API Gateway

User Guide for Providers

User Guide for Providers

Overview

API Gateway provides high-performance and highly available API hosting service to help users to publish or access to the APIs on Alibaba Cloud products such as ECS and Container Service. It manages the entire API lifecycle from release and management to maintenance. You can quickly open data or services at low costs and risks through simple operations.

API Gateway provides the following features:

API management

You can manage the lifecycle of an API, including creation, testing, release, deprecation, and version switching.

Easy data conversion

You can configure a mapping rule to convert the calling request into the format required by the backend.

Presetting of request verification

You can preset the verification of the parameter type and values (range, enumeration, regular expression, and JSON Schema) for gateway to preclude the invalid requests, reduce the utilization rate of your backend.

Flexible throttling

You can set throttling for APIs, users, and APPs by minute, hour, or day.

In addition, you can also specialize some users or APPs with the independent throttling.

Easy security protection

API Gateway supports AppKey authentication and HMAC (SHA-1,SHA-256) signature.

API Gateway supports SSL/TSL encryption and uses Alibaba Cloud Security to prevent viruses and attacks.

Comprehensive monitoring and warning

API Gateway provides visualized API monitoring in real time, including the calling traffic, calling method, response time, and error rate, and supports query of historical records for comprehensive analysis. You can also configure and subscribe to the warning method (SMS or email) to check the API running status in real time.

Lower cost of publication

API Gateway automatically generates API documentation and SDKs (service end and mobile end), reducing the cost of publication of API.

Manage an API

API definitions refer to the definitions related to the API request structure when you create an API. You can view, edit, delete, create, or copy an API definition on the console. Pay attention to the following points when you are working with API definitions:

1. Editing the definition of a released API does not affect the definition in the production environment unless you release and synchronize it to the production environment.
2. It is not allowed to directly delete the API definition. Deprecate the API definition before deleting it.
3. You can copy the definition from the test/production environment to overwrite the latest definition, and then, if needed, click **Edit** to modify the definition.

API release management

You can release or deprecate an API in a test or production environment with the following attentions:

1. You can access the second-level domain name or independent domain name to call the API that is released to the test or production environment.
2. The latest released version of an API overwrites the preceding version in the test/production environment and takes effect in real time.
3. When you deprecate an API in the test/production environment, the binding policy, keys, app, and authorization persists are automatically deprecated unless the API is released to

production again. To revoke this relationship, you must delete it.

API authorization management

You can establish or revoke the authorization relationship between an API and an app. API Gateway verifies the permission relationship. During authorization, pay attention to the following points:

1. You can authorize one or more APIs to one or more apps. We recommend that you do not operate APIs in multiple groups at the same time during batch operation.
2. During batch operation, select an API and related environment. For example, if an API has been released to both the test and production environments, but only the test environment is chosen, only the API in the test environment is authorized.
3. You can locate an app based on the AppID or Alibaba Mail account provided by the customer.
4. When you need to revoke the authorization for an app under an API, you can view the API authorization list and delete the app from the list.

Release history and version switching

You can view the release history of each of you APIs, including the version number, notes, test/production, and time of each release.

When viewing the release history, you can select a version and switch to it. The new version directly overwrites the previous one and takes effect in real time.

Backend Signature

What Is a Signature Key

A signature key is the Key-Secret pair you create, based on which the backend service verifies the request received from the gateway. Pay attention to the following points:

1. An unchangable region must be selected during key creation. The key can only be bound to APIs in the same region.
2. One API can be bound with only one key. The key can be replaced, modified, bound to, or unbound from the API.
3. After binding a key to an API, the signature information is added to all the requests sent from the gateway to the API at your service backend. You must resolve the signature information through symmetric calculation at the backend to verify the gateway’ s identity. For more information about adding signature to the HTTP service, see Backend HTTP

Service Signature.

Modify or Replace the Leaked Key

To modify the Key-Secret pair once a key is leaked or to substitute a key bound to an API with another key, proceed the following steps:

1. Configure the backend to support two keys: the original key and to-be-modified or replaced key, so that the request during the switching process can pass signature verification regardless the key modification or replacement.
2. After the backend is configured, modify the key. Verify that the new Key and Secret take effect and delete the leaked or obsolete key.

Throttling

What is throttling policy

You can set throttling for APIs, users, and apps by minute, hour, or day, or you can sort out the specific users or apps with designated throttling policy. The throttling policy is described as follows:

Throttling policy contains the following dimensions:

API traffic limit	The call times within a unit time for the API bound by the policy must not exceed the set value. The time unit may be minute, hour, or day, for example, 5,000 times per minute.
App traffic limit	The call times called by each app within a unit time for an API bound to the policy must not exceed the set value, for example, 50,000 times per hour.
User traffic limit	The call times called by each Alibaba Cloud account within a unit time must not exceed the set value. An Alibaba Cloud account may have multiple apps. The traffic limit for an Alibaba Cloud account is exactly the limit on the total traffic of all apps in this account. For example, the traffic may be 500,000 times per day.

The three values can be set in one throttling policy. Note that the user traffic limit

must not exceed the API traffic limit, and the app traffic limit must not exceed the user traffic limit.

In addition, you can set an additional threshold value as the traffic limit value (not allowed to exceed the value of API traffic limit) for special apps or users. However, the basic app traffic limit and user traffic limit settings in the throttling policy are no longer applicable to the special apps or users.

An unchangable region must be selected for the throttling policy, and the throttling policy can only be applied to APIs in the same region.

The traffic of a single IP address is restricted within 100 QPS regarding with the value of API traffic limit.

A throttling policy can be bound to multiple APIs, with the limit value and special object settings appliable to each API separately. The lattest policy bound to the API overwrites the previous one and takes effect immediately.

To add a special app or user, you must obtain the app ID (AppID) or the Alibaba Mail account of the user.

On the API Gateway console, you can create, modify, delete, view, bind, and unbind a throttling policy.

Backend Signature Demo

Overview

API Gateway provides the backend HTTP service signature verification function. To enable backend signature, you must create a signature key and bind the key to the corresponding API. (keep this key properly. API Gateway encrypts and stores the key to guarantee the security of the key.) After backend signature is enabled, API Gateway adds signature information to the request destined to the backend HTTP service. The backend HTTP service reads the signature string of API Gateway and performs local signature calculation on the received request to check whether the gateway signature and local signature result are consistent.

All the parameters you have defined are added to the signature, including the service parameters you have entered, and constant system parameters and API Gateway system parameters (such as CaClientIp) you have defined.

How to read the API Gateway signature

- Save the signature calculated by the gateway in the header of the request. The Header name is X-Ca-Signature.

How to add a signature at the backend HTTP service

For more information about the demo (Java) of signature calculation, see https://github.com/aliyun/api-gateway-demo-sign-backend-java.

The signature calculation procedure is as follows:

Organize data involved in signature adding

```
String stringToSign=
HTTPMethod + "\n" + // All letters in the HTTPMethod must be capitalized.
Content-MD5 + "\n" + // Check whether Content-MD5 is empty. If yes, add a linefeed "\n".
Headers + // If Headers is empty, "\n" is not required. The specified Headers includes "\n". For more information, see the headers organization method described as follows.
Url
```

Calculate the signature

```
Mac hmacSha256 = Mac.getInstance("HmacSHA256");
byte[] keyBytes = secret.getBytes("UTF-8");
hmacSha256.init(new SecretKeySpec(keyBytes, 0, keyBytes.length, "HmacSHA256"));
String sign = new String(Base64.encodeBase64(Sha256.doFinal(stringToSign.getBytes("UTF-8")),"UTF-8"));
```

secret is the signature key bound to an API.

Description

Content-MD5

Content-MD5 indicates the MD5 value of the body. MD5 is calculated only when HTTPMethod is **PUT**

or **POST** and the body is not a form. The calculation method is as follows:

```
String content-MD5 = Base64.encodeBase64(MD5(bodyStream.getbytes("UTF-8")));
```

Headers

Headers indicates the keys and values of the headers involved in signature calculation. Read the keys of all headers involved in signature calculation from the header of the request. The key is X-Ca-Proxy-Signature-Headers. Multiple keys are separated by commas.

Headers organization method

Rank the keys of all headers involved in signature calculation in lexicographic order, and change all uppercase letters in the key of the header to lowercase, and splice the keys in the following method:

```
String headers =
HeaderKey1.toLowerCase() + ":" + HeaderValue1 + "\n"\+
HeaderKey2.toLowerCase() + ":" + HeaderValue2 + "\n"\+
...
HeaderKeyN.toLowerCase() + ":" + HeaderValueN + "\n"
```

URL

URL indicates the Form parameter in the Path + Query + Body. The organization method is as follows:If Query or Form is not empty, add a **?**, rank the keys of Query+Form in lexicographic order, and then splice them in the following method. If Query or Form is empty, then URL is equal to Path.

```
String url =
Path +
"?" +
Key1 + "=" + Value1 +
"&" + Key2 + "=" + Value2 +
...
"&" + KeyN + "=" + ValueN
```

Note that Query or Form may have multiple values. If multiple values exist, use the first value for signature calculation.

Debugging mode

To access and debug the backend signature conveniently, you can enable the Debug mode. The debugging procedure is as follows:

Add **X-Ca-Request-Mode = debug** to the **header** of the request destined to API Gateway.

The backend service can only read **X-Ca-Proxy-Signature-String-To-Sign** from the **header** because the linefeed is not allowed in the HTTP Header and thereby is replaced with "|".

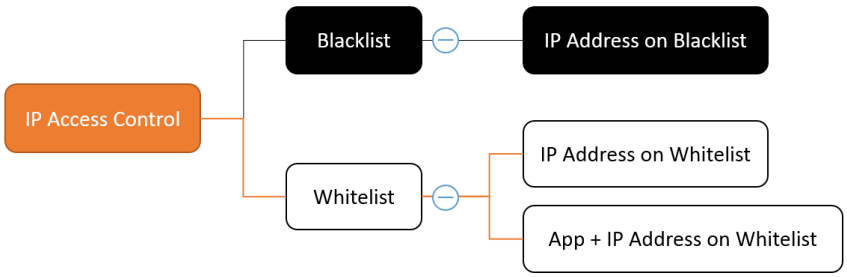
NOTE: **X-Ca-Proxy-Signature-String-To-Sign** is not involved in backend signature calculation.

Verify the time stamp

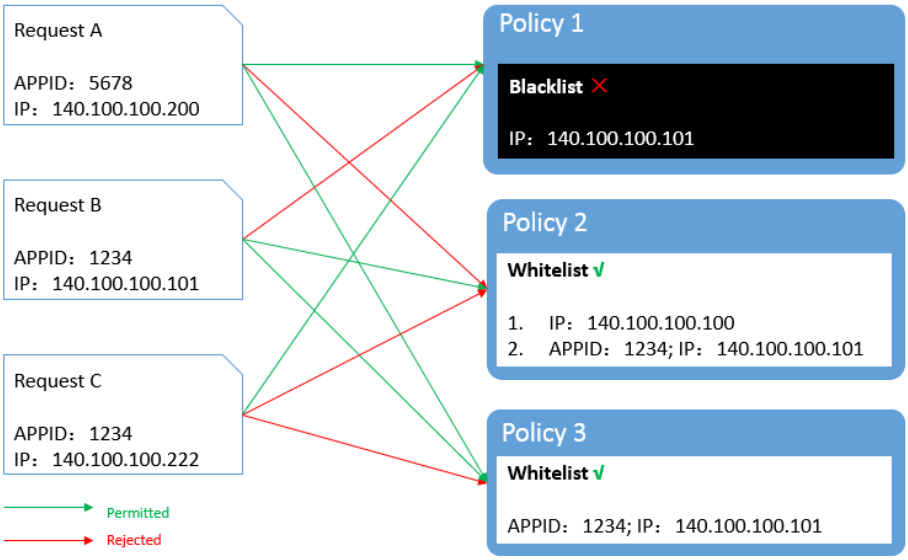
When the backend verifies the time stamp of the request, the system parameter **CaRequestHandleTime** is selectable in API definition and its value is the Greenwich mean time when the gateway receives the request.

IP access control

IP access control is one of the API security components provided by the API Gateway and controls the source IP addresses (or IP address segments) that can call APIs. You can add an IP address to the whitelist or blacklist of an API to permit or reject the API requests from this IP address.



- A whitelist can contain IP addresses or its combination with application IDs. Requests from IP addressed not listed on whitelist will be rejected.
    - For IP addresses, only IP addresses from specified source are allowed to visit.
    - For IP address and application ID combinations, application IDs can only visit from their combined IP addresses. Visits from other IP addresses will be rejected.
- Requests from IP addresses on the blacklist will be rejected by API Gateway.



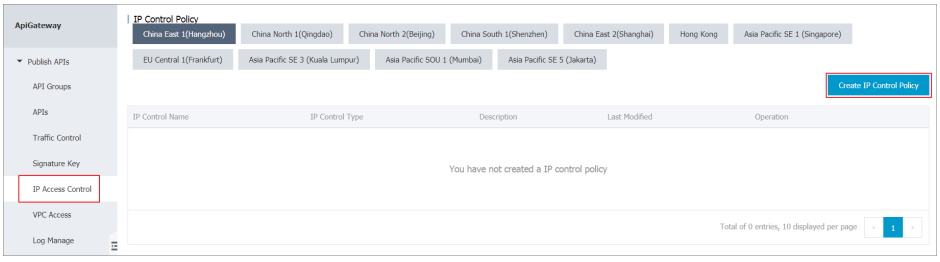
How to use this function

Add an IP access control policy

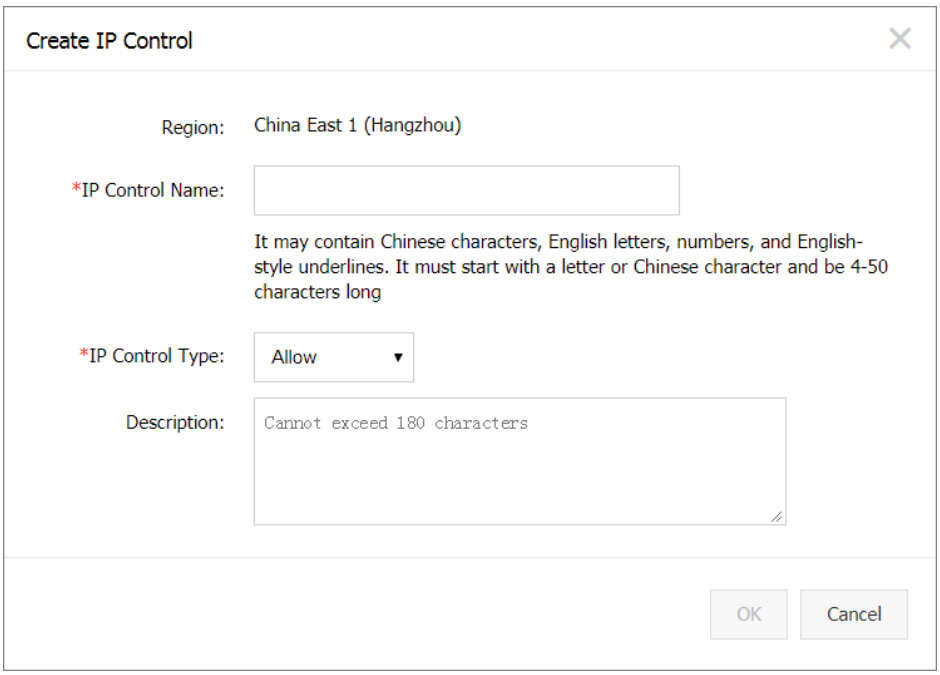
Create an IP access control policy and bind it to the API to which the access needs to be controlled.

Create an IP access control policy

Open API Gateway Console and choose “Publish APIs” > “IP Access Control”.



Click “Create IP Control Policy” to display the access control creation window.

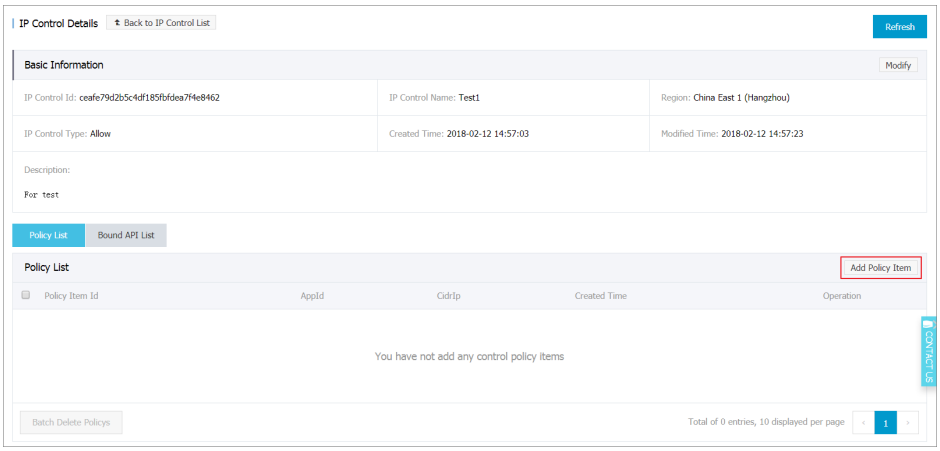


Enter the required information and click “OK”.

- If you set the access control type to Allow, you are configuring a whitelist.
- If you set the access control type to Refuse, you are configuring a blacklist.

Add a policy

After you create a whitelist or blacklist, you must enter the control policies corresponding to the list type. For a whitelist, you can enter the application ID, IP address, or combination of an application ID and an IP address. For a blacklist, enter an IP address.

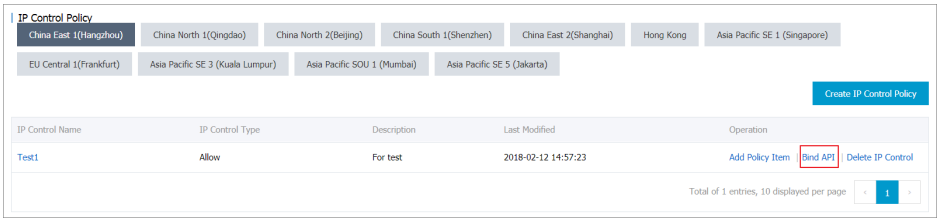


Click "OK" to complete the configuration.

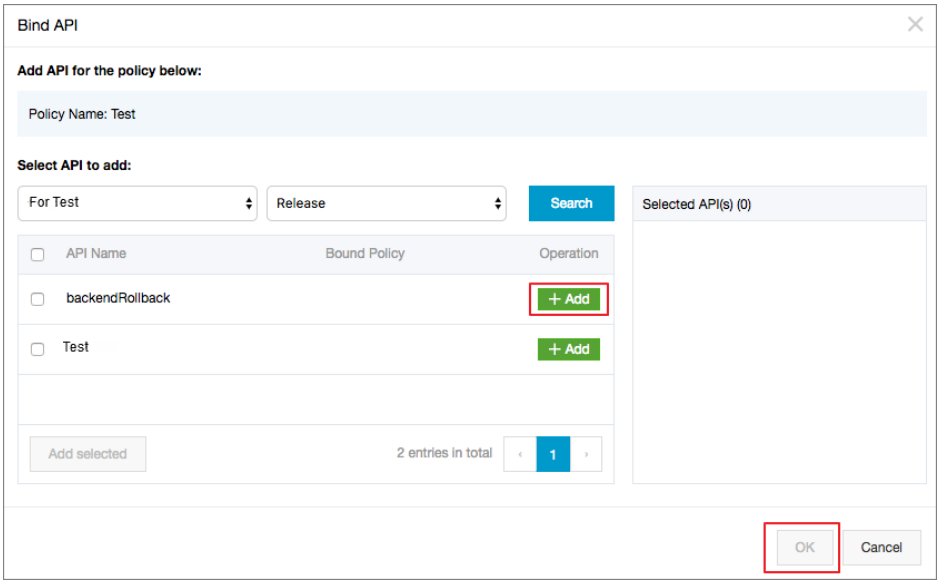
API binding

Bind the IP control policy to an API for the policy to take effect.

On the IP control policy list:



Find the required policy and bind API.



Select the corresponding API to bind the policy to it.

NOTE: Each API can have only one access control policy bound to it, no matter whether the policy is a blacklist or whitelist.

Delete an IP access control policy

Select a policy from the IP control policy list and delete it.

NOTE: If an IP control policy has been bound to an API, unbind it from the API before deleting it.

Check the bound API

You can find the API to which a policy is bound on the IP access control details page.

FAQ

When will the operation of binding or deleting an IP control policy take effect?

On the API Gateway, a policy binding operation takes effect immediately.

Can an API have different IP control policies bound in different environments?

Yes. You can bind different IP control policies to an API in different environments. We recommend that you bind a specified IP address to the test environment and pre-release environment to ensure security of the test environment.

Why is application blacklist not supported?

API calls require application authorization. To prohibit API calls for an application, you only need to delete its authorization. Therefore, application blacklist is not needed.