

# **API Gateway**

## **Getting Started**

### **Product Documentation**



## Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

## Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

## Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

# Contents

## Getting Started

- Step 1. Get the Access Permission
- Step 2. Create a Service
- Step 3. Create an API with Mock as the Backend Type
- Step 4. Publish the Service
- Step 5. Create a Plugin and Bind It to the API
- Step 6. Debug the API

# Getting Started

## Step 1. Get the Access Permission

Last updated : 2024-01-03 10:03:44

### Overview

To activate the API Gateway service, you need to use resources in other Tencent Cloud services. Therefore, you must authorize API Gateway to access your Tencent Cloud resources.

### Prerequisites

You have [registered a Tencent Cloud account](#) and completed [identity verification](#).

**Note:**

After you register a Tencent Cloud account, the system will create a root account for you by default, which is used to quickly access Tencent Cloud resources.

### Directions

1. When you log in to the [API Gateway console](#) with the **root account** for the first time, you need to click **Activate Now** on the console overview page to activate the API Gateway service.
2. Since API Gateway hasn't been granted a service role, it cannot access the resources of other Tencent Cloud services. You need to click **Authorize** to enter the [CAM console](#) for authorization.
3. Click **Agree** to authorize the API Gateway service to access your resources.

**Note:**

For more information on how to use API Gateway with a sub-account or collaborator, see [Permission Management](#).

# Step 2. Create a Service

Last updated : 2024-01-03 10:03:44

## Overview

Generally, certain features are implemented through a group of correlated APIs. The service management module of API Gateway helps you manage such APIs in an efficient and convenient way. This document describes how to create a service in the API Gateway console.

## Prerequisites

You have obtained the [access permission](#).

## Directions

1. Log in to the [API Gateway console](#).
2. On the left sidebar, click **Service** to enter the service list page.
3. Click **Create** and enter the service information.

### Create Service ✕

Region **Mumbai**

Service Name   
Up to 50 chars, supporting a-z, A-Z, 0-9, and underscores.

Frontend Type **HTTP&HTTPS** HTTP HTTPS

Billing Mode **Pay as you go**

Access Type  Private VPC  Public Network  
When a private VPC is selected, the generated private VPC domain name can be accessed in the VPC network in the same region of the service

Instance **Shared Type** Dedicated  
Featuring dedicated underlying resources, dedicated instances can provide ultra-high performance and SLA guarantee. [Buy Now](#)

Tag   ✕  
[+ Add](#)

Remarks

Fees  
Call fee: **0.00** and up (tiered pricing)  
Traffic fee: **0.00**

**Submit**

Service Name: it is required and can contain up to 50 letters, digits, and underscores. **exampleservice** is entered here as an example.

Instance Type: for the differences between the two instance types, see [Instance Specification](#). **Shared** is selected here as an example.

Remarks: remarks of the service. **Test** is entered here as an example.

Frontend Type: protocol type supported by the service. **HTTP&HTTPS** is selected here as an example.

Access Mode: when you select VPC, the generated VPC domain name can be accessed in VPCs in the same region as the service.

Tag: it is optional and makes it easier to categorize and manage resources.

4. Click **Submit** to create the service.

# Step 3. Create an API with Mock as the Backend Type

Last updated : 2024-01-03 10:03:44

## Overview

This document describes how to create an API connecting to Mock on the backend under a created service in the API Gateway console.

### Caution:

If the API connects to Mock in the backend, only fixed data can be returned. Therefore, we recommend you use Mock for testings but not in actual business scenarios.

## Prerequisites

You have [created a service](#).

## Directions

1. In the service list, click a service name to enter the API list page.
2. On the **General API** tab, select **Create** and enter the API's frontend configuration information.  
API Name: name of an API. **exampleapi** is entered here as an example.  
Frontend Type: HTTP and WebSocket are supported. **HTTP** is selected here as an example.  
Path: access path of the API. **/** is entered here as an example.  
Request Method: request method of the API. **GET** is selected here as an example.  
Authentication Type: authentication type of the API. **No authentication** is selected here as an example.  
CORS Support: whether the API supports cross-origin resource sharing. **Yes** is selected here as an example.  
Remarks: remarks of the API. **Test** is entered here as an example.  
Parameter Configuration: frontend parameters of the API. Nothing is entered here.



← **Create API**

1 **Frontend Configuration** > 2 **Backend Configuration** > 3 **Response Result**

Service: SCF\_API\_SERVICE

API Name:   
Up to 60 chars

Frontend Type:

Path:   
 1. Supports starting with "/" and "=". Starting with "/" means fuzzy match, while starting with "=" means exact match.  
 2. Supports uppercase and lowercase letters, numbers, and [-\_\*/~%]  
 3. The Path parameter must be wrapped with curly braces {} as a separate part of the path (such as /(param)/)  
 4. When the path starts with "=", adding request parameter of type Path is not supported.

Request Method:

Authentication Type:       
 An authentication-free mode under which APIs are accessible to all users, featuring a low security level. For more information, see [user guide for authentication-free mode](#)

CORS is supported:   
 1. When it's enabled, "access-control-allow-origin : \*" will be added to the response header by default.  
 2. To customize CORS configuration, please create a CORS plugin and bind it with the API. See [CORS Plugin Usage Guide](#)

Remarks:

Parameter Configurations

Parameter Name	Parameter Location ⓘ	Type	Default Value ⓘ	Required
<a href="#">New Parameters (0/30)</a>				

3. Click **Next** and enter the backend configuration information of the API.

Backend Type: type of the backend service of the API. **Mock** is selected here as an example.

Returned Data: data to be returned by Mock. **hello world, hello apigateway** is entered here as an example.

← Create API

1 Frontend Configuration > 2 Backend Configuration

Backend Type	<b>Public URL/IP</b> Provide backend services externally through the public network	<b>VPC resources</b> Connects to hosts and containers in VPC via upstreams and private CLB	<b>Serverless Cloud Function (SCF)</b> Serverless computing service provided by Tencent Cloud	<b>Mock</b> Simulate response data for testing	<b>EventBridge</b> API gateway set entry
Return data	Hello APIGateway!				

Previous Complete

4. Click **Complete**.

# Step 4. Publish the Service

Last updated : 2024-01-03 10:03:44

## Overview

A created service is in unpublished status by default and can be normally accessed only after it is published. You can publish it after configuring the service API.

## Directions

1. Find the service just created on the service list page, click **Publish** in the **Operation** column, and enter the service release information.

Environment: environment in which the service is published. The release environment is selected here.

Remarks: remarks of the service release. **Release for testing** is entered here as an example.

### Publish service ✕

Region	Guangzhou
Service Name	SCF_API_SERVICE
Release Environment	Publish
Remarks	<input type="text" value="Please enter remarks"/>

2. Click **Submit** to publish the service.

3. After the service is published, you can access the API at the sub-domain name provided by the service.

# Step 5. Create a Plugin and Bind It to the API

Last updated : 2024-01-03 10:03:44

## Overview

This document describes how to create a plugin and bind it to a created API in the API Gateway console.

## Prerequisites

You have [created an API](#).

## Directions

1. In the [API Gateway console](#), select **Plugin > System Plugin** on the left sidebar.
2. Click **Create** and enter the plugin information.

[←](#) **Create Plugin**

Region **Guangzhou**

Plugin Name

Up to 50 chars, supporting a-z, A-Z, 0-9, and underscores.

Type **IP access control** ▼

IP access control can restrict the source IPs of API callers to protect APIs. For more information, see [IP Access Control](#)

Plugin Description

Please enter description

Attribute \* **Allowlisted** **Blocklist**

IP \*

IP address or CIDR is supported. Multiple values are separated with semicolons.

**Save**

Plugin Name: it can contain up to 50 letters, digits, and underscores. **exampleplugin** is entered here as an example.

Type: select **IP access control**.

Plugin Description: description of the plugin. **Test** is entered here as an example.

Attribute: blocklist or allowlist. **Allowlist** is selected here as an example.

IP: enter the IP address or CIDR block that can access the API.

Tag: it is optional and makes it easier to categorize and manage resources.

3. Click **Save**.

4. On the plugin list page, click **Bind API** in the **Operation** column of the just created plugin.

### Bind API

Plugin Name

Service

Environment

**Select the API to be bound**

Please enter API name/API ID to filter

<input type="checkbox"/>	ID/Name	Path	Method
<input type="checkbox"/>	api-cjo3653i APIGWhtmlDemo-...	/APIGWhtmlDemo...	ANY

Support for holding shift key down for multiple selection

已选择(0)

ID/Name	Path	Method
No data yet		

Service: select the just created **exampleservice** service.

Select Environment: select **Release**.

Select APIs to bind: select the just created **exampleapi** API.

5. Click **OK**.

# Step 6. Debug the API

Last updated : 2024-01-03 10:03:44

## Overview

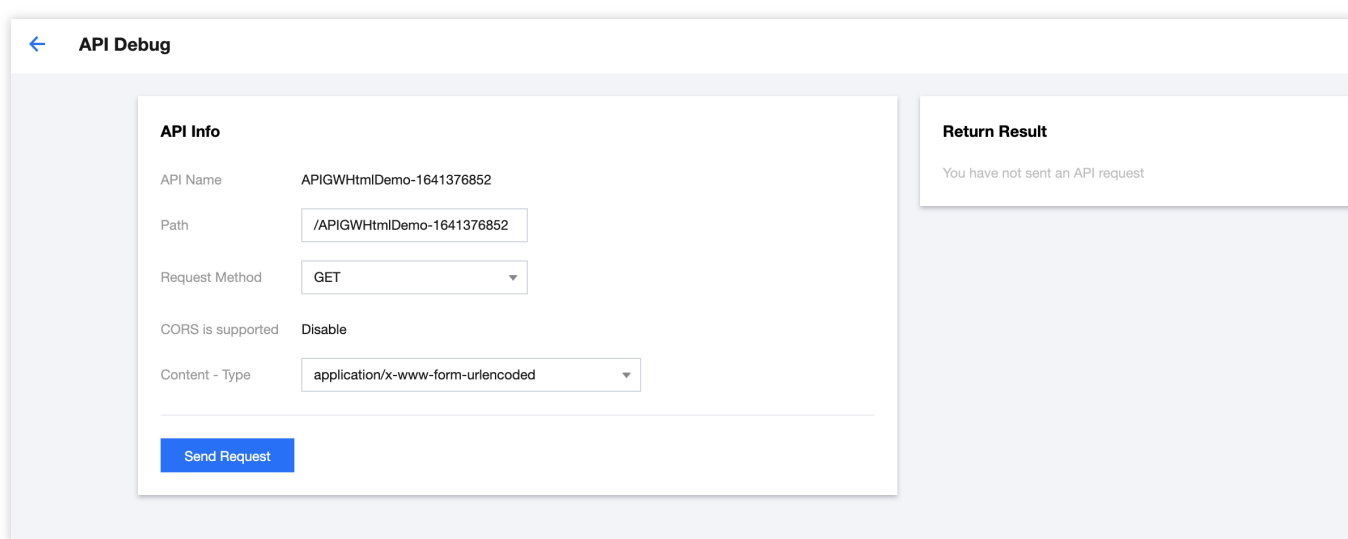
The API debugging page allows you to verify the correctness of an API immediately after completing its configuration by initiating a simulated API call and viewing the specific request response. If the API fails to work as expected, you can modify the configuration according to the response to make it meet your design expectations.

## Prerequisites

You have [created an API](#) .

## Directions

1. Find the API just created in [step 3](#) on the API list page and click **Debug** in the **Operation** column to enter the API debugging page.
2. Select `application/x-www-form-urlencoded` as the `Content-Type` .
3. Click **Send Request** and view the result returned after debugging.



The screenshot shows the 'API Debug' interface. On the left, there is a form titled 'API Info' with the following fields:

- API Name: APIGWHtmlDemo-1641376852
- Path: /APIGWHtmlDemo-1641376852
- Request Method: GET
- CORS is supported: Disable
- Content - Type: application/x-www-form-urlencoded

At the bottom of the form is a blue button labeled 'Send Request'. On the right side of the interface, there is a 'Return Result' section with the text: 'You have not sent an API request'.