

Alibaba Cloud IoT Platform

Developer Guide (Cloud)

Issue: 20180912

Legal disclaimer

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

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.
5. By law, all the content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade

secrets. No part of the Alibaba Cloud website, product programs, or content shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates).

6. Please contact Alibaba Cloud directly if you discover any errors in this document.

Generic conventions

Table -1: Style conventions

Style	Description	Example
	This warning information indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
	This warning information indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restore business.
	This indicates warning information, supplementary instructions, and other content that the user must understand.	 Note: Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	 Note: You can use Ctrl + A to select all files.
>	Multi-level menu cascade.	Settings > Network > Set network type
Bold	It is used for buttons, menus, page names, and other UI elements.	Click OK .
Courier font	It is used for commands.	Run the <code>cd /d C:/windows</code> command to enter the Windows system folder.
<i>Italics</i>	It is used for parameters and variables.	<code>bae log list --instanceid Instance_ID</code>
[] or [a b]	It indicates that it is a optional value, and only one item can be selected.	<code>ipconfig [-all -t]</code>
{ } or {a b}	It indicates that it is a required value, and only one item can be selected.	<code>swich {stand slave}</code>

Contents

Legal disclaimer.....	I
Generic conventions.....	I
1 SDK reference.....	1
1.1 Download SDKs.....	1
1.2 Java SDK.....	1
1.3 Use the Python SDK.....	2
1.4 PHP SDK.....	3
1.5 Use the .NET SDK.....	4
2 API reference.....	6
2.1 Overview.....	6
2.2 Call the IoT Platform API.....	6
2.3 Common parameters.....	7
2.4 Signature.....	10
2.5 Manage products.....	16
2.5.1 CreateProduct.....	16
2.5.2 QueryProduct.....	19
2.5.3 QueryProductList.....	22
2.5.4 UpdateProduct.....	26
2.6 Manage devices.....	27
2.6.1 RegisterDevice.....	27
2.6.2 QueryDeviceDetail.....	30
2.6.3 GetGatewayBySubDevice.....	33
2.6.4 QueryDevice.....	37
2.6.5 DeleteDevice.....	39
2.6.6 GetDeviceStatus.....	41
2.6.7 BatchGetDeviceState.....	43
2.6.8 DisableThing.....	45
2.6.9 EnableThing.....	47
2.6.10 BatchCheckDeviceNames.....	48
2.6.11 BatchRegisterDeviceWithApplyId.....	50
2.6.12 BatchRegisterDevice.....	52
2.6.13 QueryBatchRegisterDeviceStatus.....	54
2.6.14 QueryPageByApplyId.....	56
2.6.15 QueryDeviceEventData.....	58
2.6.16 QueryDevicePropertyData.....	62
2.6.17 QueryDeviceServiceData.....	65
2.6.18 InvokeThingService.....	68
2.6.19 QueryDevicePropertyStatus.....	71
2.6.20 SetDeviceProperty.....	73
2.6.21 QueryDeviceProp.....	76

2.6.22 SaveDeviceProp.....	77
2.6.23 DeleteDeviceProp.....	79
2.6.24 GetThingTopo.....	80
2.6.25 NotifyAddThingTopo.....	83
2.6.26 RemoveThingTopo.....	85
2.6.27 QueryDeviceStatistics.....	87
2.7 Manage Topics.....	88
2.7.1 CreateProductTopic.....	88
2.7.2 UpdateProductTopic.....	90
2.7.3 QueryProductTopic.....	91
2.7.4 DeleteProductTopic.....	93
2.7.5 CreateTopicRouteTable.....	94
2.7.6 DeleteTopicRouteTable.....	95
2.7.7 QueryTopicReverseRouteTable.....	96
2.7.8 QueryTopicRouteTable.....	97
2.8 Error codes.....	98
2.9 Rules engine.....	107
2.9.1 CreateRule.....	107
2.9.2 CreateRuleAction.....	109
2.9.3 DeleteRule.....	115
2.9.4 DeleteRuleAction.....	116
2.9.5 GetRule.....	117
2.9.6 GetRuleAction.....	119
2.9.7 ListRule.....	120
2.9.8 ListRuleActions.....	123
2.9.9 StartRule.....	124
2.9.10 StopRule.....	125
2.9.11 UpdateRule.....	126
2.10 Communications.....	128
2.10.1 Pub.....	128
2.10.2 PubBroadcast.....	130
2.10.3 RRpc.....	131
2.11 Device shadows.....	133
2.11.1 GetDeviceShadow.....	133
2.11.2 UpdateDeviceShadow.....	134

1 SDK reference

1.1 Download SDKs

IoT Platform SDKs support the following programming languages: Java, Python, PHP, and .NET.

Download an IoT SDK

Click one of the following links to download the corresponding SDK source code:

- [IoT SDK demo](#)

**Note:**

Platform API demo resources are available for Java, Python, PHP, and .NET.

- [IoT Java SDK](#)
- [IoT Python SDK](#)
- [IoT PHP SDK](#)
- [IoT .Net SDK](#)

1.2 Java SDK

The IoT Java SDK allows you to create Java applications to interact with IoT platform easily.

Install the IoT Java SDK

1. Install a Java development environment.

You can download a Java IDE from [Java official website](#) , and install the IDE by following the instructions.

2. Install the IoT Java SDK.

- a. Visit [Apache Maven official website](#) to download a Maven installation package.
- b. Add a Maven dependency.

Maven coordinates of the IoT Java SDK dependency

```
<!-- https://mvnrepository.com/artifact/com.aliyun/aliyun-java-sdk-iot -->
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-iot</artifactId>
  <version>5.0.0</version>
```

```
</dependency>
```

Public libraries

```
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>3.5.1</version>
</dependency>
```

Initialize the SDK



Note:

The following example uses the China (Shanghai) region and the corresponding endpoint. You must specify the region of your IoT Platform and the endpoint of your service to use the SDK.

```
String accessKey = "<your accessKey>";
String accessSecret = "<your accessSecret>";
DefaultProfile.addEndpoint("cn-shanghai", "cn-shanghai", "Iot", "iot.
cn-shanghai.aliyuncs.com");
IClientProfile profile = DefaultProfile.getProfile("cn-shanghai",
accessKey, accessSecret);
DefaultAcsClient client = new DefaultAcsClient(profile); //Initialize
the SDK
```

accessKeyId is the AccessKeyId of your account, and **accessSecret** is the AccessKeySecret paired with the AccessKeyId. You can go to [AccessKeys](#) page in the Alibaba Cloud console to create or view your AccessKey.

Initiate a call

The following example demonstrates how to call the **Pub** operation to publish messages to a topic.

```
PubRequest request = new PubRequest();
request.setProductKey("productKey");
request.setMessageContent(Base64.encodeBase64String("hello world".
getBytes()));
request.setTopicFullName("/productKey/deviceName/get");
request.setQos(0); //Currently, QoS0 and QoS1 are supported.
PubResponse response = client.getAcsResponse(request);
System.out.println(response.getSuccess());
System.out.println(response.getErrorMessage());
```

1.3 Use the Python SDK

Install the Python SDK

1. Install a Python IDE.

Download the Python installation package from [Official website of Python](#) and install Python.

2. Install pip. Skip this step if pip has already been installed.

Download the pip installation package from [Official website of pip](#) and install pip.

3. Install the Python SDK.

Run the following commands as an administrator to install the Python SDK.

```
sudo pip install aliyun-python-sdk-core
sudo pip install aliyun-python-sdk-iot
```

4. Run the following commands to import the related IoT Python SDK files to a Python file.

```
from aliyunsdkcore import client
from aliyunsdkiot.request.v20180120 import RegisterDeviceRequest
from aliyunsdkiot.request.v20180120 import PubRequest
...
```

Initialize the SDK

```
accessKeyId = '<your accessKey>'
accessKeySecret = '<your accessSecret>'
clt = client.AcsClient(accessKeyId, accessKeySecret, 'cn-shanghai')
```

accessKeyId is the AccessKeyId of your account, **and accessKeySecret** is the AccessKeySecret corresponding to the AccessKeyId. You can create and view your AccessKey on the **AccessKey Management** in the console.

Initiate a call

In this example, call the **Pub** API to publish messages to the device.

```
request = PubRequest.PubRequest()
request.set_accept_format('json') # Set the response format. The
default is XML.
request.set_ProductKey('productKey')
request.set_TopicFullName('/productKey/deviceName/get') #Full name of
the topic to which the messages are sent.
request.set_MessageContent('aGVsbG8gd29ybGQ=') #hello world Base64
String
request.set_Qos(0)
result = clt.do_action_with_exception(request)
print 'result : ' + result
```

1.4 PHP SDK

You can use the PHP SDK to manage IoT Platform.

Install the IoT PHP SDK

1. Install the PHP development environment.

Go to the [PHP website](#) to download the PHP installation package and complete the installation.

2. Download and then extract the IoT PHP SDK package.

Click [IoT PHP SDK Download Address](#) to download the PHP SDK package, and then extract the package to the specified directory. The PHP SDK is a software development kit that does not require installation.

Initialize the SDK

```
<? php
include_once 'aliyun-php-sdk-core/Config.php';
use \Iot\Request\V20180120 as Iot;
//Set your AccessKeyId, AccessSecret, and ProductKey.
$accessKeyId = "";
$accessSecret = "";
$iClientProfile = DefaultProfile::getProfile("cn-shanghai", $
accessKeyId, $accessSecret);
$client = new DefaultAcsClient($iClientProfile);
```

Initiate a call

The following example shows how to call the **Pub** operation to publish data to a device.

```
$request = new Iot\PubRequest();
$request->setProductKey("productKey");
$request->setMessageContent("aGVsbG93b3JsZA="); //hello world Base64
String.
$request->fig ("/product key/ergonomic ename/get "); // Full name of
the topic that the message is sent to.
$response = $client->getAcsResponse($request);
print_r($response);
```

1.5 Use the .NET SDK

Install the IoT .NET SDK

1. Install a .NET development environment.

Recommendations:

- Microsoft .NET Framework 4.0 and later.
- Visual Studio 2010 and later.

2. Import IoT .NET SDK core library from the Alibaba Cloud.

Download the package of the IoT .NET SDK core library from [.NET SDK releases](#), and extract a DLL file from the package.

3. Add a reference to the core library.

- a. In the **Solution Explorer** of Visual Studio, right-click your project, and select **Reference**.
- b. Click **Add Reference**.
- c. Click **Browse** in the dialog box.

- d. Select the DDL file *aliyun-net-sdk-Iot.dll*, and click **OK**.

Initialize the SDK

```
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Exceptions;
using Aliyun.Acs.Core.Profile;
IClientProfile clientProfile = DefaultProfile.GetProfile("<your-region-id>", "<your-access-key-id>", "<your-access-key-secret>");
DefaultAcsClient client = new DefaultAcsClient(clientProfile);
```

Initiate a call

In this example, call the **Pub** API to send messages to a topic.

```
PubRequest request = new PubRequest();
request.ProductKey = "<productKey>";
request.TopicFullName = "/" + request.ProductKey + "<deviceName>/get";
byte[] payload = Encoding.Default.GetBytes("Hello World.");
String payloadStr = Convert.ToBase64String(payload);
request.MessageContent = payloadStr;
request.Qos = 0;
try
{
    PubResponse response = client.GetAcsResponse(request);
    Console.WriteLine("publish message result: " + response.Success);
    Console.WriteLine(response.ErrorMessage);
}
catch (ServerException e)
{
    Console.WriteLine(e.ErrorCode);
    Console.WriteLine(e.ErrorMessage);
}
catch (ClientException e)
{
    Console.WriteLine(e.ErrorCode);
    Console.WriteLine(e.ErrorMessage);
}
```

2 API reference

2.1 Overview

If you want to call the API provided by IoT Platform, you can send an HTTP or HTTPS GET or POST request to the API endpoint. Specify the required request parameters in your request. The system returns a response after handling the request.

The following sections describe how to call the API.

- [Call the IoT Platform API](#)
- [Common parameters](#)

To ensure the security of your Alibaba Cloud account, we recommend that you call the API as a RAM user.

For more information about how to grant RAM users access to the API, see [API permissions](#).

2.2 Call the IoT Platform API

Request structure

You can call the API provided by IoT Platform by sending HTTP or HTTPS requests.

The request structure is as follows:

```
http://Endpoint/?Action=xx&Parameters
```

Specifically:

- **Endpoint**: address where you can access IoT Platform. The following addresses are available:
 - China (Shanghai): `iot.cn-shanghai.aliyuncs.com`
 - Singapore: `iot.ap-southeast-1.aliyuncs.com`
 - US (Silicon Valley): `iot.us-west-1.aliyuncs.com`
 - Japan (Tokyo): `iot.ap-northeast-1.aliyuncs.com`
 - Germany (Frankfurt): `iot.eu-central-1.aliyuncs.com`
- **Action**: operation to be performed. For example, when you can call **CreateProduct**, the operation creates a product.
- **Parameters**: request parameters. Separate each parameter with an ampersand (&).

Request parameters are comprised of common request parameters and parameters customized for the API. Common parameters include the API version number, authentication information, and other information.

The following example describes how to call **Pub** to publish a message to a specified topic.

**Note:**

To assist reading, examples provided in this document are formatted.

```
https://iot.cn-shanghai.aliyuncs.com/?Action=CreateProduct
? Format=XML
&Version=2017-04-20
&Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dgI%3D
&SignatureMethod=HMAC-SHA1
&SignatureNonce=15215528852396
&SignatureVersion=1.0
&AccessKeyId=...
&Timestamp=2017-07-19T12:00:00Z
&RegionId=cn-shanghai
...
```

Authorization

To ensure the security of your Alibaba Cloud account, we recommend that you call the API as a RAM user. Before calling the API as a RAM user, you need to create and attach specific authorization policies to the RAM user to allow access to the API.

For more information about API permissions that can be granted to RAM users, see [API permissions](#).

2.3 Common parameters

This section lists the common request parameters and response parameters of the IoT Platform API.

Common request parameters

Common request parameters are the request parameters required every time you call the API.

Parameter	Type	Required	Description
Format	String	No	Format of the response data. JSON and XML are supported. The default value is XML.
Version	String	Yes	The API version number. The parameter value needs to be in the YYYY-MM-DD format. The latest version is 2018-01-20. Each API operation may have multiple versions.

Parameter	Type	Required	Description
AccessKeyId	String	Yes	The access key ID issued by Alibaba Cloud for users to access services.
Signature	String	Yes	Signature string.
SignatureMethod	String	Yes	The signature method. Currently, HMAC-SHA1 is supported.
Timestamp	String	Yes	Required timestamp Date is in ISO8601 format, and must use UTC time. The parameter value needs to be in the YYYY-MM-DDThh:mm:ssZ format. For example, you can set the parameter to 2016-01-04T12:00:00Z, which indicates January 4, 2016 20:00:00 Beijing Time (UTC +8).
SignatureVersion	String	Yes	The signature algorithm version. The current version is 1.0.
SignatureNonce	String	Yes	The unique random number. It is used to prevent replay attacks. You need to use different random numbers in different requests.
RegionId	String	Yes	The region where your devices are located. The region you specify must match the region specified in the console, for example, cn-shanghai.

Example

```
https://iot.cn-shanghai.aliyuncs.com/  
? Format=XML  
&Version=2018-01-20  
&Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dgI%3D  
&SignatureMethod=HMAC-SHA1  
&SignatureNonce=15215528852396  
&SignatureVersion=1.0  
&AccessKeyId=...  
&Timestamp=2018-05-20T12:00:00Z  
&RegionId=cn-shanghai
```

Common response parameters

The response parameters are displayed in specific formats. If HTTP status code 2xx is returned, the API operation call was successful. If HTTP status code 4xx or 5xx is returned, the API operation call has failed. The response data can be in either XML or JSON. You can specify the

format of response data when sending your request. By default, response data is in the XML format.

Regardless of whether your API operation call is successful or not, the system returns

RequestId as a unique identifier.

- **Formats of data returned by a successful call**

- JSON format

```
{
  "RequestId": "4C467B38-3910-447D-87BC-AC049166F216"
  /* Response data */
}
```

- XML format

```
<? xml version="1.0" encoding="UTF-8"? >
<!--Root node of the response data-->
<API operation name+Response>
  <!--Request tag-->
  <RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId>
  <!--Response data-->
</API operation name+Response>
```

- **Formats of data returned by a failed call**

If any error occurs when you call an API operation, no data is returned. You can use the error code to locate the cause.

If an error occurs, an HTTP status code 4xx or 5xx is returned. The returned message body contains the specific error code and error message. It also contains a globally unique request ID (RequestId). If you cannot confirm what error occurs, contact Alibaba Cloud customer service and provide your request ID for assistance.

- JSON format

```
{
  "RequestId": "8906582E-6722-409A-A6C4-0E7863B733A5",
  "Code": "UnsupportedOperation",
  "Message": "The specified action is not supported."
}
```

- XML format

```
<? xml version="1.0" encoding="UTF-8"? >
<Error>
  <RequestId>8906582E-6722-409A-A6C4-0E7863B733A5</RequestId>
  <Code>UnsupportedOperation</Code>
  <Message>The specified action is not supported.</Message>
```

```
</Error>
```

2.4 Signature

IoT Platform performs sender identity authentication for every access request. Therefore, you must include the signature information in requests, irrespective of the use of HTTP or HTTPS protocol.

Signing method

When you are signing the parameters, you can view the AccessKeyId and the AccessKeySecret for your Alibaba Cloud account on the [AccessKey](#) page in the console, and then encrypt them in symmetry. An AccessKeyId is used to identify the visitor, and an AccessKeySecret is the secret key used to encrypt the signature string and verify the signature string on the server. You must keep your AccessKey confidential to minimize the risk of data breaches.



Note:

We provide SDKs in multiple programming languages, such as Java, Python, and PHP. If you use these SDKs, you can skip the authentication process. See [Download SDKs](#).

Sign the request parameters as follows:

1. Create a canonicalized query string.

- a. Sort the request parameters in a lexicographical order.

Sort the request parameters, including the [public request parameters](#) (except for **Signature** parameter) and the special parameters of an interface in a lexicographical order.



Note:

When you make a request by using the GET method, these parameters constitute the part of the URL following ? and are connected by &).

- b. Encode the parameter names and values.

Encode the request parameters by using the UTF-8 character set and according to the [RFC3986](#) rules. The encoding rules are as follows:

- Do not encode the letters A-Z and a-z , digits 0-9, and characters including hyphens -, underscores _, periods ., and tildes ~ .

- Encode other characters in the %XY format, with XY representing the hexadecimal ASCII value of the characters. For example, the half-width double quotes " are encoded as %22.
- Encode the extended UTF-8 characters in %XY%ZA... format.
- Encode whitespaces to %20 instead of the plus signs +.

Alternatively, you can use libraries that support the application/x-www-form-urlencoded MIME-type URL encoding.

For example, you use `java.net.URLEncoder` of Java language. Firstly, use `percentEncode` to encode a string, and then replace plus signs + in the encoded string with %20, asterisks * with %2A, and %7E with tildes ~. See the following example.

```
private static final String ENCODING = "UTF-8";
private static String percentEncode(String value) throws
UnsupportedEncodingException {
    return value != null ? URLEncoder.encode(value, ENCODING).replace(
        "+", "%20").replace("*", "%2A").replace("%7E", "~") : null;
}
```

- c. Connect the encoded parameter names and the values with equal signs =.
- d. Connect all the request parameters with ampersands &. The order of the parameters must be the same as that in step a.

You now have a canonicalized query string.

2. Construct the string to be signed.

You can use `percentEncode` to process the canonicalized query string that is created in step

1. See the following rules:

```
StringToSign=
    HTTPMethod + "&" + //HTTPMethod: HTTP method used for making
    request, for example GET.
    percentEncode("/") + "&" + //percentEncode("/"): Encode backslash
    (/) to %2F.
    percentEncode(CanonicalizedQueryString) //Encode the canonicalized
    query string that is created in the step 1.
```

3. Calculate the HMAC value.

Calculate the HMAC-SHA1 value of the `StringToSign` created in step 2, according to the [RFC2104](#) rules. The Java Base64 encoding method is used in the following example.

```
Signature = Base64( HMAC-SHA1( AccessSecret, UTF-8-Encoding-Of(
StringToSign) ) )
```



Note:

When calculating for the signature value, you use the **AccessKeySecret** of your account and an ampersand & (the hexadecimal ASCII value of an ampersand is 38) as the key. The hash algorithm used is SHA1.

4. Calculate the signature value.

Encode the preceding HMAC value into a string based on Base64 encoding rules to obtain the signature value.

5. Add the signature in the request.

Encode the value of **Signature** according to the [RFC3986](#) rules, and add the encoded signature value in the request.

Example 1: Parameter concatenation method

Take calling the API Pub as an example, and suppose your AccessKey information is `AccessKeyId=testid` and `AccessKeySecret=testsecret`.

1. Create a canonicalized query string.

```
http://iot.cn-shanghai.aliyuncs.com/?MessageContent=aGVsbG93b3JsZA%3D&Action=Pub&Timestamp=2017-10-02T09%3A39%3A41Z&SignatureVersion=1.0&ServiceCode=iot&Format=XML&Qos=0&SignatureNonce=0715a395-aedf-4a41-bab7-746b43d38d88&Version=2017-04-20&AccessKeyId=testid&SignatureMethod=HMAC-SHA1&RegionId=cn-shanghai&ProductKey=12345abcdeZ&TopicFullName=%2FproductKey%2Ftestdevice%2Fget
```

2. Create a `StringToSign` based on the canonicalized query string.

```
GET&%2F&AccessKeyId%3Dtestid%26Action%3DPub%26Format%3DXML%26MessageContent%3DaGVsbG93b3JsZA%253D%26ProductKey%3D12345abcdeZ%26Qos%3D0%26RegionId%3Dcn-shanghai%26ServiceCode%3Diot%26SignatureMethod%3DHMAC-SHA1%26SignatureNonce%3D0715a395-aedf-4a41-bab7-746b43d38d88%26SignatureVersion%3D1.0%26Timestamp%3D2017-10-02T09%253A39%253A41Z%26TopicFullName%3D%252FproductKey%252Ftestdevice%252Fget%26Version%3D2017-04-20
```

3. Calculate the HMAC-SHA1 value.

Because you have the `AccessKeySecret=testsecret`, the key used in RFC2104 calculation is `testsecret&`. The HMAC-SHA1 value is:

```
Y9eWn4nF8QPh3c4zAFkM/k/u7eA=
```

4. Add this value to the request URL as the **Signature** parameter value to produce the following request URL:

```
http://iot.cn-shanghai.aliyuncs.com/?MessageContent=aGVsbG93b3JsZA%3D&Action=Pub&Timestamp=2017-10-02T09%3A39%3A41Z&SignatureVersion=1.0&ServiceCode=iot&Format=XML&Qos=0&SignatureNonce=0715a395-aedf-4a41-bab7-746b43d38d88&Version=2017-04-20&AccessKeyId=testid&Signature=Y9eWn4nF8QPh3c4zAFkM%2Fk%2Fu7eA%3D&SignatureMethod=HMAC-SHA1&RegionId=cn-shanghai&ProductKey=12345abcdeZ&TopicFullName=%2FproductKey%2Ftestdevice%2Fget
```

Example 2: Java language coding

The codes in this example do not depend on any third-party library package and can be directly used. The signing procedure is as follows:

1. Create a canonicalized query string (including sorting the parameters, and URL encoding the parameter names and values).

```
public static Map<String, String> splitQueryString(String url)
    throws URISyntaxException, UnsupportedEncodingException {
    URI uri = new URI(url);
    String query = uri.getQuery();
    final String[] pairs = query.split("&");
    TreeMap<String, String> queryMap = new TreeMap<String, String>
>();
    for (String pair : pairs) {
        final int idx = pair.indexOf("=");
        final String key = idx > 0 ? pair.substring(0, idx) : pair;
        if (! queryMap.containsKey(key)) {
            queryMap.put(key, URLDecoder.decode(pair.substring(idx
+ 1),
                                CHARSET_UTF8));
        }
    }
    return queryMap;
}

/** Perform URL encoding on the parameter names and values*/
public static String generate(String method, Map<String, String>
parameter,
                                String accessKeySecret) throws
Exception {
    String signString = generateSignString(method, parameter);
    System.out.println("signString---" + signString);
    byte[] signBytes = hmacSHA1Signature(accessKeySecret + "&",
signString);
    String signature = newStringByBase64(signBytes);
    System.out.println("signature---" + signature);
    if ("POST".equals(method))
        return signature;
}
```

```
        return URLEncoder.encode(signature, "UTF-8");
    }
}
```

2. Create the StringToSign.

```
public static String generateSignString(String httpMethod,
                                      Map<String, String>
parameter) throws IOException {
    TreeMap<String, String> sortParameter = new TreeMap<String,
String>();
    sortParameter.putAll(parameter);
    String canonicalizedQueryString = UrlUtil.generateQueryString(
        sortParameter, true);
    if (null == httpMethod) {
        throw new RuntimeException("httpMethod can not be empty");
    }
    /** Create the StringToSign */
    StringBuilder stringToSign = new StringBuilder();
    stringToSign.append(httpMethod).append(SEPARATOR);
    stringToSign.append(percentEncode("/")).append(SEPARATOR);
    stringToSign.append(percentEncode(canonicalizedQueryString));
    return stringToSign.toString();
}
```

3. Calculate for the HMAC value of the StringToSign.

```
public static byte[] hmacSHA1Signature(String secret, String
baseString)
    throws Exception {
    if (isEmpty(secret)) {
        throw new IOException("secret can not be empty");
    }
    if (isEmpty(baseString)) {
        return null;
    }
    Mac mac = Mac.getInstance("HmacSHA1");
    SecretKeySpec keySpec = new SecretKeySpec(
        secret.getBytes(CHARSET_UTF8), ALGORITHM);
    mac.init(keySpec);
    return mac.doFinal(baseString.getBytes(CHARSET_UTF8));
}
private static boolean isEmpty(String str) {
    return (str == null || str.length() == 0);
}
```

4. Encode the HMAC value into a string according to the Base64 encoding rules, and then you obtain the signature value.

```
public static String newStringByBase64(byte[] bytes)
    throws UnsupportedEncodingException {
    if (bytes == null || bytes.length == 0) {
        return null;
    }
    return new String(new BASE64Encoder().encode(bytes));
}
public static String composeStringToSign(Map<String, String>
queries) {
    String[] sortedKeys = (String[]) queries.keySet()
        .toArray(new String[0]);
    Arrays.sort(sortedKeys);
}
```

```

        StringBuilder canonicalizedQueryString = new StringBuilder();
        for (String key : sortedKeys) { canonicalizedQueryString.append
("&").append(percentEncode(key))
            .append("=")
            .append(percentEncode((String) queries.get(key)));
        }
        StringBuilder stringToSign = new StringBuilder();
        stringToSign.append("GET");
        stringToSign.append("&");
        stringToSign.append(percentEncode("/"));
        stringToSign.append("&"); stringToSign.append(percentEncode(
canonicalizedQueryString.toString()
            .substring(1)));
        return stringToSign.toString();
    }
    public static String percentEncode(String value) {
        try {
            return value == null ? null : URLEncoder
                .encode(value, CHARSET_UTF8).replace("+", "%20")
                .replace("*", "%2A").replace("%7E", "~");
        } catch (Exception e) {
        }
        return "";
    }
}
/**
 * get SignatureNonce
 ** */
public static String getUniqueNonce() {
    UUID uuid = UUID.randomUUID();
    return uuid.toString();
}
/**
 * get timestamp
 ** */
public static String getISO8601Time() {
    Date nowDate = new Date();
    SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss'Z'");
    df.setTimeZone(new SimpleTimeZone(0, "GMT"));
    return df.format(nowDate);
}
}

```

5. Add the signature in the request.

```

public static String composeUrl(String endpoint, Map<String, String>
    > queries)
    throws UnsupportedOperationException {
    Map<String, String> mapQueries = queries;
    StringBuilder urlBuilder = new StringBuilder("");
    urlBuilder.append("http");
    urlBuilder.append("://").append(endpoint);
    if (-1 == urlBuilder.indexOf("?")) {
        urlBuilder.append("/?") ;
    }
    urlBuilder.append(concatQueryString(mapQueries));
    return urlBuilder.toString();
}
public static String concatQueryString(Map<String, String>
parameters)
    throws UnsupportedOperationException {
    if (null == parameters) {
        return null;
    }
}

```

```

    }
    StringBuilder urlBuilder = new StringBuilder("");
    for (Map.Entry<String, String> entry : parameters.entrySet()) {
        String key = (String) entry.getKey();
        String val = (String) entry.getValue();
        urlBuilder.append(encode(key));
        if (val != null) {
            urlBuilder.append("=").append(encode(val));
        }
        urlBuilder.append("&");
    }
    int strIndex = urlBuilder.length();
    if (parameters.size() > 0) {
        urlBuilder.deleteCharAt(strIndex - 1);
    }
    return urlBuilder.toString();
}
public static String encode(String value)
    throws UnsupportedOperationException {
    return URLEncoder.encode(value, "UTF-8");
}
}

```

2.5 Manage products

2.5.1 CreateProduct

You can call this operation to create a product.

Restrictions

This operation can only be used to create Basic Edition products. You create Pro Edition products in the IoT Platform console.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to CreateProduct .
Name	String	Yes	The name of the product that you want to create. The product name can be 4 to 30 characters in length and can contain letters, numbers, and underscores (_).  Note: The product name is unique under the current account.
NodeType	Integer	No	The node type of the product. Values include:

Name	Type	Required	Description
			0: Device. A device node cannot contain sub-devices. It can connect to IoT Hub either directly or as a sub-device of a gateway. 1: Gateway. A gateway can contain sub-devices. It can manage sub-devices, maintain the topological relationship of sub-devices, and synchronize the topological relationship to the cloud. The system uses 0 by default.
CatId	Long	No	Category information.  Note: The parameter is deprecated. You can set this field to any long integer, such as 10000.
Desc	String	No	The description of the product that you want to create. The description can be a maximum of 100 characters in length.
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
ProductInfo	ProductInfo	The information of the created product that is returned when the call is successful. For more information, see Table 2-1: ProductInfo .

Table 2-1: ProductInfo

Name	Type	Description
ProductName	String	The name of the created product.

Name	Type	Description
ProductKey	String	The GUID issued by IoT Platform when you create a new product.  Note: Secure ProductKey of the created product. You need to use ProductKey in other operations.
ProductDesc	String	The description of the product.
CreateUserId	Long	The identifier of the user that created the product.
CatId	Long	Category information.  Note: This parameter is deprecated.
FromSource	String	The source. Value: iotHub.
NodeType	Integer	The product node type. Values include: 0: Indicates a device node. A device node cannot contain sub-devices. It can connect to IoT Hub either directly or as a sub-device of a gateway. 1: Indicates a gateway node. A gateway can contain sub-devices. It can manage sub-devices, maintain the topological relationship of sub-devices, and synchronize the topological relationship to the cloud.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=CreateProduct
&Name=TestProduct
&CatId=10000
&Desc=Create Product test
&<common request parameters>
```

Response example

- JSON format

```
{
  RequestId:8AE93DAB-958F-49BD-BE45-41353C6DEBCE,
  Success:true,
  ProductInfo:{
    ProductKey:al*****,
    CatId:10000,
```

```

NodeType:0,
  ProductName:TestProduct,
  ProductDesc:Create Product test
}
}

```

- XML format

```

<? xml version="1.0" encoding="UTF-8"? >
  <CreateProductResponse>
    <RequestId>8AE93DAB-958F-49BD-BE45-41353C6DEBCE</RequestId>
    <ProductInfo>
      <ProductKey>a1*****</ProductKey>
      <ProductDesc>Create Product test</ProductDesc>
      <CatId>10000</CatId>
      <ProductName>TestProductx</ProductName>
    </ProductInfo>
    <NodeType>0</NodeType>
    <Success>true</Success>
  </CreateProductResponse>

```

2.5.2 QueryProduct

Use this operation to query the details of a specified product.

Limits

You cannot call this operation more than 50 times per second.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryProduct.
ProductKey	String	Yes	The ProductKey of the product. ProductKey is a product key issued by IoT Platform to a new product. It is a globally unique identifier (GUID) associated with the product. You can view this information in the returned results of the product you have created.
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.

Name	Type	Description
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Code	String	The error code returned when the call fails. For more information about error codes, see Error codes .
Data	Data	The data returned when the call is successful. See the following table Data.

Table 2-2: Data

Name	Type	Description
GmtCreate	Long	The time when the product was created.
DataFormat	Integer	The type of data communication protocol between the device and the cloud. Values: 0: Passthrough mode. Use a custom serial port data format. In this mode, the device can report raw data such as binary data streams. The Alibaba Cloud IoT Platform runs the data parsing script that you configured in the cloud to convert the raw data into Alink JSON-formatted data. 1: Alink JSON. The data exchange protocol between devices and the cloud. It is defined by the Alibaba Cloud IoT Platform using the JSON format.  Note: This parameter is only available for Pro Edition products.
Description	String	The product description.
DeviceCount	Integer	The number of devices of this product.
NodeType	Integer	The product node type. Values include: 0: Indicates a device node. A device cannot contain sub-devices. It can connect to IoT Hub either directly or as a sub-device of a gateway. 1: Indicates a gateway node. A gateway can contain sub-devices. It can manage sub-devices, maintain the topology of sub-devices, and synchronize the topology to the cloud.

Name	Type	Description
ProductKey	String	The product ID. It is a globally unique identifier (GUID) associated with the product.
ProductName	String	The product name.
ProductSecret	String	The product key.
CategoryName	String	The device type of a Pro Edition product. The value is the device type which you specified when you were creating the product.
CategoryKey	String	The identifier of the device type of a Pro Edition product.
AliyunCommodityCode	String	Values include: • <code>iothub</code>: IoT Platform Basic. • <code>iothub_senior</code>: IoT Platform Pro.
Id2	Boolean	Indicates whether the product is an ID2 product (currently , ID2 product is not released).

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryProduct
&ProductKey=a1*****
& <common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "DataFormat": 1,
    "ProductKey": "a1*****",
    "NodeType": 0,
    "ProductName": "iot_test",
    "ProductSecret": "4alWjhgiTG****",
    "CategoryName": "",
    "CategoryKey": "",
    "DeviceCount": 7221,
    "GmtCreate": 1516534408000,
    "AliyunCommodityCode": "iothub_senior",
    "Description": "Test"
  }
}
```

```
}

```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
<QueryProductResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <DataFormat>1</DataFormat>
    <ProductKey>a1*****</ProductKey>
    <NodeType>0</NodeType>
    <ProductName>iot_test</ProductKey>
    <ProductSecret>4alWjhgiTGU*****</ProductSecret>
    <CategoryName></CategoryName>
    <CategoryKey></CategoryKey>
    <DeviceCount>7221</DeviceCount>
    <GmtCreate>1516534408000</GmtCreate>
    <AliyunCommodityCode>iothub_senior</AliyunCommodityCode>
    <Description>Test</Description>
  </Data>
</QueryProductResponse>
```

2.5.3 QueryProductList

Use this operation to view the list of all products.

Restrictions

Use this operation to view the list of all IoT products under the current account.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryProductList .
CurrentPage	Integer	Yes	Specifies the page of returned results to be displayed.
PageSize	Integer	Yes	The number of results per page. The maximum is 200.
AliyunCommodityCode	String	No	The product type. Values include: iothub_senior: IoT Platform Pro iothub: IoT Platform Basic  Note: The system returns a list of all products if you have not specified this parameter.

Name	Type	Required	Description
Common Request Parameters	Common Request Parameters	Yes	See Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	Data	The data returned when the call is successful. For more information, see the following table.

Table 2-3: Data

Name	Type	Description
PageCount	Integer	The total number of pages.
PageSize	Integer	The number of records displayed per page.
CurrentPage	Integer	The current page number.
Total	Integer	The total number of records.
List	List	The product list. For product details, see: ProductInfo .

Table 2-4: ProductInfo

Name	Type	Description
DataFormat	Integer	The type of data communication protocol between the device and the cloud. Use the value: 0: Passthrough mode. Use a custom serial port data format. In this mode, the device can report raw data such as binary data streams. The Alibaba Cloud IoT Platform runs the data parsing script that you configured in the cloud to change the raw data into Alink JSON-formatted data.

Name	Type	Description
		1: Alink JSON. The data exchange protocol between devices and the cloud. It is defined by the Alibaba Cloud IoT Platform using the JSON format.  Note: This parameter is only available for Pro Edition products.
ProductKey	String	The product key. The GUID issued by the IoT platform when you create a new product.
NodeType	Integer	The product node type. Values include: 0: Indicates a device node. A device node cannot contain sub-devices. It can connect to IoT Hub either directly or as a sub-device of a gateway. 1: Indicates a gateway node. A gateway can contain sub-devices. It can manage sub-devices, maintain the topology of sub-devices, and synchronize the topology to the cloud.
ProductName	String	The product name.
DeviceCount	Integer	The number of devices under the product.
GmtCreate	Long	The time when the product was created.
Description	String	The product description.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryProductList
&CurrentPage=1
&PageSize=10
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "PageCount": 4,
    "PageSize": 20,
    "List": {
      "ProductInfo": [
```



```

    {
      "DataFormat": 0,
      "ProductKey": "a1*****",
      "NodeType": 0,
      "ProductName": "test123123123",
      "DeviceCount": 4,
      "GmtCreate": 1517479077000,
      "Description": "Description"
    },
    {
      "DataFormat": 1,
      "ProductKey": "a1*****",
      "NodeType": 0,
      "ProductName": "Test Event",
      "DeviceCount": 0,
      "GmtCreate": 1517466635000,
      "Description": "Description"
    }
  ],
  "CurrentPage": 1,
  "Total": 79
}

```

- XML format

```

<? xml version="1.0" encoding="UTF-8"? >
<QueryProductListResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <PageCount>4</PageCount>
    <PageSize>20</PageSize>
    <List>
      <ProductInfo>
        <DataFormat>0</DataFormat>
        <ProductKey>a1*****</ProductKey>
        <NodeType>0</NodeType>
        <ProductName>test123123123</ProductName>
        <DeviceCount>4</DeviceCount>
        <GmtCreate>1517479077000</GmtCreate>
        <Description>Description</Description>
      </ProductInfo>
      <ProductInfo>
        <DataFormat>1</DataFormat>
        <ProductKey>a1*****</ProductKey>
        <NodeType>0</NodeType>
        <ProductName>Test Event</ProductName>
        <DeviceCount>0</DeviceCount>
        <GmtCreate>1517466635000</GmtCreate>
        <Description>Description</Description>
      </ProductInfo>
    </List>
    <CurrentPage>1</CurrentPage>
    <Total>79</Total>
  </Data>
</QueryProductListResponse>

```

```
</QueryProductListResponse>
```

2.5.4 UpdateProduct

You can call this operation to modify the information for the specified product.

Limits

Calls to the operation are restricted to no more than 50 requests/second.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: UpdateProduct
ProductKey	String	Yes	The identifier of the product that you want to modify.
CatId	Long	No	Product category.  Note: This is a legacy field from the previous version. It is not used in the current version, You can input any long integer value, such as 10000.
ProductName	String	No	The new product name that you want to use. The product name must meet the following requirements: consists of 4-30 characters, including Chinese characters (Each Chinese character counts as two characters), English letters, numbers, and underscores (_).  Note: All product names under the account must be unique.
ProductDescription	String	No	The new product description that you want to use. The description can be up to 100 characters.
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
http://iot.cn-shanghai.aliyuncs.com/?Action=UpdateProduct
&ProductKey=al*****
&ProductName=TestProductNew
& <common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "C4FDA54C-4201-487F-92E9-022F42387458",
  "Success": true,
}
```

- XML format

```
<? xml version="1.0" encoding="UTF-8"? >
<UpdateProductResponse>
  <RequestId>C4FDA54C-4201-487F-92E9-022F42387458</RequestId>
  <Success>true</Success>
</UpdateProductResponse>
```

2.6 Manage devices

2.6.1 RegisterDevice

You can call this operation to register a device under the specified product.

Descriptions:

Registering a device creates a device under the product. After a device is successfully registered under the specified product, Alibaba Cloud IoT Platform issues a GUID (**IoTId**) for the device.

When you operate a device, you need to provide its unique `IoTId`.

**Note:**

Except for the **IoTId**, you can also identify a device using the combination of the **ProductKey** and **DeviceName**. The **ProductKey** is the identifier that IoT Platform issues for a product when it is created, and the **DeviceName** is the device name that either you specified or the system randomly generated when the device was registered. The priority of **IoTId** is higher than the combination of the **ProductKey** and **DeviceName**.

- If you want to register multiple devices under a product at a time and have the device names randomly generated, you can call the **BatchRegisterDevice** operation. See [BatchRegisterDevice](#)
- If you want to register multiple devices under a product at a time and want to specify the name for each device, you need to call **BatchCheckDeviceNames** and **BatchRegisterDeviceWithApplyId** operations in order. See [BatchCheckDeviceNames](#) and [BatchRegisterDeviceWithApplyId](#).

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: RegisterDevice.
ProductKey	String	Yes	The identifier of the product for which you want to register devices.
DeviceName	String	No	The name of the device that you want to register. The device name should contain 4-32 characters, including English letters, numbers, and special characters, for example , hyphens (-), underscores (_), at signs (@), periods (.) , and colons (:). The DeviceName is typically used in conjunction with the ProductKey to identify a specified device. Note: If the specified parameter is not set, the device names will be randomly generated.
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	Data	The device information returned when the call is successful. For more information, see DeviceInfo .

Table 2-5: DeviceInfo

Name	Type	Description
ProductKey	String	The identifier of the product that the device belongs to.
DeviceName	String	Device name.  Note: Save it in a secure location for future reference.
DeviceSecret	String	The private key of a device.  Note: Save it in a secure location for future reference.
IotId	String	The device ID issued by IoT Platform for the device is a unique identifier.  Note: Save it in a secure location for future reference.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=RegisterDevice
&ProductKey=al*****
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "DeviceName": "CqXL5h5ysRTA4NxjABjj",
    "ProductKey": "a1*****",
    "DeviceSecret": "tXHf4ezGEHcwdyMwoCDHGBmk9avi****",
    "IotId": "CqXL5h5ysRTA4NxjABjj0010fa****"
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<RegisterDeviceResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <DeviceName>CqXL5h5ysRTA4NxjABjj</DeviceName>
    <ProductKey>a1rYuVFha9S</ProductKey>
    <DeviceSecret>tXHf4ezGEHcwdyMwoCDHGBmk9avimccx</DeviceSecret>
  >
    <IotId>CqXL5h5ysRTA4NxjABjj0010fa8f00</IotId>
  </Data>
</RegisterDeviceResponse>
```

2.6.2 QueryDeviceDetail

You can call this operation to search the details of a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryDeviceDetail.
IotId	string	No	The device ID.  Note: If you use this parameter, ProductKey and DeviceName are not required. IotId is a globally unique identifier (GUID), and corresponds to a combination of ProductKey and DeviceName. If you specify both IotId and the combination of ProductKey and DeviceName, the system follows IotId.
ProductKey	String	No	The product key of the device.

Name	Type	Required	Description
			 Note: If you use this parameter, DeviceName is required.
DeviceName	String	No	The name of the device that you want to search for.  Note: If you use this parameter, ProductKey is required.
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Code	String	The error code returned when the call fails. For error codes, see Error codes .
Data	Data	The device details returned when the call is successful. See the following table Data.

Table 2-6: Data

Name	Type	Description
ProductKey	String	The product key of the specified device.
ProductName	String	The product name of the specified device.
DeviceName	String	The device name.
DeviceSecret	String	The device key.
IotId	String	The GUID issued by IoT Platform to the device.
UtcCreate	String	The time when the device was created, in UTC.

Name	Type	Description
GmtCreate	String	The time when the device was created, in GMT.
UtcActive	String	The time when the device was activated, in UTC.
GmtActive	String	The time when the device was activated, in GMT.
UtcOnline	String	The last time when the device went online, in UTC.
GmtOnline	String	The last time when the device went online, in GMT.
Status	String	The specified device status. Value: ONLINE OFFLINE UNACTIVE DISABLE
FirmwareVersion	String	The firmware version number of the device.
IpAddress	String	The IP address of the device.
NodeType	Integer	Indicates a node type. Values include: 0: Indicates a device node. A device node cannot contain sub-devices. It can connect to IoT Hub either directly or as a sub-device of a gateway. 1: Indicates a gateway node. A gateway can be mounted with sub-devices. It can manage sub-devices, maintain the topology of sub-devices, and synchronize the topology to the cloud.
Region	String	The region of the device (corresponding to the region in the IoT Platform console.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDeviceDetail
&ProductKey=a1rYuV*****
&DeviceName=SR8FiTu1R9tLUR2V1bmi
&common request parameters
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "DeviceName": "SR8FiTu1R9tLUR2V1bmi",
```



```

    "GmtActive": "2018-08-06 10:48:41",
    "UtcActive": "2018-08-06T02:48:41.000Z",
    "ProductKey": "a1rYuV*****",
    "DeviceSecret": "CPwUjMUgzdvaZv56TMy6773V3v3*****",
    "GmtCreate": "2018-08-06 10:47:50",
    "UtcCreate": "2018-08-06T02:47:50.000Z",
    "IotId": "SR8FiTu1R9tlUR2V1bmi0010*****",
    "Status": "ONLINE",
    "Region": "cn-shanghai",
    "NodeType": 0,
    "GmtOnline": "2018-08-06 13:43:12",
    "UtcOnline": "2018-08-06T05:43:12.000Z",
    "ProductName": "test",
    "IpAddress": "10.0.0.1",
    "FirmwareVersion": "V1.0"
  },
}

```

- XML format

```

<? xml version='1.0' encoding='utf-8'? >
<QueryDeviceDetailResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <DeviceName>SR8FiTu1R9tlUR2V1bmi</DeviceName>
    <GmtActive>2018-08-06 10:48:41</GmtActive>
    <UtcActive>2018-08-06T02:48:41.000Z</UtcActive>
    <ProductKey>a1rYuVF*****</ProductKey>
    <DeviceSecret>CPwUjMUgzdvaZv56TMy6773V3v3W*****</DeviceSecret>
  >
    <GmtCreate>2018-08-06 10:47:50</GmCreate>
    <UtcCreate>2018-08-06T02:47:50.000Z</UtcCreate>
    <IotId>SR8FiTu1R9tlUR2V1bmi0010a5*****</IotId>
    <Status>ONLINE</Status>
    <Region>cn-shanghai</Region>
    <NodeType>0</NodeType>
    <GmtOnline>2018-08-06 13:43:2</GmtOnline>
    <UtcOnline>2018-08-06T05:43:12.000Z</UtcOnline>
    <ProductName>test</ProductName>
    <IpAddress>10.0.0.1</IpAddress>
    <FirmwareVersion>V1.0</FirmwareVersion>
  </Data>
</QueryDeviceDetailResponse>

```

2.6.3 GetGatewayBySubDevice

Call this API to query the gateway device information using the sub-device information.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: GetGatewayBySubDevice.
IotId	String	No	The ID of the sub-device.

Name	Type	Required	Description
			 Note: If you specify this parameter, then you do not specify ProductKey or DeviceName . IoTId is the GUID. Each IoTId corresponds to a combination of ProductKey and DeviceName . If you specify both IoTId and the combination of ProductKey and DeviceName , the system follows IoTId .
ProductKey	String	No	The product key of the specified sub-device.  Note: If you specify this parameter, then you must also specify DeviceName .
DeviceName	String	No	The name of the sub-device.  Note: You must specify this parameter together with the ProductKey parameter.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	DeviceDetailInfo	The information of the gateway device. For the detailed information, see the following table.

Table 2-7: DeviceDetailInfo

Name	Type	Description
ProductKey	String	The product key of the gateway device
ProductName	String	The product name of the gateway device.
DeviceName	String	The gateway device name
DeviceSecret	String	The device secret of the gateway device.
lotId	String	The GUID issued by IoT Platform to the gateway device .
GmtCreate	String	The time when the gateway device was created, in GMT.
GmtActive	String	The time when the gateway device was activated, in GMT.
GmtOnline	String	The last time when the gateway device went online, in GMT.
UtcCreate	String	The time when the gateway device was created, in UTC.
UtcActive	String	The time when the gateway device was activated, in UTC.
UtcOnline	String	The last time when the gateway device went online, in UTC.
Status	String	The status of the gateway device. Value options: ONLINE OFFLINE UNACTIVE DISABLE
FirmwareVersion	String	The firmware version number of the gateway device.
IpAddress	String	The IP address of the gateway device.

Name	Type	Description
NodeType	Integer	The node type of the device, which is a value of 1 indicating that it is a gateway.
Region	String	The region of the gateway device.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=GetGatewayBySubDevice
&ProductKey=a1*****S
&DeviceName=XTzosqEOgxFXKPRgd8zl
&common request parameters
```

Response example

- JSON format

```
{
  "RequestId": "F227A41E-8A0F-4829-A1B1-727619DB58A3",
  "Data": {
    "Status": "UNACTIVE",
    "ProductName": "TEST",
    "DeviceSecret": "nIC0JkFJnG*****TWnvXHydEjX",
    "UtcOnline": "",
    "IotId": "9a1MTdk9brqQ2bhdG40Y001094fd00",
    "GmtCreate": "2018-08-07 15:54:15",
    "UtcCreate": "2018-08-07T07:54:15.000Z",
    "UtcActive": "",
    "GmtActive": "",
    "NodeType": 1,
    "Region": "cn-shanghai",
    "GmtOnline": "",
    "ProductKey": "a15tzTmkWZ2",
    "DeviceName": "d896e0ff000105bb",
    "IpAddress": "10.0.0.1",
    "FirmwareVersion": "1.2.3"
  },
  "Success": true
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<GetGatewayBySubDeviceResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <Status>UNACTIVE</Status>
    <ProductName>TEST</ProductName>
    <DeviceSecret>nIC0JkFJnG*****TWnvXHydEjX</DeviceSecret>
  </Data>
</GetGatewayBySubDeviceResponse>
```

```

    <UtcOnline></UtcOnline>
    <IotId>SR8FiTu1R9tlUR2V1bmi0010*****</IotId>
    <GmtCreate>2018-08-07 15:54:15</GmtCreate>
    <UtcCreate>2018-08-07T07:54:15.000Z</GmtOnline>
    <UtcActive></UtcActive>
    <GmtActive></GmtActive>
    <NodeType>1</NodeType>
    <Region>cn-shanghai</Region>
    <GmtOnline></GmtOnline>
    <ProductKey>a15tzTmkWZ2</ProductKey>
    <DeviceName>TEST</DeviceName>
    <IpAddress>10.0.0.1</IpAddress>
    <FirmwareVersion>1.2.3</FirmwareVersion>
  </Data>
</GetGatewayBySubDeviceResponse>

```

2.6.4 QueryDevice

You can call this operation to query the list of all devices under a specified product.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryDevice.
ProductKey	String	Yes	The product key of the device.
PageSize	Integer	No	The number of results per page. The maximum is 200. The default value is 10.
CurrentPage	Integer	No	The page of returned results to be displayed. The default value is 1.
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
PageCount	Integer	The total number of pages.
PageSize	Integer	The number of items displayed per page.

Name	Type	Description
Page	Integer	The current page number.
Total	Integer	The total number of items.
Data	Data	The list of device information returned when the call is successful. For more information, see DeviceInfo .

Table 2-8: DeviceInfo

Name	Type	Description
DeviceId	String	The device ID (deprecated).  Note: This parameter is deprecated. Do not use this to identify a device. If you want to search for the deviceId (the GUID of the device), use the QueryDeviceDetail operation and specify ProductKey and DeviceName .
DeviceName	String	The device name.
DeviceSecret	String	The device key.
GmtCreate	String	The time when the device was created.
GmtModified	String	The time when the device information was modified.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDevice
&ProductKey=al*****
&PageSize=10
&CurrentPage=1
&<Common request parameters>
```

Response example

- JSON format

```
{
  PageCount:1,
  Data:{
    DeviceInfo:[
      {DeviceId:..., DeviceName:..., ProductKey:...,
DeviceSecret:..., GmtCreate:Thu, 17-Nov-2016 02:08:12 GMT,
GmtModified:Thu, 17-Nov-2016 02:08:12 GMT}
    ],
  },
  PageSize:10,
```

```
Page:1,  
Total:9  
RequestId:06DC77A0-4622-42DB-9EE0-26A6E1FA08D3,  
Success:true,  
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >  
<QueryDeviceResponse>  
  <RequestId>9D25B7F9-DF08-4F77-B1E9-008F78522B37</RequestId>  
  <Success>true</Success>  
  <Page>1</Page>  
  <PageSize>10</PageSize>  
  <PageCount>1</PageCount>  
  <Total>6</Total>  
  <Data>  
    <DeviceInfo>  
      <DeviceId>...</DeviceId>  
      <DeviceName>...</DeviceName>  
      <DeviceSecret>...</DeviceSecret>  
      <GmtCreate>Thu, 10-Nov-2016 05:35:56 GMT</GmtCreate>  
      <GmtModified>Thu, 10-Nov-2016 05:35:56 GMT</GmtModifie  
d>  
      <DeviceStatus>NotActive</DeviceStatus>  
    </DeviceInfo>  
    ...  
  </Data>  
</QueryDeviceResponse>
```

2.6.5 DeleteDevice

You can call this operation to delete a specified device.

Descriptions

If a device is no longer needed, you can delete the device. After the device is deleted, the device ID (**IotId**) will be invalid, and any information associated with the device will be deleted. You will no longer be able to perform any action on the device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to DeleteDevice .
IotId	String	No	The device ID that you want to delete.  Note: If you set this parameter, you do not need to set ProductKey and DeviceName parameters. IotId , as the unique identifier of the device, corresponds

Name	Type	Required	Description
			to the combination of ProductKey & DeviceName . If you set both the IotId and the ProductKey&DeviceName , the IotId takes priority.
ProductKey	String	No	The identifier of the product that the devices belongs to.  Note: If you set this parameter, you must also set the parameter DeviceName .
DeviceName	String	No	The name of the device that you want to delete.  Note: If you set this parameter, you must also set the parameter ProductKey .
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicate whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error messages returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=DeleteDevice
&IotId=MpEKNuEUJzIORNANAWJX0010929900*****
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true
}
```

- XML format

```
<? xml version="1.0" encoding="UTF-8"? >
<DeleteDeviceResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
</DeleteDeviceResponse>
```

2.6.6 GetDeviceStatus

You can call this operation to view the running status of a specified device.

Request parameters:

Parameter name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to GetDeviceStatus.
IotId	String	No	The identifier of the specified device.  Note: If you specify this parameter, then you do not need to specify ProductKey or DeviceName. As the GUID of the device, IotId corresponds to the combination of ProductKey and DeviceName. If you specify both IotId and the combination of ProductKey&DeviceName, then IotId is used.
ProductKey	String	No	Specifies the product key of the specified device.  Note: If you specify this parameter, then you must also specify DeviceName.
DeviceName	String	No	The name of the specified device.  Note:

Parameter name	Type	Required	Description
			If you use this parameter, ProductKey is required.
Common request parameters		Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	Data	The device status returned when the call is successful. For more information, see DeviceStatusInfo .

Table 2-9: DeviceStatusInfo

Name	Type	Description
Status	String	The specified device status. Valid value: ONLINE OFFLINE UNACTIVE DISABLE

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=GetDeviceStatus
&IotId=MpEKNuEUJzIORNANAWJX0010929900*****
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true
}
```

```
"Data": {
  "Status": "UNACTIVE"
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<GetDeviceStatusResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <Status>UNACTIVE</Status>
  </Data>
</GetDeviceStatusResponse>
```

2.6.7 BatchGetDeviceState

You can call this operation to view the status of multiple specified devices under the same product.

Restrictions

You can view the status of up to 50 specified devices under the same product when you call this operation.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to BatchGetDeviceState .
ProductKey	String	Yes	The product key of the specified devices.
DeviceName	String	Yes	The names of the specified devices.  Note: You can search for up to 50 specified devices when you call this operation.
Common request parameters	-	Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.

Name	Type	Description
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
DeviceStatusList	DeviceStatus	The list of device status returned when the call is successful. For more information, see DeviceStatus .

Table 2-10: DeviceStatus

Name	Type	Description
DeviceName	String	The name of the specified device.
Status	String	The status of the specified device. Values include: online: The online status. offline: The offline status. unactive: Indicates that the specified device has not been activated. disable: Indicates that the specified device has been disabled.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=BatchGetDeviceState
&productKey=al*****
&DeviceName.1=device1
&DeviceName.2=device2
&<common request parameters>
```

Response example

- JSON format

```
{
  DeviceStatusList:{
    DeviceStatus:[
      {Status:UNACTIVE, DeviceName:device1},
      {Status:UNACTIVE, DeviceName:device2}
    ],
  },
  RequestId:"1A540BD7-176C-42D4-B3C0-A2C549DD00A3",
}
```

```
}    Success:true
```

- XML format

```
<? xml version="1.0" encoding="UTF-8"? >
  <BatchGetDeviceStateResponse>
    <RequestId>1AB5E6B0-AFCB-47B1-B6D4-C2BD32D63E14</RequestId>
    <Success>true</Success>
    <DeviceStatusList>
      <DeviceStatus>
        <Status>UNACTIVE</Status>
        <DeviceName>device1</DeviceName>
      </DeviceStatus>
      <DeviceStatus>
        <Status>UNACTIVE</Status>
        <DeviceName>device2</DeviceName>
      </DeviceStatus>
    </DeviceStatusList>
  </BatchGetDeviceStateResponse>
```

2.6.8 DisableThing

You can call this operation to disable a specified device.

Description

You can call this operation to disable a specified device. The disabled device cannot access IoT Platform. You cannot perform operations related this device. However, the information about this device is available in the system. You can use the **EnableThing** API operation to connect the disabled device to IoT Platform again.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to DisableThing.
IotId	String	No	The identifier of the device that you want to disable.  Note: If you use this parameter, ProductKey and DeviceName are not required. IotId is a globally unique identifier (GUID), and corresponds to a combination of ProductKey and DeviceName. If you use both IotId and the combination of ProductKey and DeviceName, the system follows IotId.

Name	Type	Required	Description
ProductKey	String	No	The product key of the device that you want to disable.  Note: If you use this parameter, DeviceName is required.
DeviceName	String	No	The name of the device that you want to disable.  Note: If you use this parameter, ProductKey is required.
Common request parameters	-	Yes	For more information about common request parameters, see Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=DisableThing
&IotId=MpEKNuEUJzIORNANAWJX0010929900*****
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<DisableThingResponse>
```

```
<RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
<Success>true</Success>
</DisableThingResponse>
```

2.6.9 EnableThing

You can call this operation to enable a device that has been disabled.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation you want to perform. Set the value to EnableThing .
lotId	String	No	Disabled device ID.  Note: If this parameter is provided, the ProductKey and DeviceName parameters are not required. lotId is the GUID. Each lotId corresponds to a combination of ProductKey and DeviceName . If you provide both lotId and the combination of ProductKey and DeviceName , lotId is used.
ProductKey	String	No	Key of the product to which a disabled device belongs.  Note: You must specify this parameter together with the DeviceName parameter.
DeviceName	String	No	Name of a disabled device.  Note: You must specify this parameter together with the ProductKey parameter.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether this call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.

Name	Type	Description
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=EnableThing
&IotId=SR8FiTu1R9tlUR2V1bmi0010a5****
&<Common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<EnableThingResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
</EnableThingResponse>
```

2.6.10 BatchCheckDeviceNames

Use this operation to customize multiple device names in a specified product. IoT Platform checks the validity of these names.

Restrictions

This operation needs to be used with the **BatchRegisterDeviceWithApplyId** operation to register and name multiple devices. You can customize up to 1,000 device names using this API.

You must first call **BatchCheckDeviceNames** to specify the device names that you want to create. IoT Platform checks the validity of the device names you submitted, and then responds with an application ID (**ApplyId**). Use **ApplyId** and call **BatchRegisterDeviceWithApplyId** to register multiple devices. If the device name list contains an invalid device name, then the device application process fails. You can call **QueryBatchRegisterDeviceStatus** to check the application status. The returned result lists the invalid device names.

Call **BatchRegisterDevice** if you want to register multiple devices under the same product without naming them. The system automatically generates the names.

Call **RegisterDevice** if you want to register one device under a product.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation you want to perform. Set the value to BatchCheckDeviceNames .
ProductKey	String	Yes	The product key of the devices to be registered.
DeviceName	String	Yes	The list of device names that you want to register. Each device name can contain 4 to 32 characters. They can contain letters, numbers, and special characters (hyphens, underscores, @ symbols, periods, and colons).  Note: In a single call, up to 1,000 device names can be specified.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	Data	The data returned when the call is successful.

Table 2-11: Data

Name	Type	Description
ApplyId	Long	The application ID returned by the system when a call is successful. You can use this ApplyId in the BatchRegisterDeviceWithApplyId call to register multiple devices.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=BatchCheckDeviceNames
&productKey=al*****
&DeviceName.1=device1
&DeviceName.3=device2
&DeviceName.2=device3
&DeviceName.4=device4
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "ApplyId": 4399
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<BatchCheckDeviceNamesResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <ApplyId>4401</ApplyId>
  </Data>
</BatchCheckDeviceNamesResponse>
```

2.6.11 BatchRegisterDeviceWithApplyId

You can call this operation to register multiple devices by using the application ID (ApplyId).

Limits

The operation is used in conjunction with the **BatchCheckDeviceNames** operation to enable registration of multiple devices with individually specified names under a product.

You call the **BatchCheckDeviceNames** operation to set the names of devices that you want to register. After the device names that you submitted are validated, IoT Platform returns the application ID (ApplyId). You can then register these devices by calling the **BatchRegisterDeviceWithApplyId** operation using the ApplyId.

If you want to register multiple devices under a product at a time with randomly generated device names, you can call the **BatchRegisterDevice** operation.

If you want to register a device individually under a product, you can call the **RegisterDevice** operation.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: BatchRegisterDeviceWithApplyId .
ProductKey	String	Yes	The identifier of the product for which you want to register devices.
ApplyId	Long	Yes	Application ID (ApplyId) of the devices to be registered. The ApplyId is returned by calling the BatchCheckDeviceNames operation.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicate whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	Data	The data returned when the call is successful.

Table 2-12: Data

Name	Type	Description
ApplyId	Long	The application ID (ApplyId) set for the request of registration.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=BatchRegisterDeviceWithApplyId
&ProductKey=al*****
&ApplyId=4416
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "ApplyId": 4416
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<BatchRegisterDeviceWithApplyId>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <ApplyId>4416</ApplyId>
  </Data>
</BatchRegisterDeviceWithApplyId>
```

2.6.12 BatchRegisterDevice

You can call this operation to register multiple devices under a specified product at a time with device names randomly generated.

Limits

You can call this operation to register multiple devices under a product and assign randomly generated device names. You can create up to 1,000 devices in a single call.

If you want to register a device individually under a product, you can call the **RegisterDevice** operation.

If you want to register multiple devices under a product at a time, and specify the name for each device individually, you need to call **BatchCheckDeviceNames** and **BatchRegisterDeviceWithApplyId** operations in order.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: RegisterDevice .
ProductKey	String	Yes	The identifier of the product for which you want to register devices.
Count	Integer	Yes	The number of registered devices, which cannot exceed 1,000.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. The <code>true</code> indicates that the call is successful. A value of the <code>false</code> indicates that the call has failed .
ErrorMessage	String	The error message returned when the call fails.
Data	Data	The data returned when the call is successful.

Table 2-13: Data

Name	Type	Description
ApplyId	Long	The application ID (ApplyId) returned for the request when the call is successful.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=BatchRegisterDevice
&ProductKey=al*****
&Count=10
&<Common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true ,
  "Data": {
    "ApplyId": 4415
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<BatchRegisterDeviceResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <ApplyId>4415</ApplyId>
  </Data>
```

```
</BatchRegisterDeviceResponse>
```

2.6.13 QueryBatchRegisterDeviceStatus

You can call this operation to query the processing status and result of device registration of multiple devices.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryBatchRegisterDeviceStatus .
ProductKey	String	Yes	The product key of the device.
ApplyId	Long	Yes	The application ID. The application ID is in the result of BatchRegisterDeviceWithApplyId or BatchRegisterDevice when they are successfully called.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	DeviceApplyStatusInfo	The application information returned when the call is successful. For more information, see DeviceApplyStatusInfo .

Table 2-14: DeviceApplyStatusInfo

Name	Type	Description
Status	String	The processing status and result. Values include: CHECK: Check. CHECK_SUCCESS: Check is successful. CHECK_FAILED: Check has failed. CREATE: Create. CREATE_SUCCESS: Create is successful.

Name	Type	Description
		CREATE_FAILED: Create has failed.
ValidList	List	A collection of device names that have been successfully created, represented by DeviceName.
InvalidList	List	A collection of DeviceNames that were not created successfully.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryBatchRegisterDeviceStatus
&ProductKey=AL*****
&ApplyId=4416
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "Status": "CREATE_SUCCESS",
    "ValidList": {
      "Name": [

    ]
    },
    "InvalidList": {
      "Name": [

    ]
    }
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<QueryBatchCheckDeviceNamesStatusResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <Status>CREATE_SUCCESS</Status>
    <ValidList/>
    <InvalidList/>
  </Data>
```

```
</QueryBatchCheckDeviceNamesStatusResponse>
```

2.6.14 QueryPageByApplyId

Use this operation to query the information of the multiple devices you have registered.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryPageByApplyId .
ApplyId	Long	Yes	Specifies the application ID. View the application ID in the returned results of BatchRegisterDeviceWithApplyId and BatchRegisterDevice .
PageSize	Integer	Yes	The number of items per page in the result.
CurrentPage	Integer	Yes	Specifies which page of the results is displayed.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. <code>true</code> indicates a successful call, and <code>false</code> indicates a failed call.
ErrorMessage	String	The error message returned when a call fails.
PageCount	Integer	The total number of pages.
PageSize	Integer	The number of records per page.
Page	Integer	The current page number.
Total	Integer	The total number of records.
ApplyDeviceList	ApplyDeviceInfo	The list of registered devices generated when the call is successful. For more information, see <code>ApplyDeviceInfo</code> .

Table 2-15: ApplyDeviceInfo

Name	Type	Description
DeviceId	String	The device ID (legacy edition).
		 Note:

Name	Type	Description
		This is a legacy parameter that cannot be used to identify a device. If you want to search for the deviceId (the GUID of the device), use the QueryDeviceDetail operation and specify ProductKey and DeviceName to view deviceId in the results. The DeviceName value can be viewed in the result of this operation.
DeviceName	String	The device name.
DeviceSecret	String	The device secret key.  Note: Keep this information confidential.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryPageByApplyId
&ApplyId=4416
&PageSize=10
&CurrentPage=1
&<common request parameters>
```

Response example

```
{
  "RequestId": "BB71E443-4447-4024-A000-EDE09922891E",
  "Success": true,
  Page: 1,
  PageSize: 10,
  PageCount: 1,
  Total: 4,
  ApplyDeviceList: {
    ApplyDeviceInfo: [
      {DeviceId: ..., DeviceName: ..., DeviceSecret: ...},
      {DeviceId: ..., DeviceName: ..., DeviceSecret: ...}
    ]
  }
}
```

```
}
```

2.6.15 QueryDeviceEventData

Use this operation to query the event records of a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryDeviceEventData .
lotId	String	No	Device ID.  Note: If you specify this parameter, then you do not need to specify ProductKey or DeviceName . As the unique identifier of the device, lotId corresponds to the combination of ProductKey and DeviceName . If you specify both lotId and the combination of ProductKey and DeviceName , then lotId is used.
ProductKey	String	No	Specifies the product key of the device.  Note: If you specify this parameter, then you must also specify DeviceName .
DeviceName	String	No	The name of the device.  Note: If you specify this parameter, you must also specify ProductKey .
EventType	String	No	Event type. Value: info: Information. alert: Alert. error: Error.
Identifier	String	Yes	The identifier of the event. To view the event identifier of the Pro Edition device, go to the Feature Definitions page of the Pro Edition product that the device belongs to in the console.

Name	Type	Required	Description
StartTime	Long	Yes	The start time of the event record.
EndTime	Long	Yes	The end time of the event record.
PageSize	Integer	No	The number of records per page in the result.
Asc	Integer	No	The sorting order of the event records in returned results. Values: 0: Descending. 1: Ascending.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call succeeded. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	Data	The device event record returned when the call is successful. For more information, see Data .

Table 2-16: Data

Name	Type	Description
List	List<EventInfo>	Event collection. Each element represents an event. See EventInfo for element structure.
NextValid	Boolean	Indicates whether the next page is available. A value of <code>true</code> indicates available, and a value of <code>false</code> indicates unavailable.
NextTime	Long	The start time of event record on the next page.

Table 2-17: EventInfo

Name	Type	Description
Name	String	Event name.
Time	Long	The time the event occurred.

Name	Type	Description
OutputData	String	The output parameter of the event, which is a string in MAP format. For more information about the data structure, see OutputData .
EventType	String	Event type.
Identifier	String	Event identifier.

Table 2-18: OutputData

Name	Type	Description
Key	String	Parameter identifier.
Value	Long	Value of parameter.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDeviceEventData
&IotId=abcdef1234567*****
&ProductKey=AL*****
&DeviceName=device1
&EventType=info
&Identifier=lightLevel
&StartTime=1516538300303L
&EndTime=1516541900303L
&PageSize=10
&Asc=1
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "NextValid": false,
    "NextTime": 1516449221253,
    "List": {
      "EventInfo": [
        {
          "Name": "testEventInfoName",
          "Time": "1516517974638",
          "OutputData": "{\"structArgs\":{\"structchildFLOATf71c20e\":1.23,\"structchildINT6b6b626\":3,\"structchildDATE663436a\": \"1516517966152\", \"structchildDOUBLE08d0f74\":1.23,\"structchildTEXTdc764f9\": \"07b68264b0ba42c18e5f\", \"structchildB00Ld260729\":0,\"structchildENUMbe62590\":1}, \"enumArgs\":0, \"boolArgs\":0,
```

```

    \"floatArgs\":2.3,\"dateArgs\": \"1516517966152\", \"intArgs\":1, \"
    doubleArgs\":2.3,\"textArgs\": \"dV56zbkzjBjw1Ti1dA52\"}],
    \"EventType\": \"info\",
    \"Identifier\": \"testEventInfo\"
  },
  {
    \"Name\": \"testEventInfoName\",
    \"Time\": \"1516449221254\",
    \"OutputData\": \"{\\\"structArgs\\\":{\\\"structchildFLOATf71c
    20e\\\":1.23,\\\"structchildINT6b6b626\\\":3,\\\"structchildDATE663436a\\\":
    \\\"1516449212507\\\",\\\"structchildDOUBLE08d0f74\\\":1.23,\\\"structchil
    dTEXTdc764f9\\\":\\\"a1f3583dde3944289639\\\",\\\"structchildB00Ld260729
    \\\":0,\\\"structchildENUMbe62590\\\":1},\\\"enumArgs\\\":0,\\\"boolArgs\\\":0,
    \\\"floatArgs\\\":2.3,\\\"dateArgs\\\": \"1516449212507\", \"intArgs\":1, \"
    doubleArgs\":2.3,\"textArgs\": \"1z4XNBvvA7eZw8XViaJp\"}],
    \"EventType\": \"info\",
    \"Identifier\": \"testEventInfo\"
  }
]
}
}
}

```

- XML format

```

<? xml version='1.0' encoding='utf-8'? >
<QueryDeviceEventDataResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <NextValid>false</NextValid>
    <NextTime>1516449221253</NextTime>
    <List>
      <EventInfo>
        <Name>testEventInfoName</Name>
        <Time>1516517974638</Time>
        <OutputData>{\"structArgs\":{\"structchildFLOATf71c
        20e\":1.23,\"structchildINT6b6b626\":3,\"structchildDATE663436a\":
        1516517966152\", \"structchildDOUBLE08d0f74\":1.23,\"structchildTEXTdc764
        f9\": \"07b68264b0ba42c18e5f\", \"structchildB00Ld260729\":0,\"structchil
        dENUMbe62590\":1}, \"enumArgs\":0,\"boolArgs\":0,\"floatArgs\":2.3,\"
        dateArgs\": \"1516517966152\", \"intArgs\":1,\"doubleArgs\":2.3,\"textArgs\":
        dV56zbkzjBjw1Ti1dA52\"}</OutputData>
        <EventType>info</EventType>
        <Identifier>testEventInfo</Identifier>
      </EventInfo>
      <EventInfo>
        <Name>testEventInfoName</Name>
        <Time>1516449221254</Time>
        <OutputData>{\"structArgs\":{\"structchildFLOATf71c
        20e\":1.23,\"structchildINT6b6b626\":3,\"structchildDATE663436a\":
        1516449212507\", \"structchildDOUBLE08d0f74\":1.23,\"structchildTEXTdc764
        f9\": \"a1f3583dde3944289639\", \"structchildB00Ld260729\":0,\"structchil
        dENUMbe62590\":1}, \"enumArgs\":0,\"boolArgs\":0,\"floatArgs\":2.3,\"
        dateArgs\": \"1516449212507\", \"intArgs\":1,\"doubleArgs\":2.3,\"textArgs\":
        1z4XNBvvA7eZw8XViaJp\"}</OutputData>
        <EventType>info</EventType>
        <Identifier>testEventInfo</Identifier>
      </EventInfo>
    </List>
  </Data>

```

```
</QueryDeviceEventDataResponse>
```

2.6.16 QueryDevicePropertyData

Use this operation to query the property records of a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to run. Set the value to QueryDevicePropertyData .
lotId	String	No	The device ID, which is the GUID for the device.  Note: If you specify this parameter, then you do not need to specify ProductKey or DeviceName . As the GUID of the device, lotId corresponds to the combination of ProductKey and DeviceName . If you specify both lotId and the combination of ProductKey and DeviceName , then lotId is used.
ProductKey	String	No	Specifies the product key of the device.  Note: If you specify this parameter, then you must also specify DeviceName .
DeviceName	String	No	Specifies the device name.  Note: If you specify this parameter, you must also specify ProductKey .
Identifier	String	Yes	The property identifier that you want to query. To view the property identifier of the Pro Edition device, go to the Define Feature page of the Pro Edition product that the device belongs to in the console.
StartTime	Long	Yes	The start time of the property record. The value is a millisecond value.

Name	Type	Required	Description
EndTime	Long	Yes	The end time of the property record. The value is a millisecond value.
PageSize	Integer	Yes	Shows the number of items per page in the result.
Asc	Integer	Yes	The sorting order of the property records in returned results. Value: 0: Descending. 1: Ascending.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	Data	The device property record returned when a call succeeds. For more information, see Data .

Table 2-19: Data

Name	Type	Description
List	List<PropertyInfo>	The collection of properties. Each element represents a property. See PropertyInfo for element structure.
NextValid	Boolean	Indicates whether the next page is available. <code>true</code> indicates available, and <code>false</code> indicates unavailable.
NextTime	Long	The start time of the property record on the next page.

Table 2-20: PropertyInfo

Name	Type	Description
Value	String	The property value.
Time	Long	The time when the property was modified.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDevicePropertyData
&IotId=abcdef1234567*****
&ProductKey=a1*****
&PeviceName=device1
&Identifier=lightLevel
&StartTime=1516538300303L
&EndTime=1516541900303L
&PageSize=10
&Asc=1
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "NextValid": true,
    "NextTime": 1516541821599,
    "List": {
      "PropertyInfo": [
        {
          "Value": "32",
          "Time": 1516541894876
        },
        {
          "Value": "2",
          "Time": 1516541885630
        },
        {
          "Value": "95",
          "Time": 1516541875947
        },
        {
          "Value": "14",
          "Time": 1516541830905
        },
        {
          "Value": "78",
          "Time": 1516541821600
        }
      ]
    }
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<QueryDevicePropertyDataResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
```



```

<NextValid>true</NextValid>
<NextTime>1516541821599</NextTime>
<List>
  <PropertyInfo>
    <Value>32</Value>
    <Time>1516541894876</Time>
  </PropertyInfo>
  <PropertyInfo>
    <Value>2</Value>
    <Time>1516541885630</Time>
  </PropertyInfo>
  <PropertyInfo>
    <Value>95</Value>
    <Time>1516541875947</Time>
  </PropertyInfo>
  <PropertyInfo>
    <Value>69</Value>
    <Time>1516541870871</Time>
  </PropertyInfo>
  <PropertyInfo>
    <Value>14</Value>
    <Time>1516541830905</Time>
  </PropertyInfo>
  <PropertyInfo>
    <Value>78</Value>
    <Time>1516541821600</Time>
  </PropertyInfo>
</List>
</Data>
</QueryDevicePropertyDataResponse>

```

2.6.17 QueryDeviceServiceData

Use this operation to query the service records of a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to run. Set the value to QueryDeviceServiceData .
IotId	String	No	Device ID.  Note: If you specify this parameter, then you do not need to specify ProductKey or DeviceName. As the GUID of the device, IotId corresponds to the combination of ProductKey and DeviceName. If you specify both IotId and the combination of ProductKey and DeviceName, then IotId is used.
ProductKey	String	No	Specifies the product key of the device.

Name	Type	Required	Description
			 Note: If you specify this parameter, then you must also specify DeviceName .
DeviceName	String	No	The device name.  Note: If you specify this parameter, then you must also specify ProductKey .
Identifier	String	Yes	The identifier of the service. To view the service identifier of the Pro Edition device, go to the Feature Definitions page of the Pro Edition product that the device belongs to in the console.
StartTime	Long	Yes	The start time of the service record.
EndTime	Long	Yes	The end time of the service record.
PageSize	Integer	No	The number of records per page in the result.
Asc	Integer	No	The sorting order of the service records in returned results. Values: 0: Descending. 1: Ascending.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	Data	The device service record returned when the call is successful. For more information, see Data .

Table 2-21: Data

Name	Type	Description
List	List<ServiceInfo>	Event collection. Each element represents a service. See ServiceInfo for element structure.
NextValid	Boolean	Indicates whether the next page is available. <code>true</code> indicates available, and <code>false</code> indicates unavailable.
NextTime	Long	The start time of service record on the next page.

Table 2-22: ServiceInfo

Name	Type	Description
Name	String	Service name.
Time	Long	The time when the service was executed.
OutputData	String	The output parameter of the service, which is a string in MAP format, following the key:value pattern.
InputData	String	The input parameter of the service, which is a string in MAP format, following the key:value pattern.
Identifier	String	Service identifier.

Example

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDeviceServiceData
&IotId=...
&ProductKey=...
&PeviceName=...
&Identifier=set
&StartTime=1516538300303L
&EndTime=1516541900303L
&PageSize=10
&Asc=1
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "NextValid": true,
    "NextTime": 1517315865197,
    "List": {
```

```

    "ServiceInfo": [
      {
        "Name": "set",
        "Time": 1517315865198,
        "OutputData": "{ \"code\":200, \"data\":{}, \"id\": \"100686\", \"message\": \"success\", \"version\": \"1.0\" }",
        "Inputdata": "{ \"lightadjusted Level \": 123 }",
        "Identifier": "set"
      }
    ]
  }
}

```

- XML format

```

<? xml version='1.0' encoding='utf-8'? >
<QueryDeviceServiceDataResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <NextValid>true</NextValid>
    <NextTime>1517315865197</NextTime>
    <List>
      <ServiceInfo>
        <Name>set</Name>
        <Time>1517315865198</Time>
        <OutputData>{"code":200,"data":{}, "id": "100686", "message": "success", "version": "1.0"}</OutputData>
        <InputData>{"LightAdjustLevel":123}</InputData>
        <Identifier>set</Identifier>
      </ServiceInfo>
    </List>
  </Data>
</QueryDeviceServiceDataResponse>

```

2.6.18 InvokeThingService

You can call this operation to run specified services on a device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to InvokeThingService.
lotId	String	No	The device ID.

 **Note:**
If you use this parameter, **ProductKey** and **DeviceName** are not required. **lotId** is the GUID. Each lotId corresponds to a combination of **ProductKey** and **DeviceName**. If you specify both **lotId**

Name	Type	Required	Description
			and the combination of ProductKey and DeviceName , the system follows IotId .
ProductKey	String	No	The product key of the device.  Note: If you specify this parameter, DeviceName is required.
DeviceName	String	No	The device name.  Note: If you specify this parameter, you must also specify ProductKey .
Identifier	String	Yes	The event identifier that you want to query. You can view the event identifier of a Pro Edition device on the Define Feature page of the Pro Edition product that the device belongs to in the console.
Args	Map	Yes	The arguments that you want to use to run the service. For more information, see Args .

Table 2-23: Args

Name	Type	Description
key	String	The parameter identifier.
value	Object	The parameter value.

Response parameters

Name	Type	Description
RequestId	String	The unique identifier generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Name	Type	Description
Code	String	The error code returned when the call fails. For more information about error codes, see Error codes .
Data	Data	The data returned when the call is successful. For more information, see the following table Data.

Table 2-24: Data

Name	Type	Description
Result	String	The result of a synchronous call.
MessageId	String	The ID of the message which IoT Platform sends to the device.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=InvokeThingService
&ProductKey=al*****
&DeviceName=device1
&Identifier=service
&Args=%7B%22service%22%3A1%7D
&common request parameters
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "Result": "...",
    "MessageId": "abcabc123"
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<InvokeThingServiceResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <Result>...</Result>
    <MessageId>abcabc123</MessageId>
  </Data>
```

```
</InvokeThingServiceResponse>
```

2.6.19 QueryDevicePropertyStatus

Use this operation to query the property snapshot in a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryDevicePropertyStatus .
IotId	String	No	The device ID that you want to query.  Note: If you specify this parameter, then you do not need to specify ProductKey or DeviceName . As the GUID of the device, IotId corresponds to the combination of ProductKey and DeviceName . If you specify both IotId and the combination of ProductKey and DeviceName , then IotId is used.
ProductKey	String	No	Specifies the product key of the device.  Note: If you specify this parameter, you must also specify DeviceName .
DeviceName	String	No	Specifies the device name that you want to query.  Note: If you specify this parameter, you must also specify ProductKey .
Common Request Parameters		Yes	See Common parameters .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	Data	The data returned when the call is successful.

Table 2-25: Data

Name	Type	Description
List	List	A collection of properties returned when the call is successful. For more information, see PropertyStatusInfo .

Table 2-26: PropertyStatusInfo

Name	Type	Description
Identifier	String	The property identifier.
Name	String	The property name.
DataType	String	The property format type. Values include: int: Integer. float: Float. double: Double. enum: Enumeration. bool: Boolean. text: Character. date: Time (string-type UTC timestamp in milliseconds). array: Array. struct: Object.
Time	String	The time when the property was modified.
Value	String	The property value.
Unit	String	The property unit.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDevicePropertyStatus
&IotId=SR8FiTu1R9tlUR2V1bmi0010*****
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "List": {
      "PropertyStatusInfo": [
        {
          "Name": "doublePropertyName",
          "Value": "50.0",
          "Time": "1517553572362",
          "DataType": "double",
          "Identifier": "doubleProperty",
          "Unit": "C"
        }
      ]
    }
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<QueryDevicePropertyStatusResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <List>
      <PropertyStatusInfo>
        <Time>1517553572362</Time>
        <Name>doublePropertyName</Name>
        <DataType>double</DataType>
        <Identifier>doubleProperty</Identifier>
        <Value>50.0</Value>
        <Unit>C</Unit>
      </PropertyStatusInfo>
    </List>
  </Data>
</QueryDevicePropertyStatusResponse>
```

2.6.20 SetDeviceProperty

You can call this operation to set properties for a specified device. A successful result of calling this API only indicates that IoT Platform has successfully sent a property setting request to the device, but does not indicate that the device SDK has successfully received the request and

performed the operation. Only if the device SDK responds to the request and sets the required property, you have successfully set the property for the device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to <code>SetDeviceProperty</code> .
lotId	string	No	The device ID.  Note: If you specify this parameter, ProductKey and DeviceName are not required. IotId is the GUID. Each IotId corresponds to a combination of ProductKey and DeviceName. If you specify both IotId and the combination of ProductKey & DeviceName, the system follows IotId.
ProductKey	String	No	The product key of the device.  Note: If you specify this parameter, DeviceName is required.
DeviceName	String	No	The device name.  Note: If you specify this parameter, you must also specify ProductKey.
Items	Map	Yes	For more information about the properties that you can set, see Items .

Table 2-27: Item

Name	Type	Description
key	String	The identifier of the property that you want to set. To view the event identifier of the Pro Edition device, go to the Define Feature page of the Pro Edition product that the device belongs to in the console.

Name	Type	Description
		 Note: You must specify a read-write property. This setting fails if you specify a read-only property.
value	Object	The property value. The property value, which must be within the data type and range of the property that you want to set.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Code	String	The error code returned when the call fails. For more information about error codes, see Error codes .
Data	Data	The data returned when the call is successful. See the following table Data.

Table 2-28: Data

Name	Type	Description
MessageId	String	The ID of the message that IoT Platform sends to the device.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=setDeviceProperty
&ProductKey=al*****
&DeviceName=device1
&Items=%7B%22LightAdjustLevel%22%3A1%7D
&common request parameters
```

Response example

- JSON format

```
{
```

```
"RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
"Success": true,
"Data": {
  "MessageId": "abcabc123"
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<SetDevicePropertyResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <MessageId>abcabc123</MessageId>
  </Data>
</SetDevicePropertyResponse>
```

2.6.21 QueryDeviceProp

You can call this operation to query the list of tags of a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryDeviceProp .
ProductKey	String	Yes	The product key of the device.
DeviceName	String	Yes	The device name.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Props	String	The list of device tags returned when the call is successful.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDeviceProp
&ProductKey=...
&DeviceName=...
```

&<common request parameters>

Response example

- JSON format

```
{  "Props":{"firmwareVersion":"12","color":"red"},  "RequestId":"F28BDA06-D8DB-4F70-BAED-DDA59AD6EF46",  "Success":true}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
<QueryDevicePropResponse>
  <RequestId>F28BDA06-D8DB-4F70-BAED-DDA59AD6EF46</RequestId>
  <Success>true</Success>
  <Props>{"firmwareVersion":"12","color":"red"}</Props>
</QueryDevicePropResponse>
```

2.6.22 SaveDeviceProp

You can call this operation to set a tag of a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to SaveDeviceProp .
ProductKey	String	Yes	The product key of the target device.
DeviceName	String	Yes	The name of the target device.
Props	String	Yes	For more information about the structure of each device tag that you can set, see Prop .  Note: - The device tag is in JSON format, and the prop is a device tag in string format. The prop consists of the key and the value, neither of which can be empty. - The total size of props does not exceed 5KB. - Each device has up to 100 tags. - You can modify or add up to 100 tags each time.

Table 2-29: Prop

Name	Type	Required	Description
key	String	Yes	The tag key, which must be 2 to 32 English characters in length. The specified tag key overwrites an existing identical tag key. If the specified tag key does not exist, the system adds this tag key.
value	Object	Yes	The tag key.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples**Request example**

```
https://iot.cn-shanghai.aliyuncs.com/?Action=SaveDeviceProp
&ProductKey=al*****
&DeviceName=device
&Props=abc:123
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<SaveDevicePropResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
```

```
</SaveDevicePropResponse>
```

2.6.23 DeleteDeviceProp

You can call this operation to delete the specified tag under a device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: DeleteDeviceProp .
ProductKey	String	Yes	The identifier of the product that the devices belongs to.
DeviceName	String	Yes	The name of the device that you want to delete.
Propkey	String	Yes	The key of the property for the device that you want to delete.  Note: IoT Platform will search the specified key in the tags of the target device, and then delete the tags associated with the key. If no records in the tags of the target device match the specified key, no action is performed.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=DeleteDeviceProp
&ProductKey=...
&DeviceName=...
&PropKey=temperature
```

& <common request parameters>

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<DeleteDevicePropResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
</DeleteDevicePropResponse>
```

2.6.24 GetThingTopo

You can call this operation to search for the topology of a specified device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to GetThingTopo .
IotId	String	No	The identifier of the target device.  Note: If you use this parameter, ProductKey and DeviceName are not required. IotId is a globally unique identifier (GUID), and corresponds to a combination of ProductKey and DeviceName . If you use both IotId and the combination of ProductKey and DeviceName , the system follows IotId .
ProductKey	String	No	The product key of the target device.  Note: If you use this parameter, DeviceName is required.
DeviceName	String	No	The name of the target device.

Name	Type	Required	Description
			 Note: If you use this parameter, ProductKey is required.
PageSize	Integer	Yes	The number of items per page displayed in the result.
PageNo	Integer	Yes	The page from which items start to be displayed in the result.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	DeviceDataInfo	The collection of device information in the response when the call was successful. For more information, see DeviceDataInfo .

Table 2-30: DeviceDataInfo

Name	Type	Description
PageCount	Integer	The total number of pages.
PageSize	Integer	The number of items per page displayed in the result.
List	List	A collection of device information. For more information about the structure of each element in the collection, see DeviceInfo .
CurrentPage	Integer	The current page.
Total	Integer	The total number of records.

Table 2-31: DeviceInfo

Name	Type	Description
Id	String	The identifier of the specified device.

Name	Type	Description
ProductKey	String	The product key of the specified device.
DeviceName	String	The name of the specified device.
DeviceSecret	String	The key of the specified device.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=GetThingTopo
&ProductKey=...
&DeviceName=...
&PageSize=10
&PageNo=1
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true
  "Data": {
    "PageCount": 1,
    "PageSize": 10,
    "List": {
      "deviceInfo": [
        {
          "DeviceName": "...",
          "ProductKey": "...",
          "IotId": "..."
        }
      ]
    },
    "CurrentPage": 1,
    "Total": 1
  },
},
}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
<GetThingTopoResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <PageCount>1</PageCount>
    <PageSize>10</PageSize>
    <List>
      <DeviceInfo>
        <DeviceName>...</DeviceName>
        <ProductKey>...</ProductKey>
        <DeviceSecret>...</DeviceSecret>
      </DeviceInfo>
    </List>
  </Data>
</GetThingTopoResponse>
```

```

        <IotId>...</IotId>
      </DeviceInfo>
    </List>
    <CurrentPage>1</CurrentPage>
    <Total>1</Total>
  </Data>
</GetThingTopoResponse>

```

2.6.25 NotifyAddThingTopo

You can call this operation to notify the specified gateway to add the topology.

Request parameters

name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to <code>NotifyAddThingTopo</code> .
GwlotId	String	No	The identifier or lotId of the gateway that you want to notify.  Note: If you specify this parameter, GwProductKey and GwDeviceName are not required. GwIotId is a globally unique identifier (GUID), and corresponds to a combination of GwProductKey and GwDeviceName. If you specify both GwIotId and the combination of GwProductKey and GwDeviceName, the system follows GwIotId.
GwProductKey	String	No	The product key of the product which the gateway device belongs to.  Note: If you specify this parameter, GwDeviceName is required.
GwDeviceName	String	No	The name of the gateway device.  Note: If you specify this parameter, GwProductKey is required.
DeviceListStr	List	Yes	The array of sub-devices that you want to mount to the specified gateway. The array is a JSON string. For more information about the array structure, see <code>DeviceList</code> .

Table 2-32: DeviceList

Name	Type	Required	Description
ProductKey	String	Yes	The product key of the specified sub-device.
DeviceName	String	Yes	The name of the specified sub-device.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Code	String	The error code returned when the call fails. For more information on error codes, see Error codes .
Data	Data	The data returned when the call is successful. For more information, see the following table Data.

Table 2-33: Data

Name	Type	Description
MessageId	String	The ID of the message which IoT Platform sends to the gateway device.

Examples**Request example**

```
https://iot.cn-shanghai.aliyuncs.com/?Action=NotifyAddThingTopo
&GwProductKey=...
&GwDeviceName=...
&DeviceListStr=[{"productKey":"...", "deviceName":"..."}, {"productKey
":"...", "deviceName":"..."}]
&common request parameters
```

Response example

- JSON format

```
{
  "RequestId": "419A3FC1-B517-4958-9414-5546765FA51F",
```

```
"Success": true,
"Data": {
  "MessageId": "abcabc123"
}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
<NotifyAddThingTopoResponse>
  <RequestId>419A3FC1-B517-4958-9414-5546765FA51F</RequestId>
  <Success>true</Success>
  <Data>
    <MessageId>abcabc123</MessageId>
  </Data>
</NotifyAddThingTopoResponse>
```

2.6.26 RemoveThingTopo

You can call this operation to remove the topology between a specified gateway and sub-device.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to RemoveThingTopo .
IoId	String	No	The identifier of the device that you want to delete.  Note: If you use this parameter, ProductKey and DeviceName are not required. IoId is a globally unique identifier (GUID), and corresponds to a combination of ProductKey and DeviceName . If you use both IoId and the combination of ProductKey and DeviceName , the system follows IoId .
ProductKey	String	No	The product key of the device that you want to delete.  Note: If you use this parameter, DeviceName is required.
DeviceName	String	No	The name of the device that you want to delete.

Name	Type	Required	Description
			 Note: If you use this parameter, ProductKey is required.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	Boolean	The result of removing the topology. A value of <code>true</code> indicates that the deletion is successful, and a value of <code>false</code> indicates that the deletion has failed.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=RemoveThingTopo
&IotId=CqXL5h5ysRTA4NxjABjj0010fa****
&<common request parameters>
```

Response example

- JSON format

```
{
  "Data": true,
  "RequestId": "098BEEF4-58F4-4181-A891-5D37DB6C4C3B",
  "Success": true
}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
<RemoveThingTopoResponse>
  <RequestId>098BEEF4-58F4-4181-A891-5D37DB6C4C</RequestId>
  <Success>true</Success>
  <Data>true</Data>
```

```
</RemoveThingTopoResponse>
```

2.6.27 QueryDeviceStatistics

Use this operation to query device statistics under a specified product.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set this value to QueryDeviceStatistics .
ProductKey	String	Yes	The product key of devices. The IoT platform returns the device statistics under the specified product.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when a call fails.
Data	StatisticsInfo	The device information returned when a call is successful. For more information, see StatisticsInfo .

Table 2-34: StatisticsInfo

Name	Type	Description
DeviceCount	Integer	The total number of devices.
OnlineCount	Integer	The number of devices online.
ActiveCount	Integer	The number of activated devices.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryDeviceStatistics
&ProductKey=...
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "57b144cf-09fc-4916-a272-a62902d5b207",
  "Success": true,
  "Data": {
    "DeviceCount": 100,
    "OnlineCount": 10,
    "ActiveCount": 50
  }
}
```

- XML format

```
<? xml version='1.0' encoding='utf-8'? >
<QueryDeviceStatisticsResponse>
  <RequestId>57b144cf-09fc-4916-a272-a62902d5b207</RequestId>
  <Success>true</Success>
  <Data>
    <DeviceCount>100</DeviceCount>
    <OnlineCount>10</OnlineCount>
    <ActiveCount>50</ActiveCount>
  </Data>
</QueryDeviceStatisticsResponse>
```

2.7 Manage Topics

2.7.1 CreateProductTopic

You can call this operation to create a product topic category for a specified product.

Restrictions

A product contains a maximum of 50 topic categories.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to CreateProductTopic .
ProductKey	String	Yes.	The product key that is used for creating the topic category.
TopicShort Name	String	Yes	The custom category level that is used in the topic category. By default, the topic category contains two category levels: <code>productKey</code> and <code>\${deviceName}</code> . They are separated by slashes (/). The topic category is in the format of <code>productKey/\${deviceName}/topicShortName</code> .

Name	Type	Required	Description
			 Note: Each category level contains letters, numbers, and underscores (_) only, and cannot be empty.
Operation	String	Yes	Operation permissions of the specified device on the topic category. Values include: SUB: Subscribe. PUB: Publish. ALL: Subscribe and publish.
Desc	String	No	The description of the topic category.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
TopicId	Long	The topic identifier that is generated by IoT Hub for the created topic category when the call is successful.  Note: Secure this topic identifier. You need to provide the topic identifier when calling the operation related to the topic category.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=CreateProductTopic
&ProductKey=...
&TopicShortName=submit
&Operation=PUB
&Desc=submit a test topic
&CommonParameters
```

Response example

```
{
```

```
"RequestId": "FCC27691-9151-4B93-9622-9C90F30542EC",  
"Success": true,  
"TopicId": 10000  
}
```

2.7.2 UpdateProductTopic

You can call this operation to modify the specified topic category.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: UpdateProductTopic .
TopicId	Long	Yes	The ID of the topic category that you want to modify.
TopicShort Name	String	No	The name of the user-defined category level that you want to set. By default, the topic category contains two category levels: <code>productKey</code> and <code>\${deviceName}</code> . The forward slash (/) is used to delimit the topic hierarchy, and the format is <code>productKey/\${deviceName}/topicShortName</code> .  Note: The name of each category level can contain letters, numbers, and underscores (_). A namespace cannot be empty.
Operation	String	No	The operation permission of the device on this topic category, value includes: SUB: Subscribe. SUB: Publish ALL: Publish and Subscribe.
Desc	String	No	Descriptions of topic categories.

Response example

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.

Name	Type	Description
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=UpdateProductTopic
&ProductKey=...
&TopicId=...
&TopicShortName=resubmit
&Operation=PUB
&Desc=resubmit a test topic
&<common request parameters>
```

Response example

```
{
  "RequestId": "FCC27691-9151-4B93-9622-9C90F30542EC",
  "Success": true
}
```

2.7.3 QueryProductTopic

You can call this operation to query the topic category of a specified product.

Request example

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryProductTopic .
ProductKey	String	Yes	The product key of the topic category.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	ProductTopicInfo	The list of topic categories returned when the call is successful. For more information, see ProductTopicInfo .

Table 2-35: ProductTopicInfo

Name	Type	Description
ProductKey	String	The product key.
Id	String	The topic ID of the topic category.
TopicShortName	String	The category levels except for productKey and \${deviceName}.
Operation	String	The permission that the device has to operate the topic category. Values include: 0: Publish. 1: Subscribe. 2: Publish and subscribe.
Desc	String	The description of the topic category.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryProductTopic
&ProductKey=...
&common request parameters
```

Response example

```
{
  "Data": [{
    "Id": "10000",
    "Operation": "0",
    "ProductKey": "HMyB***",
    "TopicShortName": "/HMyB***/${deviceName}/update"
  },
  {
    "Id": "10001",
    "Operation": "2",
    "ProductKey": "HMyB***",
    "TopicShortName": "/HMyB***/${deviceName}/update/error"
  },
  {
    "Id": "10002",
    "Operation": "1",
    "ProductKey": "HMyB***",
    "TopicShortName": "/HMyB***/${deviceName}/get"
  }
],
  "RequestId": "B953EAFF-CFF6-4FF8-BC94-8B89F018E4DD",
  "Success": true
}
```

```
}
```

2.7.4 DeleteProductTopic

You can call this operation to delete a specified topic category.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to DeleteProductTopic .
TopicId	Long	Yes	The identifier of the topic category that you want to delete.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=DeleteProductTopic
&TopicId=10000
&common_parameters
```

Response example

```
{
  "RequestId": "FCC27691-9151-4B93-9622-9C90F30542EC",
  "Success": true
}
```

```
}
```

2.7.5 CreateTopicRouteTable

You can call this operation to create the message routing relationship between topics.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to CreateTopicRouteTable .
SrcTopic	String	Yes	The source topic that has been subscribed to.
DstTopic	List	Yes	The list of target topics that you want to subscribe to and that are routed from SrcTopic .

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
FailureTopics	List	The list of topics that are returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=CreateTopicRouteTable
&SrcTopic=/CXi4***/device1/get
&DstTopic=[{/CXi4***/device2/get},{/CXi4***/device3/get}]
&<common request parameters>
```

Response parameters

```
{
  "RequestId": "FCC27691-9151-4B93-9622-9C90F30542EC",
  "Success": true,
  "FailureTopics": []
}
```

```
}
```

2.7.6 DeleteTopicRouteTable

You can call this operation to cancel topic forwarding for specified topics.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to DeleteTopicRouteTable .
SrcTopic	String	Yes	The source topic.
DstTopics	List	Yes	The list of target topics for the forwarding relationship that you want to cancel.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
FailureTopics	List	The list of topics for which the forwarding relationship the system has failed to cancel.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=DeleteTopicRouteTable
&SrcTopic=/CXi4***/device1/get
&DstTopics=[{/CXi4***/device2/get},{/CXi4***/device3/get}]
&common_parameters
```

Response example

```
{
  "RequestId": "FCC27691-9151-4B93-9622-9C90F30542EC",
  "Success": true,
  "FailureTopics": []
}
```

```
}
```

2.7.7 QueryTopicReverseRouteTable

You can call this operation to query a source topic that is subscribed to by a specified topic. This is known as the reverse route table.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryTopicReverseRouteTable .
Topic	String	Yes	The target topic, which receives messages.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
SrcTopics	List	The list of subscribed (source) topics returned when the call is successful.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryTopicReverseRouteTable
&Topic=/CXi4***/device21/get
&common_parameters
```

Response example

```
{
  "RequestId": "FCC27691-9151-4B93-9622-9C90F30542EC",
  "Success": true,
  "SrcTopics": ["/CXi4***/device1/get", "/CXi4***/device4/get"]
}
```



```
}
```

2.7.8 QueryTopicRouteTable

You can call this operation to query the target topics that subscribe to specified source topic. This process is known as querying the route table for the specified source topic. The operation only supports the query of user topics.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to QueryTopicRouteTable .
Topic	String	Yes	The source topic, which publishes messages.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
DstTopics	List	The list of target topics returned when the call is successful. Target topics subscribe to source topics.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=QueryTopicRouteTable
&Topic=CXi4***/device1/get
&common_parameters
```

Response example

```
{
  "RequestId": "FCC27691-9151-4B93-9622-9C90F30542EC",
  "Success": true,
  "DstTopics": ["/CXi4***/device2/get", "/CXi4***/device3/get"]
}
```

```
}
```

2.8 Error codes

This article lists the error codes which are returned when calls to IoT Platform APIs fail.

System error codes

An error code beginning with `iot.system` indicates a system error.

Error code	Description
<code>iot.system.SystemException</code>	A system exception occurred.

Common error codes

Error codes beginning with `iot.common` are common error codes.

Error code	Description
<code>iot.common.InvalidPageParams</code>	The specified page size or page number is invalid.
<code>iot.common.InvalidTenant</code>	The tenant is invalid.
<code>iot.common.QueryDeviceActionError</code>	Failed to query the device.
<code>iot.common.QueryDevicePropertyActionError</code>	Failed to query the device property.
<code>iot.common.QueryProductActionError</code>	Failed to query the product.
<code>iot.common.QueryProductCountActionError</code>	Failed to query the total number of products.
<code>iot.common.RamActionPermissionDeny</code>	You do not have the RAM permission to access the resource.

Error codes related to IoT Platform products

Error codes beginning with `iot.prod` are error codes related to IoT Platform products.

Error code	Description
<code>iot.prod.AlreadyExistedProductName</code>	The same product name already exists.
<code>iot.prod.CreateProductFailed</code>	Failed to create the product.
<code>iot.prod.CreateProductTopicFailed</code>	Failed to create the topic category.
<code>iot.prod.InvalidAliyunCommodityCode</code>	The value of <code>AliyunCommodityCode</code> is invalid.
<code>iot.prod.InvalidFormattedCatId</code>	The format of the device type is invalid.
<code>iot.prod.InvalidFormattedProductkey</code>	The format of the <code>ProductKey</code> is invalid.

Error code	Description
iot.prod.InvalidFormattedProductName	The format of the ProductName is invalid.
iot.prod.LongProductDesc	The product description length exceeds the maximum limit.
iot.prod.InvalidNodeType	The node type of the product is invalid.
iot.prod.NotExistedProduct	The specified product does not exist.
iot.prod.NotOpenID2Service	You have not activated the ID2 service.
iot.prod.NotSeniorProduct	The product is not a Pro Edition product.
iot.prod.NullProductKey	ProductKey cannot be empty.
iot.prod.NullProductName	ProductName cannot be empty.
iot.prod.ProductCountExceedMax	The total number of products exceeds the maximum limit.
iot.prod.QueryDeviceCountActionError	Failed to query the total number of devices.
iot.prod.QueryProductAbilitiesFailed	Failed to query the product feature.
iot.prod.QueryProductAbilityFailed	Failed to query the product feature.
iot.prod.QueryProductListActionError	Failed to query the product list.
iot.prod.UpdateProductFailed	Failed to update the product information.

Error codes related to devices

Error codes beginning with `iot.device` are error codes related to devices.

Error code	Description
iot.device.AddTopoRelationFailed	Failed to add topological relationship.
iot.device.AlreadyExistedDeviceName	The device name already exists.
iot.device.ApplyManyDevicesFailed	Failed to batch register devices.
iot.device.CreateDeviceFailed	Failed to create the device.
iot.device.CreateDeviceTaskIsRunning	The device creation task is still in progress.
iot.device.DeviceApplyIsNotFound	The ApplyId is not found.
iot.device.DeviceCountExceeded	Number of devices requested exceeds the maximum limit.
iot.device.DeleteDeviceFailed	Failed to delete the device.
iot.device.DeleteDevicePropertyFailed	Failed to delete the device property.

Error code	Description
iot.device.DisableDeviceFailed	Failed to disable the device.
iot.device.EnableDeviceFailed	Failed to enable the device.
iot.device.InactiveDevice	The device is inactive.
iot.device.InvalidFormattedApplyId	The ApplyId is incorrect.
iot.device.IncorrentDeviceApplyInfo	The device application information is not correct.
iot.device.InvalidFormattedDeviceName	The format of the device name is invalid.
iot.device.InvalidFormattedDevicePropertyKey	The format of the device property name is invalid.
iot.device.InvalidFormattedDevicePropertiesString	The format of device property string is invalid.
iot.device.InvalidIoTId	The device IoTId is incorrect.
iot.device.InvalidTimeBucket	The specified time range is invalid.
iot.device.InvokeThingServiceFailed	Failed to invoke the service.
iot.device.LongDevicePropertiesString	The length of the property string exceeds the maximum limit.
iot.device.NoneDeviceNameElement	The device name list cannot be empty.
iot.device.NoneDeviceProperties	No valid device property is found.
iot.device.NotExistedDevice	The specified device does not exist.
iot.device.NullApplyId	ApplyId cannot be empty.
iot.device.NullDeviceName	Device name cannot be empty.
iot.device.NullDevicePropertyKey	Device property name cannot be empty.
iot.device.NullDevicePropertiesString	Device property cannot be empty.
iot.device.QueryDeviceApplyActionError	An error occurred when the device application information was queried.
iot.device.QueryDeviceAttrDataHistoryFailed	Failed to query the device property data.
iot.device.QueryDeviceAttrStatusFailed	Failed to query the device property status.
iot.device.QueryDeviceEventHistoryFailed	Failed to query the device event data.
iot.device.QueryDeviceListActionError	Failed to query the device list.
iot.device.QueryDeviceServiceHistoryFailed	Failed to query the device service data.
iot.device.QueryDeviceStatisticsFailed	Failed to query the device statistics.

Error code	Description
iot.device.QueryDeviceStatusFailed	Failed to query the device status.
iot.device.QueryTopoRelationFailed	Failed to query the topological relationship.
iot.device.RemoveTopoRelationFailed	Failed to delete the topological relationship.
iot.device.SaveOrUpdateDevicePropertiesFailed	Failed to add or update the device property.
iot.device.SetDevicePropertyFailed	Failed to set device property.
iot.device.TooManyDevicePropertiesPerTime	The total number of device properties exceeds the limit.
iot.device.TopoRelationCountExceeded	The number of topology relationships exceeds the limit.
iot.device.VerifyDeviceFailed	Failed to verify the device.

Message related error codes.

Error codes beginning with `iot.messagebroker` are error codes related to message communication. This type of error code appears when you fail to call APIs related to message communication, device shadow, and rule engine. (For error codes related to rule engine APIs, see the following section.)

Error code	Description
iot.messagebroker.CreateTopicRouteFailed	Failed to create the route between the specified topics.
iot.messagebroker.CreateTopicTemplateException	An exception occurred when creating the topic category.
iot.messagebroker.CreateTopicTemplateFailed	Failed to create the topic category.
iot.messagebroker.DeleteTopicTemplateException	An exception occurred when deleting the topic category.
iot.messagebroker.DeleteTopicTemplateFailed	Failed to delete the topic category.
iot.messagebroker.DestTopicNameArraySizeIsLarge	The number of target topics for one source topic exceeds the limit.
iot.messagebroker.DeleteTopicRouteFailed	Failed to delete the route between the topics.
iot.messagebroker.DesireInfoInShadowMessageIsNotJson	desire information of device shadow is not in JSON format.

Error code	Description
iot.messagebroker.DesireValueIsNullInShadowMessage	desire information of device shadow cannot be empty.
iot.messagebroker.ElementKeyOrValueIsNullInDesire	desire information key or value is empty.
iot.messagebroker.ElementKeyOrValueIsNullInReport	report information key or value is empty.
iot.messagebroker.HALFCONN	Failed because the device is semi-connected.
iot.messagebroker.InvalidFormattedSrcTopicName	Format of the source topic is invalid.
iot.messagebroker.InvalidFormattedTopicName	Format of the topic is invalid.
iot.messagebroker.InvalidFormattedTopicTemplateId	Format of the topic category is invalid.
iot.messagebroker.InvalidTimeoutValue	The timeout value is invalid.
iot.messagebroker.InvalidTopicTemplateOperationValue	The operation value of the topic category is not correct.
iot.messagebroker.InvalidVersionValueInShadowMessage	The version value of device shadow is not correct.
iot.messagebroker.MethodValueIsNotUpdate	The method value of device shadow is not update.
iot.messagebroker.MessageContentIsNotBase64Encode	The message content is not base64 encoded.
iot.messagebroker.NoneElementInDesire	desire cannot be empty.
iot.messagebroker.NoneElementInReport	report cannot be empty.
iot.messagebroker.NoneElementDestTopicNameInArray	Target topic list cannot be empty.
iot.messagebroker.NotFoundDesireInShadowMessage	desire information of state of device shadow is not found.
iot.messagebroker.NotFoundMethodInShadowMessage	method information of device shadow is not found.
iot.messagebroker.NotFoundReportInShadowMessage	report information of device shadow is not found.
iot.messagebroker.NotFoundStateInShadowMessage	state information of device shadow is not found.

Error code	Description
iot.messagebroker.NotFoundVersionOrNullVersionValue	version information is not complete or the value of version is empty.
iot.messagebroker.NotMatchedProductKeyWithSrcTopicOwner	The product, which the source topic belongs to , does not belong to the current account.
iot.messagebroker.NullMessageContent	Message content cannot be empty.
iot.messagebroker.NullShadowMessage	Content of device shadow cannot be empty.
iot.messagebroker.NullSrcTopicName	Source topic name cannot be empty.
iot.messagebroker.NullTopicName	Topic cannot be empty.
iot.messagebroker.NullTopicTemplateId	Topic category cannot be empty.
iot.messagebroker.NullTopicTemplateOperation	Operation of the topic category cannot be empty.
iot.messagebroker.OFFLINE	Failed because the device is offline.
iot.messagebroker.PublishMessageException	An exception occurred when publishing message.
iot.messagebroker.PublishMessageFailed	Failed to publish message.
iot.messagebroker.QueryDeviceShadowActionError	Failed to query device shadow.
iot.messagebroker.QueryProductTopicListActionError	Failed to query topic category list.
iot.messageborker.QueryTopicReverseRouteTableListActionError	Failed to query source topic list.
iot.messageborker.QueryTopicRouteTableListActionError	Failed to query route table list.
iot.messagebroker.QueryTopicTemplateActionError	Failed to query topic category.
iot.messagebroker.QueryTopicTemplateException	An exception occurred when querying topic category.
iot.messagebroker.RateLimit	Failed because of rate limit.
iot.messagebroker.ReportInShadowMessagesNotJson	report information of state is not in JSON format.
iot.messagebroker.RrpcException	An exception occurred when sending message by PPRC.
iot.messagebroker.RrpcFailed	Failed to send message by RRPC.

Error code	Description
iot.messagebroker.ShadowMessageIsNotJson	Device shadow is not in JSON format.
iot.messagebroker.ShadowMessageLengthIsLarge	The length of device shadow exceeds the limit.
iot.messagebroker.TIMEOUT	Failed because of timeout.
iot.messagebroker.TooManyElementInDesire	The total number of desire elements exceeds the limit.
iot.messagebroker.TooManyElementInReport	The total number of report elements exceeds the limit.
iot.messagebroker.TopicAlreadyFound	The topic category name already exists.
iot.messagebroker.TopicTemplateCountExceedMax	The number of topic categories of the product exceeds the limit.
iot.messagebroker.TopicTemplateIsNotFound	The topic category is not found.
iot.messagebroker.UpdateDeviceShadowMessageFailed	Failed to update device shadow.
iot.messagebroker.UpdateTopicTemplateException	An exception occurred when updating the topic category.
iot.messagebroker.UpdateTopicTemplateFailed	Failed to update the topic category.

Error codes related to rule engine.

Error codes beginning with `iot.rule` and `iot.ruleng`, and some `iot.messagebroker` started error codes are related to rule engine.

Error code	Description
iot.rule.CreateRuleException	An exception occurred when creating the rule.
iot.rule.DeleteRuleFailed	Failed to delete the rule.
iot.rule.IncorrentRuleActionId	The rule action ID is not correct.
iot.rule.IncorrentRuleActionType	The type of the rule action is not correct.
iot.rule.IncorrentRuleId	The rule ID is not correct.
iot.rule.NullForwardDestForRule	The data forwarding destination cannot be empty.
iot.rule.NullSqlForRule	SQL statement for the rule cannot be empty.
iot.rule.NotFoundRule	The specified rule is not found.

Error code	Description
iot.rule.NotFoundRuleAction	The specified rule action is not found.
iot.rule.ParseRuleActionConfigError	Failed to parse the rule action configuration.
iot.rule.QueryRuleActionListError	Failed to query the rule action list.
iot.rule.QueryRulePageActionError	Failed to query the rule list.
iot.rule.RuleActionIsAlreadyCreated	The same rule action already exists.
iot.rule.RuleCountExceedMax	The total number of rules exceeds the limit.
iot.rule.RuleNameIsAlreadyExisted	The rule name already exists.
iot.rule.StartRuleFailed	Failed to start the rule.
iot.rule.StopRuleFailed	Failed to stop the rule.
iot.rule.TooManyRuleAction	The number of rule actions exceeds the limit.
iot.rule.UpdateRuleFailed	Failed to update the rule.
iot.ruleng.CreateRuleActionFailed	Failed to create the rule action.
iot.ruleng.DeleteRuleActionFailed	Failed to delete the rule action.
iot.ruleng.IncorrectType	The topic type is incorrect.
iot.ruleng.IncorrectSysTopic	The system topic is incorrect.
iot.ruleng.InvalidRamRole	The RAM role is invalid.
iot.ruleng.QueryRuleActionFailed	Failed to query the rule action.
iot.ruleng.RuleActionConfigurationIsNotJson	The rule action configuration is not in JSON format.
iot.ruleng.RuleAlreadyIsStarted	The rule is already in a started status.
iot.ruleng.NullRamRoleArn	roleArn cannot be empty.
iot.ruleng.NullRamRoleName	roleName cannot be empty.
iot.ruleng.NullRuleActionConfig	Rule action configuration cannot be empty.
iot.ruleng.NullRuleActionType	Rule action type cannot be empty.
iot.messagebroker.IncorrectRuleSql	The SQL statement of the rule is not correct.
iot.messagebroker.QueryRuleConfigActionException	An exception occurred when configuring the rule.

The following tables list the error codes related to different message targets.

Table 2-36: Error codes when the target is REPUBLISH (another IoT Platform topic)

Error code	Description
iot.messagebroker.InvalidFormattedTopicName	The theme name format is invalid.
iot.prod.NotExistedProduct	The specified product does not exist.
iot.common.QueryProductActionError	Failed to query the product.
iot.ruleng.IncorrectSysTopic	The system topic is not correct.
iot.messagebroker.NullTopicName	Topic name cannot be empty.

Table 2-37: Error codes when the target is OTS (Table Store)

Error code	Description
iot.ruleng.NullOtsInstanceName	The name of the Table Store instance cannot be empty.
iot.ruleng.NullTableNameInOtsInstance	The table name of the Table Store instance cannot be empty.
iot.ruleng.NullPrimaryKeyInOtsTable	Primary key of the table cannot not be empty.
iot.ruleng.NullPrimaryKeyNameInOts	The name of the primary key cannot be empty.
iot.ruleng.NullPrimaryKeyValueInOts	The value of the primary key cannot be empty.
iot.ruleng.IncorrectPrimaryKeyValueInOtsTable	The value of the primary key is incorrect.

Table 2-38: Error codes when the target is MNS (Message Service)

Error code	Description
iot.ruleng.NullTopicNameInMns	Name of the target theme cannot be empty.
iot.ruleng.NotFoundTopicInMns	The specified theme name is not found.
iot.ruleng.QueryMnsTopicListActionError	Failed to query the theme list in Message Service.

Table 2-39: Error codes when the target is FC (Function Compute)

Error code	Description
iot.ruleng.NullServiceNameInFc	The service name in Function Compute cannot be empty.

Error code	Description
iot.ruleng.NullFunctionNameInFc	The function name in Function Compute cannot be empty.
iot.ruleng.NotFoundServiceInFc	The specified service is not found in Function Compute.

Table 2-40: Error codes when the target is ONS (Message Queue)

Error code	Description
iot.messagebroker.NullTopicName	The target topic in Message Queue cannot be empty.

2.9 Rules engine

2.9.1 CreateRule

You can call this operation to create a rule for a specified topic.

Request parameters

**Note:**

Provide the following three parameters to enable the rule: ProductKey, ShortTopic, and Select.

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to CreateRule .
Name	String	Yes	Rule name. This can be 4 to 30 characters in length and can contain letters, numbers, and underscores (_).
ProductKey	String	No	The product key that applies this rule.
ShortTopic	String	No	The specified topic that applies this rule. Format: \${deviceName}/topicShortName. In the topic, set \${deviceName} to the specified device name, and set topicShortName to the custom category of the device. Call the QueryDevice operation to view all devices under the specified product. The response includes all DeviceName values.

Name	Type	Required	Description
			Call the QueryProductTopic operation to view all topic categories. The response includes all TopicShortName values.
Select	String	No	The SQL SELECT statement to be executed. The content follows the SQL expression.  Note: Set the parameter to the content in SELECT. For example, if the SELECT statement is <code>Select a, b, c</code>, set the parameter to <code>a, b, c</code>.
RuleDesc	String	No	The description of the rule.
DataType	String	No	The data format that is used in the rule. Values include: JSON: JSON data. BINARY: Binary data. The system uses the JSON format by default.
WHERE	String	No	The trigger condition for the rule. The content follows the SQL expression.  Note: The WHERE clause that is specified in this rule. For example, if the WHERE clause is <code>WHERE a>10</code>, set the parameter to <code>a>10</code>.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
RuleId	Long	The rule identifier that is generated by the rules engine for the rule when the call is successful.  Note:

Name	Type	Description
		Secure this rule identifier. You need to provide the rule identifier when calling the operation related to the rule.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=CreateRule
&Name=iot_test1
&ProductKey=al*****
&ShortTopic=open_api_dev/get
&Select=a,b,c
&RuleDesc=rule test
&DataType=JSON
&Where=a>10
&<common request parameters>
```

Response example

```
{
  "RequestId": "E4C0FF92-2A86-41DB-92D3-73B60310D25E",
  "RuleId": 1000,
  "Success": true
}
```

2.9.2 CreateRuleAction

Call the `CreateRuleAction` API to create a rule action under a specified rule and forward the processed topic data to other supported Alibaba Cloud services.

Limits

Click [here](#) to query the regions and Alibaba Cloud products supported by the rule engine.

After you call the **CreateRule** interface to create a rule, you can call this interface to create specific rule actions for the rule and define message forwarding targets.

A maximum of 10 rule actions can be created for a rule.

You can forward your topic messages to Message Service, Function Compute, and Table Store.



Note:

If you want to forward messages to ApsaraDB for RDS, you define the rule actions in the IoT Platform console.

Request parameters

Name	Type	Required	Description
Action	String	Yes	Action that you want to take. Set the value to <code>CreateRuleAction</code> .
RuleId	Long	Yes	ID of the rule for which you want to create an actions.
Type	String	Yes	The rule action type. Values include: MNS: Transmits the data in the topics that have been processed by the rule engine to Alibaba Cloud Message Service for message transmission. FC: Transmits the data in the topics that have been processed by the rule engine to Alibaba Cloud Function Compute for event computing. OTS: Transmits the data in the topics that have been processed by the rule engine to Alibaba Cloud Table Store for NoSQL data storage. REPUBLISH: Forwards the data in topics that have been processed by the rule engine to another IoT topic.
Configuration	String	Yes	Rule action configurations. The configurations for different rule action types are different. The following section describes their respective configurations.

Configuration of the REPUBLISH type**Table 2-41: REPUBLISH configuration**

Name	Description
topic	Target topic to which you forward data. System topics of Pro Edition products which can be used are as following: <code>/sys/\${YouProductKey}/\${YouDeviceName}/thing/service/property/set</code> <code>/sys/\${YouProductKey}/\${YouDeviceName}/thing/service/\${tsl.service.identifier}</code> For the

Name	Description
	variables, you enter your device information and TSL service identifier.
topicType	The topic type. 0 indicates the system topics of Pro Edition products. 1 indicates the customized topics.

Configuration examples of the REPUBLISH type

```
A system topic: {"topic":"/sys/a1TXXXXXWSN/xxx_cache001/thing/service/property/set","topicType":0}
A customized topic: {"topic":"/a1TXXXXXWSN/xxx_cache001/user/update","topicType":1}
```

Configuration of the OTS type

Table 2-42: OTS configuration

Name	Description
instanceName	Name of the instance that is used to receive information in Table Store.
tableName	Name of the data table that is used to receive information in Table Store.
regionName	Select the region where the target instance is located. See Regions and zones for RegionName.
role	Information about the authorized role. You assign a system service role to IoT Platform to authorize its access to your Table Store. The formats of the authorized role information are as follows: <pre>{ "roleArn": "acs:ram::6541***:role/aliyuniotaccessingotsrole", "roleName": "AliyunIOTAccessingOTSRole" }</pre> Replace 6541*** with your Alibaba Cloud account ID. Log on to the Alibaba Cloud console, and you can see your account ID on the Security Settings page of your account management. AliyunIOTAccessingOTSRole is a service role specified in RAM. The role is assigned to IoT Platform for Table Store access. For more information about roles, see the Role Management page in the RAM console.
primaryKeys	List of primary keys in the target table. For more information, see PrimaryKeys .

Table 2-43: PrimaryKeys

Name	Description
columnType	Type of the primary key. The value can be: INTEGER: Integer. STRING: String. BINARY: Binary data.
columnName	Name of the primary key.
columnValue	Value of the primary key.
option	Whether the primary key is an automatically incrementing column or not. The value can be AUTO_INCREMENT or empty. The primary key is an auto-incrementing column when its type is INTEGER and the value of Option is AUTO_INCREMENT.

Configuration example of the OTS type

```
{
  "instanceName": "testaaa",
  "tableName": "tt",
  "primaryKeys": [
    {
      "columnType": "STRING",
      "columnName": "ttt",
      "columnValue": "${tt}",
      "option": ""
    },
    {
      "columnType": "INTEGER",
      "columnName": "id",
      "columnValue": "",
      "option": "AUTO_INCREMENT"
    }
  ],
  "regionName": "cn-shanghai",
  "role": {
    "roleArn": "acs:ram::5645***:role/aliyuniotaccessingotsrole",
    "roleName": "AliyunIOTAccessingOTSRole"
  }
}
```

Configuration of the MNS type

Table 2-44: MNS configuration

Name	Description
themeName	Name of the target theme that is used to receive information in Message Service.

Name	Description
regionName	Region where the instance of the target Message Service is located. See Regions and zones for RegionName.
role	Information about the authorized role. You assign a system service role to IoT Platform to authorize its access to your Message Service. The formats of the authorized role information are as follows: <pre>{ "roleArn": "acs:ram::6541***:role/aliyuniotaccessingmnsrole", "roleName": "AliyunIOTAccessingMNSRole" }</pre> Replace 6541*** with your Alibaba Cloud account ID. Log on to the Alibaba Cloud console, and you can see your account ID on the Security Settings page of your account management. AliyunIOTAccessingMNSRole is a service role specified in RAM. The role is assigned to IoT Platform for Message Service access. For more information about roles, see the Role Management page in the RAM console.

Configuration example of the MNS type

```
{
  "themeName": "mns-test-topic1",
  "regionName": "cn-shanghai",
  "role": {
    "roleArn": "acs:ram::5645***:role/aliyuniotaccessingmnsrole",
    "roleName": "AliyunIOTAccessingMNSRole"
  }
}
```

Configuration of the FC type

Table 2-45: FC configuration

Name	Description
functionName	Name of the target function that is used to receive information in Function Compute.
serviceName	Name of the target service that is used to receive information in Function Compute.
regionName	Region where the instance of the target Function Compute is located. See Regions and zones for RegionName.

Name	Description
role	Information about the authorized role. You assign a system service role to IoT Platform to authorize its access to your Function Compute. The formats of the authorized role information are as follows: <pre>{ "roleArn": "acs:ram::6541***:role/aliyuniotaccessingfcrole", "roleName": "AliyunIOTAccessingFCRole" }</pre> Replace 6541*** with your Alibaba Cloud account ID. Log on to the Alibaba Cloud console, and you can see your account ID on the Security Settings page of your account management. AliyunIOTAccessingFCRole is a service role specified in RAM. The role is assigned to IoT Platform for Function Compute access. For more information about roles, see the Role Management page in the RAM console.

Configuration example of the FC type

```
{
  "regionName": "cn-shanghai",
  "role": {
    "roleArn": "acs:ram::5645***:role/aliyuniotaccessingfcrole",
    "roleName": "AliyunIOTAccessingFCRole"
  },
  "functionName": "weatherForecast",
  "serviceName": "weather"
}
```

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
ActionId	Long	Rule action ID generated by the rule engine for an action when the call succeeds. It is the identifier of the action.  Note: Save the information for future reference. When you call APIs related to rule actions, you may be required to enter the corresponding rule action IDs.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=CreateRuleAction
&RuleId=1000
&Type=REPUBLISH
&Configuration={"topic":"/a1*****/device1/get","topicType":1}
&common request parameters
```

Response example

```
{
  "RequestId": "21D327AF-A7DE-4E59-B5D1-ACAC8C024555",
  "ActionId": 10003,
  "Success": true
}
```

2.9.3 DeleteRule

You can call this operation to delete a specified rule.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to DeleteRule .
RuleId	Long	Yes	The identifier of the rule that you want to delete.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request parameters

```
https://iot.cn-shanghai.aliyuncs.com/?Action=DeleteRule
&RuleId=1000
```

```
&<common request parameters>
```

Response example

```
{
  "RequestId": "A8F48485-44B9-40D8-A56D-F716F384F387",
  "Success": true
}
```

2.9.4 DeleteRuleAction

You can call this operation to delete a specified rule action.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to DeleteRuleAction .
ActionId	Long	Yes	The identifier of the rule action that you want to delete.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=DeleteRuleAction
&ActionId=10001
&<common request parameters>
```

Response example

```
{
  "Requestid": "maid ",
  "Success": true
}
```

```
}
```

2.9.5 GetRule

You can call this operation to query details about a specified rule.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to <code>GetRule</code> .
RuleId	Long	Yes	The ID of the rule that you want to query.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
RuleInfo	RuleInfo	The rule details returned when the call is successful. For more information, see RuleInfo .

Table 2-46: RuleInfo

Name	Type	Description
CreateUserId	Long	The user ID that created the rule.
Created	String	The time when the rule was created.
DataType	String	The data type of the rule. Values include: <code>JSON</code> and <code>BINARY</code> .
Id	Long	The rule ID.
Modified	String	The most recent time when the rule was modified.
Name	String	The rule name.
ProductKey	String	The product key that applies the rule.
RuleDesc	String	The description of the rule.
Select	String	The values of Select in the SQL statements of the rule.

Name	Type	Description
ShortTopic	String	The specific topic to which this rule applies (does not contain the product key category) is in the following format: <code>\${deviceName}/topicShortName</code> . Where <code>\${deviceName}</code> specifies the name of the specific device, and <code>topicShortName</code> specifies the topic level customized for the specified device.
Status	String	The operating status of the rule. Values include: RUNNING: Running STOP: Stopped
Topic	String	The topic that applies the rule. The format is: <code>\${productKey}/\${deviceName}/topicShortName</code> .
Where	String	The where query condition in the SQL statements of the rule.
topicType	Integer	If you have set SQL statements for the rule, the returned values are: 0: indicates the upstream topics for products in the IoT Platform Pro. 1: indicates the customized topics. If you have not set SQL statements for the rule, the returned value is -1.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=GetRule
&RuleId=1000
&<common request parameters>
```

Response example

```
{
  "RequestId": "A8F48485-44B9-40D8-A56D-F716F384F387",
  "RuleInfo": {
    "CreateUserId": 32164***,
    "Created": "Sun Apr 08 13:42:50 CST 2018",
    "DataType": "JSON",
    "Id": 1000,
    "Modified": "Sun Apr 08 13:42:50 CST 2018",
    "Name": "rule_test_pop123",
    "ProductKey": "a1GMr***",
    "RuleDesc": "This is a test rule.",
    "Select": "a,b,c",
    "ShortTopic": "test_topic/get",
```

```
"Status": "STOP",
"Topic": "/a1GMr***test_topic/get",
"Where": "a>1"
"topicType": "1"
},
"success": true
}
```

2.9.6 GetRuleAction

You can call this operation to list information about the specified rule action.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: GetRuleAction .
ActionId	Long	Yes	The rule action ID that you want to query.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicate whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
RuleActionInfo	RuleActionInfo	The details of the rule action returned when the operation call is successful. For more information, see RuleActionInfo .

Table 2-47: RuleActionInfo

Name	Type	Description
Id	Long	The ID of rule action.
RuleId	Long	The ID of the rule that includes this rule action.
Type	String	The rule action type. Values include: REUBLISH: Forward to another topic. OTS: Save to Table Store. MNS: Send to Message Service (MNS). ONS: Send to Message Queue. HiTSDB: Save to High-Performance Time Series Database (HiTSDB).

Name	Type	Description
		FC: Send to Function Compute. DATAHUB: Send to the DataHub. RDS: Save to cloud databases.
Configuration	String	The configurations of the specified rule action.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=GetRuleAction
&ActionId=10001
&<common request parameters>
```

Response example

```
{
  "RequestId": "8FC9E36B-E0DC-4802-84EE-184E255B4E95",
  "RuleActionInfo": {
    "Configuration": "{\"endPoint\":\"http://ons.aliyun.com:8080/rocketmq/nsaddr4client\", \"regionName\":\"cn-shanghai\", \"topic\":\"test_hhf\", \"uid\":\"3216***\"}",
    "Id": 10001,
    "RuleId": 1000,
    "Type": "ONS"
  },
  "Success": true
}
```

2.9.7 ListRule

You can call this operation to query the list of rules by page.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to ListRule .
PageSize	Integer	Yes	The number of results per page. The maximum value is 100. The default value is 10.
CurrentPage	Integer	Yes	The page of returned results to be displayed. The maximum value is 1000. The default value is 1.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
Data	RuleInfo	The list of rules returned when the call is successful. For more information, see RuleInfo .
PageSize	Integer	The number of records per page.
Page	Integer	The current page number.
Total	Integer	The total number of pages.

Table 2-48: RuleInfo

Name	Type	Description
CreateUserId	Long	The user ID that created the rule.
Created	String	The time when the rule was created.
DataType	String	The data type of the rule. Values include: JSON and BINARY.
Id	Long	The rule ID.
Modified	String	The most recent time when the rule was modified.
Name	String	The rule name.
ProductKey	String	The product key that applies the rule.
RuleDesc	String	The description of the rule.
Select	String	The values of Select in the SQL statements of the rule.
ShortTopic	String	The topic that applies the rule. The short topic format does not include <code>/\${productKey}</code> . The format is: <code>\${deviceName}/\${topicShortName}.\${deviceName}</code> is the name of the specified device. <code>topicShortName</code> is the custom category level of the device.
Status	String	The operating status of the rule. Values include: RUNNING: Running STOP: Stopped

Name	Type	Description
Topic	String	The topic that applies the rule. The format is: \${productKey}/\${deviceName}/topicShortName.
Where	String	The where query condition in the SQL statements of the rule.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=ListRule
&PageSize=10
&CurrentPage=1
&<common request parameters>
```

Response example

```
{
  "Data": [{
    "CreateUserId": 32164***,
    "Created": "Sun Apr 08 13:42:50 CST 2018",
    "DataType": "JSON",
    "Id": 1000,
    "Modified": "Sun Apr 08 13:42:50 CST 2018",
    "Name": "rule_test_pop123",
    "ProductKey": "HMyBnl***",
    "RuleDesc": "Create a rule and test",
    "Select": "a,b,c",
    "ShortTopic": "test_topic/get",
    "Status": "STOP",
    "Topic": "/HMyBnl***test_topic/get",
    "Where": "a>1"
  },
  {
    "CreateUserId": 32164***,
    "Created": "Sun Apr 08 11:29:05 CST 2018",
    "DataType": "BINARY",
    "Id": 1001,
    "Modified": "Sun Apr 08 11:33:05 CST 2018",
    "Name": "Create a rule and test",
    "ProductKey": "HMyBnl***",
    "Select": "*",
    "ShortTopic": "test_topic/get",
    "Status": "RUNNING",
    "Topic": "/HMyBnl***test_topic/get"
  }
],
  "Page": 1,
  "PageSize": 2,
  "RequestId": "43ED5445-90B1-4A97-95A1-8BF6F48CD6E9",
  "Success": true,
  "Total": 30
}
```

```
}
```

2.9.8 ListRuleActions

You can call this operation to retrieve all rule actions of a specified rule.

Request parameters

Name	Type	Required	Description
Action	String	Yes.	The operation that you want to perform. Set the value to ListRuleActions .
RuleId	String	Yes	The ID of the rule that you want to query.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
RuleActionList	RuleActionInfo	The rule action information returned when the call is successful. For more information, see RuleActionInfo.

Table 2-49: RuleActionInfo

Name	Type	Description
Id	Long	The rule action ID.
RuleId	Long	The ID of the rule that includes this rule action.
Type	String	The rule action type. Values include: REUBLISH: Forward to another topic. OTS: Save to Table Store. MNS: Send to Message Service. ONS: Send to Message Queue. HiTSDB: Save to High-Performance Time Series Database. FC: Send to Function Compute. DATAHUB: Send to DataHub. RDS: Save to cloud databases.
Configuration	String	The configurations of this rule action.

Examples

Request parameters

```
https://iot.cn-shanghai.aliyuncs.com/?Action=ListRuleActions
&RuleId=10000
&<common request parameters>
```

Response example

```
{
  "RequestId": "21D327AF-A7DE-4E59-B5D1-ACAC8C024555",
  "RuleActionList": [{
    "Configuration": "{\"topic\":\"/a1GM***/bike/get\", \"uid\":\"3216***\"}",
    "Id": 10001,
    "RuleId": 1000,
    "Type": REPUBLISH
  },
  {
    "Configuration": "{\"endPoint\":\"http://ons.aliyun.com:8080/rocketmq/nsaddr4client\", \"regionName\":\"cn-shanghai\", \"topic\":\"test_ons\", \"uid\":\"3216***\"}",
    "Id": 10002,
    "RuleId": 1000,
    "Type": ONS
  }
],
  "Success": true
}
```

2.9.9 StartRule

You can call this operation to start the specified rule.

Request parameters



Note:

You need to verify that the rule has SQL expressions configured before you start the rule. To configure SQL expressions, see the [CreateRule](#) API.

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: StartRule .
RuleId	Long	Yes	The ID for the rule to be triggered.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.

Name	Type	Description
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request parameters

```
https://iot.cn-shanghai.aliyuncs.com/?Action=StartRule
&RuleId=1000
&<common request parameters>
```

Response parameters

```
{
  "RequestId": "9A2F243E-17FE-4846-BAB5-D02A25155AC4",
  "Success": true
}
```

2.9.10 StopRule

You can call this operation to stop the specified rule.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: StopRule .
RuleId	Long	Yes	The ID of the rule that you want to stop.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error messages returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=StopRule
&RuleId=10000
&<common request parameters>
```

Response example

```
{
  "RequestId": "9A2F243E-17FE-4846-BAB5-D02A25155AC4",
  "Success": true
}
```

2.9.11 UpdateRule

You can call this operation to modify the specified rule.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to: UpdateRule .
RuleId	Long	Yes	The rule ID that you want to modify.
Name	String	Yes	The rule name. The rule name must meet the following requirements: consists of 4-30 characters, including Chinese characters (Each Chinese character counts as two characters), English letters, numbers, and underscores (_).
ProductKey	String	No	The identifier of the product that uses this rule.
ShortTopic	String	No	The specific topic to which the rule is applied, and the format is: <code>\${deviceName}/topicShortName</code> Where <code>\${deviceName}</code> specifies the name of the specific device, and <code>topicShortName</code> specifies the topic level customized for the specified device. You can call the QueryDevice operation to view all the devices under the product. The result contains all the devices with the name matching the value of DeviceName . You can call the QueryProductTopic operation to view all the topic categories under the product. The result contains all the

Name	Type	Required	Description
			topics with the name matching the value of TopicShortName .
Select	String	No	The SQL SELECT statement that you want to execute. For more information, see the SQL expressions.  Note: Set the parameters following the Select keyword. For example, if the Select statement is <code>Select a, b, c</code>, set the parameters to the strings <code>a, b, c</code>.
RuleDesc	String	No	The descriptions of the rule.
Where	String	No	The trigger of the rule. For more information, see the SQL expressions.  Note: Set it to the strings after where. For example, if the where clause is <code>where a > 10</code>, set it to <code>a>10</code>.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the operation call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?Action=UpdateRule
&RuleId=1000
&Name=test_2
&ProductKey=al*****
&ShortTopic=+/get
&Select=a,b,c
&RuleDesc=test
&Where=a>10
```

&<common request parameters>

Response example

```
{
  "RequestId": "9A2F243E-17FE-4846-BAB5-D02A25155AC4",
  "Success": true
}
```

2.10 Communications

2.10.1 Pub

You can call this operation to publish messages to a specified topic.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to Pub .
ProductKey	String	Yes	The product key that publishes messages.
TopicFullName	String	Yes	The full name of the topic that receives messages. Both the user topic category and the customized topic category can be specified. The system topic category cannot be specified. - User topic category: <code>/\${productKey}/\${deviceName}/update</code> - Custom topic category: <code>productKey/\${deviceName}/topicShortName</code> <code>topicShortName</code> is the category level you have customized when you created the topic. You can call QueryProductTopic to query the list of all topics under a specified product.
MessageContent	String	Yes	The body of the message to be published. You need to convert the message into binary data and perform Base64 encoding to generate the message body.
Qos	Integer	No	Specifies the quality of service. Values include :

Name	Type	Required	Description
			0: At most once. 1: At least once. If you do not specify this parameter, the system uses 0 by default.  Note: A message can be stored on IoT Platform for up to seven days.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
MessageId	String	The message ID generated by the cloud when the message is sent.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?&Action=Pub
&ProductKey=...
&TopicFullName=%2FproductKey%2FdeviceName%2Fget
&MessageContent=aGVsbG8gd29ybGQ%3D
&Qos=0
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "BB71E443-4447-4024-A000-EDE09922891E",
  "Success": true,
  "MessageId": 889455942124347329
}
```

- XML format

```
<? xml version="1.0" encoding="UTF-8"? >
  <PubResponse>
    <RequestId>BB71E443-4447-4024-A000-EDE09922891E</RequestId>
```

```
<Success>true</Success>
<MessageId>889455942124347329</MessageId>
</PubResponse>
```

2.10.2 PubBroadcast

You can call this operation to broadcast messages to all topics of a specified product.

Request parameters

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to PubBroadcast .
ProductKey	String	Yes	The product key that broadcasts messages.
TopicFullName	String	Yes	The full name of the topic that receives broadcast messages. The format is: <code>/broadcast/\${productKey}/your custom field</code>. <code>\${productKey}</code> is the product key that receives broadcast messages. Up to 1,000 devices can subscribe to a broadcast topic. If the number of devices exceeds the maximum, you can group the devices. For example, if you have 5,000 devices, divide them into five groups each containing 1,000 devices. You need to call the broadcasting topic five times. Set the custom field to group 1, group 2, group 3, group 4, and group 5 respectively, and subscribe each group of devices to the five broadcasting topics.
MessageContent	String	Yes	The body of the message to be published. You need to convert the message into binary data and perform Base64 encoding to generate the message body.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.

Name	Type	Description
ErrorMessage	String	The error message returned when the call fails.
MessageId	String	The message ID generated by the cloud when the message is sent.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?&Action=PubBroadcast
&ProductKey=al*****
&TopicFullName=%252Fbroadcast%252FUPqSxj2vXXX%252Fxxx
&MessageContent=aGVsbG93b3JsZA==
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "BB71E443-4447-4024-A000-EDE09922891E",
  "Success": true,
}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
  <PubBroadcastResponse>
    <RequestId>BB71E443-4447-4024-A000-EDE09922891E</RequestId>
    <Success>true</Success>
  </PubBroadcastResponse>
```

2.10.3 RRpc

You can call this operation to send a request to a specified device and receive a response from the device synchronously.

Request parameters:

Parameter name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to RRpc.
ProductKey	String	Yes	The product key of the product that includes the device that will receive your request.
DeviceName	String	Yes	The name of the device that will receive your request.

Parameter name	Type	Required	Description
RequestBase64Byte	String	Yes	The string that is generated by encoding your request message using the Base64 encoding scheme.
Timeout	Integer	Yes	The wait time for a device to respond. This time is measured in milliseconds (ms), and the value range is 1,000 to 5,000.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.
MessageId	String	The message ID automatically generated after a request is sent successfully. The message ID is used to identify the request message.
RrpcCode	String	The status code returned when the call is successful. The status code is used to identify the request status. Values include: UNKNOWN: The system is abnormal. SUCCESS: The operation is successful. TIMEOUT: Waiting for device response times out. OFFLINE: The specified device is offline. HALFCONN: The specified device is offline. In this status, the device is offline but it has not been offline for a full heartbeat cycle.
PayloadBase64Byte	String	The Base64-encoded payload.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?&Action=RRpc
&ProductKey=al*****
&DeviceName=device1
&RequestBase64Byte=aGVsbG8gd29ybGQ%3D
&TimeOut=1000
```

&CommonRequestParameters**Response example**

- JSON format

```
{
  "RequestId": "41C4265E-F05D-4E2E-AB09-E031F501AF7F",
  "Success": true,
  "RpcCode": "SUCCESS",
  "PayloadBase64Byte": "d29ybGQgaGVsbG8="
  "MessageId": 889455942124347392
}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
  <RRpcResponse>
    <RequestId>41C4265E-F05D-4E2E-AB09-E031F501AF7F</RequestId/>
    <Success>true</Success>
    <RpcCode>SUCCESS</RpcCode>
    <PayloadBase64Byte>d29ybGQgaGVsbG8=</PayloadBase64Byte>
    <MessageId>889455942124347392</MessageId>
  </RRpcResponse>
```

2.11 Device shadows

2.11.1 GetDeviceShadow

You can call this operation to obtain information about the shadow of the specified device.

Request parameters

Name	Type	Required	Request parameters
Action	String	Yes	The operation that you want to perform. Set the value to GetDeviceShadow .
ProductKey	String	Yes	The key of the product that the device belongs to.
DeviceName	String	Yes	Name of the device.

Response parameters

Name	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.

Name	Type	Description
ErrorMessage	String	The error message returned when the call fails.
ShadowMessage	String	The shadow information returned when the call is successful.  Note: The structure of the shadow information varies depending on the status of the device. For more information, see Configure device shadows .

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?&Action=GetDeviceShadow
&ProductKey=al*****
&DeviceName=device1
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId": "BB71E443-4447-4024-A000-EDE09922891E",
  "Success": true,
  "ShadowMessage": {...}
}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
<GetDeviceShadowResponse>
  <RequestId>BB71E443-4447-4024-A000-EDE09922891E</RequestId>
  <Success>true</Success>
  <ShadowMessage>{...}</ShadowMessage>
</GetDeviceShadowResponse>
```

2.11.2 UpdateDeviceShadow

You can call this operation to modify the shadow information for the specified device.

Request parameters:

Name	Type	Required	Description
Action	String	Yes	The operation that you want to perform. Set the value to UpdateDeviceShadow.

Name	Type	Required	Description
ProductKey	String	Yes	The identifier of the device where you want to modify the shadow message.
DeviceName	String	Yes	The name of the device where you want to modify the shadow message.
ShadowMessage	String	Yes	The new message of the device shadow you modified. Examples: <pre>{ "method": "update", "state": { "desired": { "color": "green" } }, "version": 2 }</pre> see ShadowMessage .

Table 2-50: ShadowMessage

Name	Type	Required	Description
method	String	Yes	The specified action type. Set the value to update.
state	String	Yes	The specific state that is sent to shadow by the device, where desired is the desired state of shadow.
version	String	Yes	The version of the device shadow must be higher than the current version.

Return parameters

Parameters	Type	Description
RequestId	String	The GUID generated by Alibaba Cloud for the request.
Success	Boolean	Indicates whether the call is successful. A value of <code>true</code> indicates that the call is successful. A value of <code>false</code> indicates that the call has failed.
ErrorMessage	String	The error message returned when the call fails.

Examples

Request example

```
https://iot.cn-shanghai.aliyuncs.com/?&Action=UpdateDeviceShadow
&ProductKey=a[*****]
&DeviceName=device1
&ShadowMessage=[{"method":"update","state":{"desired":{"color":"green"}},
"reported":"","version":1}]
&<common request parameters>
```

Response example

- JSON format

```
{
  "RequestId":"BB71E443-4447-4024-A000-EDE09922891E",
  "Success":true,
}
```

- XML format

```
<? xml version='1.0' encoding='UTF-8'? >
  <UpdateDeviceShadowResponse>
    <RequestId>BB71E443-4447-4024-A000-EDE09922891E</RequestId>
    <Success>true</Success>
  </UpdateDeviceShadowResponse>
```