes a device group.
BatchAddDeviceGroupRelations	iot:BatchAddDeviceGroupRelations	*	Adds devices to a device group.
BatchDeleteDeviceGroupRelations	iot:BatchDeleteDeviceGroupRelations	*	Removes a device from a device group.
QueryDeviceGroupInfo	iot:QueryDeviceGroupInfo	*	Queries the details of a device group.
QueryDeviceGroupList	iot:QueryDeviceGroupList	*	Queries device groups.
SetDeviceGroupTags	iot:SetDeviceGroupTags	*	Creates tags for or updates the tags of a device group.
QueryDeviceGroupTagList	iot:QueryDeviceGroupTagList	*	Queries the tags of a device group.
QueryDeviceGroupByDevice	iot:QueryDeviceGroupByDevice	*	Queries the device groups to which a device belongs.
QueryDeviceListByDeviceGroup	iot:QueryDeviceListByDeviceGroup	*	Queries devices in a device group.
QuerySuperDeviceGroup	iot:QuerySuperDeviceGroup	*	Queries the details of a parent device group by sub-group ID.
QueryDeviceGroupByTags	iot:QueryDeviceGroupByTags	*	Queries device groups by tag.
StartRule	iot:StartRule	*	Enables a rule.
StopRule	iot:StopRule	*	Disables a rule.
ListRule	iot:ListRule	*	Queries rules.
GetRule	iot:GetRule	*	Queries the details of a rule.
CreateRule	iot:CreateRule	*	Creates a rule.
UpdateRule	iot:UpdateRule	*	Modifies a rule.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
DeleteRule	iot:DeleteRule	*	Deletes a rule.
CreateRuleAction	iot:CreateRuleAction	*	Creates a data forwarding method for a rule.
UpdateRuleAction	iot:UpdateRuleAction	*	Modifies the data forwarding method of a rule.
DeleteRuleAction	iot:DeleteRuleAction	*	Deletes a data forwarding method from a rule.
GetRuleAction	iot:GetRuleAction	*	Queries the details of a data forwarding method.
ListRuleActions	iot:ListRuleActions	*	Queries the data forwarding methods of a rule.
Pub	iot:Pub	*	Publish messages.
PubBroadcast	iot:PubBroadcast	*	Publishes a message to all devices that subscribe to a topic.
RRpc	iot:RRpc	*	Sends a request to a device and receives a response from the device.
CreateProductTopic	iot:CreateProductTopic	*	Creates a topic category for a product.
DeleteProductTopic	iot:DeleteProductTopic	*	Deletes a topic category.
QueryProductTopic	iot:QueryProductTopic	*	Queries the topic categories of a product.
UpdateProductTopic	iot:UpdateProductTopic	*	Modifies a topic category.
CreateTopicRouteTable	iot:CreateTopicRouteTable	*	Creates routing relationships between topics.
DeleteTopicRouteTable	iot:DeleteTopicRouteTable	*	Deletes a routing relationship.
QueryTopicReverseRouteTable	iot:QueryTopicReverseRouteTable	*	Queries the source topics of a destination topic.
QueryTopicRouteTable	iot:QueryTopicRouteTable	*	Queries the destination topics of a source topic.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
GetDeviceShadow	iot:GetDeviceShadow	*	Queries the details of a device shadow.
UpdateDeviceShadow	iot:UpdateDeviceShadow	*	Modifies a device shadow.
SetDeviceDesiredProperty	iot:SetDeviceDesiredProperty	*	Specifies desired property values for a device.
QueryDeviceDesiredProperty	iot:QueryDeviceDesiredProperty	*	Queries the property values of a device.
BatchUpdateDeviceNickname	iot:BatchUpdateDeviceNickname	*	Modifies the aliases of multiple devices.
QueryDeviceFileList	iot:QueryDeviceFileList	*	Queries the details of all files that are uploaded from a device to IoT Platform.
QueryDeviceFile	iot:QueryDeviceFile	*	Queries the details of a file that is uploaded to IoT Platform from a device.
DeleteDeviceFile	iot:DeleteDeviceFile	*	Deletes a file that is uploaded to IoT Platform from a device.
QueryDeviceCert	iot:QueryDeviceCert	*	Queries the X.509 certificate of a device.
QueryCertUrlByApplyId	iot:QueryCertUrlByApplyId	*	Queries the URL from which you can download the X.509 certificates of registered devices.
BatchAddThingTopo	iot:BatchAddThingTopo	*	Builds topological relationships between multiple sub-devices and a gateway device.
QueryDeviceByStatus	iot:QueryDeviceByStatus	*	Queries devices by status.
GenerateOTAUploadURL	iot:GenerateOTAUploadURL	*	Generates the information that is used to upload firmware files to Object Storage Service (OSS).
CreateOTAFirmware	iot:CreateOTAFirmware	*	Creates a firmware file.
DeleteOTAFirmware	iot:DeleteOTAFirmware	*	Deletes a firmware file.
ListOTAFirmware	iot:ListOTAFirmware	*	Queries all firmware files.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
QueryOTAFirmware	iot:QueryOTAFirmware	*	Queries the details of a firmware file.
CreateOTAVerifyJob	iot:CreateOTAVerifyJob	*	Creates a firmware verification batch.
CreateOTAStaticUpgradeJob	iot:CreateOTAStaticUpgradeJob	*	Creates a static update batch.
CreateOTADynamicUpgradeJob	iot:CreateOTADynamicUpgradeJob	*	Creates a dynamic update batch.
ListOTAJobByFirmware	iot:ListOTAJobByFirmware	*	Queries the update tasks of a firmware file.
ListOTAJobByDevice	iot:ListOTAJobByDevice	*	Queries all firmware update batches of a device.
QueryOTAJob	iot:QueryOTAJob	*	Queries the details of an update task.
CancelOTAStrategyByJob	iot:CancelOTAStrategyByJob	*	Cancels an update policy that is associated with a dynamic update batch.
CancelOTATaskByDevice	iot:CancelOTATaskByDevice	*	Cancels the pending device update tasks of a firmware file.
CancelOTATaskByJob	iot:CancelOTATaskByJob	*	Cancels the device update tasks of an update batch.
ListOTATaskByJob	iot:ListOTATaskByJob	*	Queries the update tasks of a device by update batch.
CreateSubscribeRelation	iot:CreateSubscribeRelation	*	Creates a Message Service (MNS) or Advanced Message Queuing Protocol (AMQP) server-side subscription.
UpdateSubscribeRelation	iot:UpdateSubscribeRelation	*	Modifies an MNS or AMQP server-side subscription.
QuerySubscribeRelation	iot:QuerySubscribeRelation	*	Queries the details of an MNS or AMQP server-side subscription.
DeleteSubscribeRelation	iot:DeleteSubscribeRelation	*	Deletes an MNS or AMQP server-side subscription.

Operation	Action in a RAM policy	Resource in the RAM policy	Description
CreateConsumerGroup	iotCreateConsumerGroup	*	Creates a consumer group that is required by an AMQP server-side subscription.
UpdateConsumerGroup	iot:UpdateConsumerGroup	*	Modifies the name of a consumer group.
QueryConsumerGroupByGroupId	iot:QueryConsumerGroupByGroupId	*	Queries the details of a consumer group by consumer group ID.
QueryConsumerGroupList	iot:QueryConsumerGroupList	*	Queries all consumer groups of an account or performs a fuzzy search by consumer group name.
QueryConsumerGroupStatus	iot:QueryConsumerGroupStatus	*	Queries the status of a consumer group when an AMQP server-side subscription is enabled. The status information includes the online client information, message consumption rate, number of accumulated messages, and the most recent message consumption time.
ResetConsumerGroupPosition	iot:ResetConsumerGroupPosition	*	Clears the accumulated messages of a consumer group when an AMQP server-side subscription is enabled.
DeleteConsumerGroup	iot:DeleteConsumerGroup	*	Deletes a consumer group.
CreateConsumerGroupSubscribeRelation	iot:CreateConsumerGroupSubscribeRelation	*	Adds a consumer group to an AMQP server-side subscription.
DeleteConsumerGroupSubscribeRelation	iot:DeleteConsumerGroupSubscribeRelation	*	Removes a consumer group from an AMQP subscription.
Configure an AMQP server-side subscription	iot:sub	*	Connects to IoT Platform by configuring an AMQP server-side subscription.

2.4. RAM user access

A RAM user can be used to access the resources of IoT Platform. This article describes how to create a RAM user, authorize a RAM user to access the resources of IoT Platform, and use a RAM user to log on to the IoT Platform console.

Context

To use a RAM user to access IoT Platform, you must create a RAM user and attach a policy that contains the access permission on IoT Platform to the RAM user. For more information about how to create custom policies, see Custom permissions.

Create a RAM user

If you already have a RAM user, skip the following steps:

1. Log on to the Resource Access Management (RAM) console.
2. In the left-side navigation pane, choose **Identities** > **Users**.
3. On the Users page, click **Create User**.
4. On the Create User page, set the **Logon Name** and **Display Name** parameters.
5. In the **Access Mode** section, select **Console access** or **Programmatic Access** and set the parameters.

   > **Note** To ensure the security of your Alibaba Cloud account, we recommend that you select only one access mode for the RAM user. This prevents the RAM user from using an AccessKey pair to access Alibaba Cloud resources after the RAM user is removed from the organization.

6. Click **OK**.
7. Authenticate your identity. Alibaba Cloud may authenticate your identity by using a verification code. The verification code is sent to the mobile number that is bound your Alibaba Cloud account. You must enter the verification code in the verification dialog box.

After you create a RAM user, you can use the RAM user to log on to the Alibaba Cloud official website and the IoT Platform console. The logon URL for RAM users is displayed on the **Overview** page of the **RAM console**.

A RAM user can access your Alibaba Cloud resources only after you authorize the RAM user. To enable a RAM user to access the resources of IoT Platform, you must grant the RAM user the access permission on IoT Platform.

Authorize a RAM user to access IoT Platform

In the **RAM** console, you can grant permissions to a single RAM user on the **Users** page. You can also grant the same permissions to all members in a RAM user group on the **Groups** page. The following example shows you how to grant permissions to a single RAM user.

1. Log on to the RAM console.
2. In the left-side navigation pane, choose **Identities** > **Users**.
3. Select the RAM user that you want to authorize and click **Add Permissions** below the list of RAM users.
4. In the Add Permissions panel, select the IoT Platform policies that you want to attach to the RAM user and click **OK**.

   > **Note** If you want to grant custom permissions to a RAM user, you must create a custom policy. For more information about how to create custom policies, see Custom permissions.

After the authorization is complete, the RAM user can access resources and perform operations as defined in the policies that are attached to the RAM user.

Log on to the console as a RAM user

If you use an Alibaba Cloud account, you can log on to the console from the Alibaba Cloud official website. If you are a RAM user, you must log on to the console from the **RAM Account Login** page.

1. Obtain the URL of the **RAM Account Login** page.

   Log on to the **RAM console** by using your Alibaba Cloud account. On the **Overview** page, copy the URL in the **Account Management** section. You can send the URL to RAM users.

2. Go to the **RAM Account Login** page to log on to the console as a RAM user.

   You can log on to the console as a RAM user by using a logon name in the following formats:

   - Logon name 1: `<$username>@<$AccountAlias>.onaliyun.com`. Example: `username@company-alias.onaliyun.com`.

     > **Note** The logon name of the RAM user is in the User Principal Name (UPN) format. All logon names that are listed in the RAM console follow this format. <$username> indicates the username of the RAM user. <$AccountAlias>.onaliyun.com indicates the default domain name.

   - Logon name 2: `<$username>@<$AccountAlias>`. Example: `username@company-alias`.

     > **Note** <$username> indicates the username of the RAM user. <$AccountAlias> indicates the account alias.

   - Logon name 3: `<$username>@<$DomainAlias>`. You can use this logon name if you have configured a domain alias.

     > **Note** <$username> indicates the username of the RAM user. <$DomainAlias> indicates the domain alias.

3. In the upper-right corner, click **Console** to go to the **Alibaba Cloud Management Console**.
4. Move the pointer over the upper-left corner in the console and choose **Products and Services** > **IoT Platform**. Then, you can go to the **IoT Platform console**.

After you log on to the **IoT Platform console** as a RAM user, you can use the RAM user to perform authorized operations in the console.

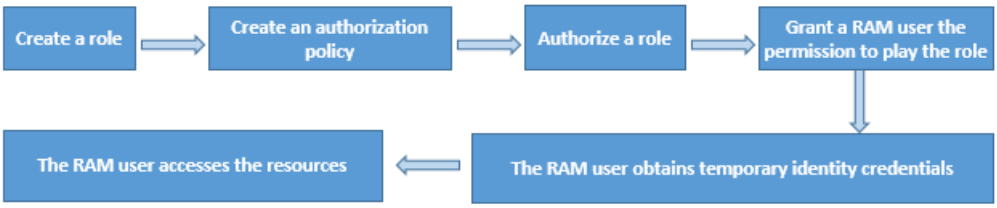
2.5. Manage permissions by using STS

Security Token Service (STS) enables more strict permission management than Resource Access Management (RAM). You can use STS to grant RAM users temporary permissions to access resources.

Context

RAM users and the permissions that are granted to RAM users are permanently valid. You can only manually delete RAM users or revoke permissions from RAM users. If the information of a RAM user is leaked and you do not delete the RAM user or revoke permissions from the RAM user, your Alibaba Cloud resources and information are exposed to risks. Therefore, we recommend that you use STS to manage key permissions or permissions that do not require long-term validity.

Process for granting temporary permissions to RAM users



Step 1: Create a RAM role

A RAM role is a virtual entity that represents a virtual user with a set of permissions.

1. Log on to the RAM console by using your Alibaba Cloud account.
2. Click **Roles**. On the Roles page, click **Create Role** to create a RAM role.
3. In the Create Role panel, select **Alibaba Cloud Account** as the trusted entity. Then, click **Next**.
4. Set the RAM Role Name and Note parameters, select **Current Alibaba Cloud Account** or **Other Alibaba Cloud Account** for the Select Trusted Alibaba Cloud Account parameter, and then click **OK**.

> **Note** If you select **Other Alibaba Cloud Account**, enter the ID of another Alibaba Cloud account.

Step 2: Create a policy

A policy defines the resource permissions that you want to grant to roles.

1. Log on to the RAM console. In the left-side navigation pane, choose **Permissions > Policies**.
2. On the Policies page, click **Create Policy**.
3. On the **Create Policy** page, click the **JSON** tab.
4. Specify the policy parameters and click **Next Step**.

   You can also click the **JSON** tab and write a policy script. For more information, see Policy structure and syntax.

   The following sample code shows a policy that has read-only permissions on the resources of IoT Platform:

```
{
    "Version":"1",
    "Statement":[
        {
            "Action":[
                "rds:DescribeDBInstances",
```

```
                "rds:DescribeDatabases",
                "rds:DescribeAccounts",
                "rds:DescribeDBInstanceNetInfo"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":"ram:ListRoles",
            "Effect":"Allow",
            "Resource":"*"
        },
        {
            "Action":[
                "mns:ListTopic"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":[
                "dhs:ListProject",
                "dhs:ListTopic",
                "dhs:GetTopic"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":[
                "ots:ListInstance",
                "ots:ListTable",
                "ots:DescribeTable"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":[
                "log:ListShards",
                "log:ListLogStores",
                "log:ListProject"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Effect":"Allow",
            "Action":[
                "iot:Query*",
                "iot:List*",
                "iot:Get*",
                "iot:BatchGet*"
            ],
```

```
            "Resource":"*"
        }
    ]
}
```

The following sample code shows a policy that has read and write permissions on the resources of IoT Platform:

```
{
    "Version":"1",
    "Statement":[
        {
            "Action":[
                "rds:DescribeDBInstances",
                "rds:DescribeDatabases",
                "rds:DescribeAccounts",
                "rds:DescribeDBInstanceNetInfo"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":"ram:ListRoles",
            "Effect":"Allow",
            "Resource":"*"
        },
        {
            "Action":[
                "mns:ListTopic"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":[
                "dhs:ListProject",
                "dhs:ListTopic",
                "dhs:GetTopic"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":[
                "ots:ListInstance",
                "ots:ListTable",
                "ots:DescribeTable"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Action":[
                "log:ListShards",
                "log:ListLogStores",
```

```
                "log:ListProject"
            ],
            "Resource":"*",
            "Effect":"Allow"
        },
        {
            "Effect":"Allow",
            "Action":"iot:*",
            "Resource":"*"
        }
    ]
}
```

5. Specify the **Name** and **Note** parameters, and then click **OK**.

After a policy is created, you can attach the policy to a RAM role. This way, the permissions that are defined in this policy are granted to the RAM role.

Step 3: Authorize a RAM role

A RAM role can access resources only after it is authorized. To authorize a single RAM role, you can click **Add Permissions** in the Actions column of the RAM role on the **Roles** page. To authorize multiple RAM roles at a time, perform the following steps:

1. In the RAM console, choose **Permissions > Grants** in the left-side navigation pane.
2. On the page that appears, click **Grant Permission**.
3. On the Grant Permission page, enter the names of RAM roles in the **Principal** field, select policies that you want to attach to the RAM roles, and then click **OK**.

After you authorize RAM roles, you can grant a RAM user the permission to assume RAM roles.

Step 4: Grant a RAM user the permission to assume a RAM role

After a policy is attached to a RAM role, the RAM role obtains the permissions that are defined in the policy. However, a RAM role is only a virtual user identity. The RAM role can be used to perform the allowed operations only after a RAM role is assumed by a RAM user. If any RAM user can assume a RAM role, security risks are caused. To prevent such risks, a RAM user can assume RAM roles only after the RAM user is authorized.

To authorize a RAM user to assume a RAM role, you can create a custom policy in which the Resource parameter is set to the ID of the RAM role. Then, you can use this policy to authorize the RAM user.

1. Log on to the RAM console. In the left-side navigation pane, choose **Permissions > Policies**.
2. On the Policies page, click **Create Policy**.
3. On the **Create Policy** page, click the **JSON** tab.
4. Specify the policy parameters and click **Next Step**.

> **Note** In the policy content, set the Resource parameter to the Alibaba Cloud Resource Name (ARN) of a RAM role. To view the ARN of a RAM role, go to the **Roles** page and click the name of the RAM role. You can view the ARN of the RAM role in the **Basic Information** section.

Example:

```
{
    "Version":"1",
    "Statement":[
        {
            "Effect":"Allow",
            "Action":"iot:QueryProduct",
            "Resource":"ARN of a RAM role"
        }
    ]
}
```

5. Specify the **Name** and **Note** parameters, and then click **OK**.
6. After the policy is created, return to the **RAM console** homepage.
7. In the left-side navigation pane, choose **Identities > Users**.
8. In the list of RAM users, select a RAM user that you want to authorize and click **Add Permissions** below the list of RAM users.
9. In the Add Permissions panel, select the created policy and click **OK**.

After the authorization is complete, the RAM user obtains the permission to assume the specified RAM role. Then, you can use STS to obtain the temporary identity credentials that are required to access resources.

Step 5: Obtain temporary identity credentials for a RAM user

Authorized RAM users can call the STS API operations or use STS SDKs to obtain the temporary identity credentials. The temporary identity credentials include an AccessKey ID, AccessKey secret, and security token. For more information about the STS API and STS SDKs, see STS API reference and STS SDK reference in the RAM documentation.

The following parameters are required when you use the STS API or SDK to obtain temporary identity credentials:

- RoleArn: the ARN of the RAM role that the RAM user is to assume.
- RoleSessionName: the name of the temporary identity credentials. This is a custom parameter.
- Policy: the policy that specifies the permissions of the RAM role to be granted to the RAM user. This parameter is used to generate a token with limited permissions of the RAM role. If you do not set this parameter, a token that has all permissions of the RAM role is returned.
- DurationSeconds: the validity period of the temporary identity credentials. This parameter is measured in seconds. The default value is 3600 and the value ranges from 900 to 3600.
- id and secret: the AccessKey ID and AccessKey secret of the RAM user.

The following examples show how to obtain temporary identity credentials.

API example: The RAM user calls the AssumeRole operation of STS to obtain the temporary identity credentials.

```
https://sts.aliyuncs.com?Action=AssumeRole
&RoleArn=acs:ram::1234567890123456:role/iotstsrole
&RoleSessionName=iotreadonlyrole
&DurationSeconds=3600
&Policy=<url_encoded_policy>
&<Common request parameters>
```

SDK example: The RAM user uses the Python command-line interface (CLI) for STS to obtain the temporary identity credentials.

```
$python ./sts.py AssumeRole RoleArn=acs:ram::1234567890123456:role/iotstsrole RoleSessionName=iotreadonlyrole Policy='{"Version":"1","Statement":[{"Effect":"Allow","Action":"iot:*","Resource":"*"}]}' DurationSeconds=3600 --id=id --secret=secret
```

After the request is successful, the temporary identity credentials are returned. The credentials include an AccessKey ID, AccessKey secret, and security token.

Step 6: Access resources as a RAM user

After a RAM user obtains the temporary identity credentials, the RAM user can pass in the credentials in SDK requests to assume the specified RAM role.

The following sample code shows that a RAM user uses STS SDK for Java to assume a RAM role. The RAM user passes in the AccessKey ID, AccessKey secret, and security token in the request and creates the IAcsClient object.

```
IClientProfile  profile = DefaultProfile.getProfile("cn-hangzhou", AccessKeyId,AccessSecret);
RpcAcsRequest request.putQueryParameter("SecurityToken", Token);
IAcsClient client = new DefaultAcsClient(profile);
AcsResponse response = client.getAcsResponse(request);
```

2.6. AliyunServiceRoleForIoTLogExport service linked role

This article describes the AliyunServiceRoleForIoTLogExport service linked role and how to delete the role.

Description

IoT Platform provides the log dump feature. This feature allows you to export the operations log of IoT Platform to a Logstore of Log Service. To implement the feature, you must obtain access to Log Service. When you enable the feature, Alibaba Cloud creates the AliyunServiceRoleForIoTLogExport service linked role. You can assign the role to IoT Platform.

Role name:

`AliyunServiceRoleForIoTLogExport`

Role policy:

`AliyunServiceRolePolicyForIoTLogExport`

Policy document:

```
{
  "Version": "1",
  "Statement": [
    {
      "Action": [
        log:PostLogStoreLogs
        "log:CreateProject",
        "log:GetLogStore",
        "log:CreateLogStore",
        "log:GetLogStore",
        "log:ListLogStores",
        "log:CreateLogStore",
        log:CreateConfig
        log:UpdateConfig
        "log:GetConfig",
        "log:CreateIndex",
        "log:GetIndex",
        "log:UpdateIndex",
        log:CreateSavedSearch
        log:UpdateSavedSearch
        "log:GetSavedSearch",
        log:DeleteSavedSearch
        "log:GetSavedSearch",
        "log:CreateDashboard",
        "log:UpdateDashboard"
        "log:GetDashboard",
        log:DeleteDashboard
        log:ListDashboard
      ],
      "Resource": "acs:log:*:*:project/*",
      "Effect": "Allow"
    },
    {
      "Action": "ram:DeleteServiceLinkedRole",
      "Resource": "*",
      "Effect": "Allow",
      "Condition": {
        "StringEquals": {
          "ram:ServiceName": "log-export.iot.aliyuncs.com"
        }
      }
    }
  ]
}
```

Delete the role

If you no longer use the AliyunServiceRoleForIoTLogExport role, delete the role.

1. Disable the log dump feature for all products step by step. For more information about how to disable the log dump feature for a product, see Disable the IoT Platform log dump feature.
2. Delete the role. For more information, see Delete a service-linked role.