

**API Gateway**

# Getting Started

**Issue**            02  
**Date**             2024-05-23



**Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2024. All rights reserved.**

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

## **Trademarks and Permissions**



HUAWEI and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

## **Notice**

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

---

# Contents

---

<b>1 Introduction.....</b>	<b>1</b>
<b>2 Opening APIs.....</b>	<b>2</b>
2.1 Process Flow.....	2
2.2 Creating an API Group.....	3
2.3 Binding a Domain Name.....	4
2.4 Creating an API.....	4
2.5 Debugging an API.....	7
2.6 (Optional) Creating an Environment.....	7
2.7 Publishing an API.....	8
<b>3 Calling APIs.....</b>	<b>10</b>
3.1 Process Flow.....	10
3.2 Creating a Credential and Getting Authorized.....	11
3.3 Adding an AppCode for Simple Authentication.....	11
3.4 Calling an API.....	12
<b>4 Getting Started with Common Practices.....</b>	<b>13</b>

# 1 Introduction

---

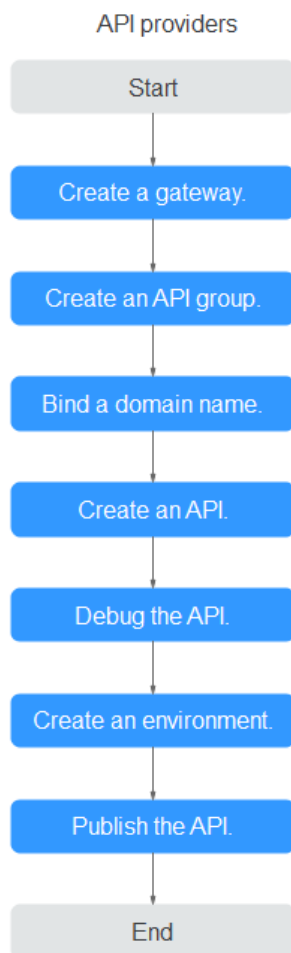
API Gateway (APIG) is a fully managed service that enables you to securely build, manage, and deploy APIs at any scale with high performance and availability. With APIG, you can easily integrate your internal service systems and selectively expose your service capabilities.

To learn about the process of exposing and calling an API, see [Opening APIs](#) and [Calling APIs](#). **Simple authentication** with an app is used for illustration.

# 2 Opening APIs

## 2.1 Process Flow

The following figure shows the process of exposing an API.



1. Creating a Gateway  
**Buy a dedicated gateway.**

2. **Creating an API Group**  
An API group facilitates management of APIs used for the same service. Create an API group and then create APIs.
3. **Binding a Domain Name**  
Before making the API available for users to access, bind an independent domain name (custom domain name) to the group to which the API belongs. Then API callers can use these domain names to call the API.
4. **Creating an API**  
When creating an API, configure the frontend and backend request paths, parameters, and protocols.
5. **Debugging an API**  
Debug the API to check whether it works normally.
6. **(Optional) Creating an Environment**  
An API can be called in different scenarios, such as the production environment (RELEASE) or other custom environments. RELEASE is the default environment defined in APIG.
7. **Publishing an API**  
Publish the API so that it can be called.

## 2.2 Creating an API Group

- Step 1** Go to the [APIG console](#).
- Step 2** Select [a dedicated gateway you purchased](#).
- Step 3** In the navigation pane, choose **API Management > API Groups**.
- Step 4** Choose **Create API Group > Create Directly**.

**Figure 2-1** Configuring API group information

**Create API Group**

\* Name

Enter 3 to 255 characters, starting with a letter or digit. Only letters, digits, and the following special characters are allowed: -\_/:()

Description

0/1,000

**Table 2-1** API group information

Parameter	Description
Name	API group name. It is recommended that you enter a name based on naming rules to facilitate search.
Description	Description of the API group.

**Step 5** Click **OK**. The system automatically allocates a debugging domain name to the API group. APIs in the group can be debugged using the domain name.

----End

## 2.3 Binding a Domain Name

**Step 1** On the **API Groups** page, click the group created in [Creating an API Group](#) to go to the group details page.

**Step 2** Click the **Group Information** tab.

**Step 3** Click **Bind Independent Domain Name** in the **Independent Domain Names** area.

**Figure 2-2** Binding an independent domain name

**Bind Independent Domain Name** ×

\* Domain Name

\* Minimum TLS Version  TLS1.1  TLS1.2

HTTP-to-HTTPS Auto Redirection

**OK**

### NOTE

The independent domain name must be registered and resolved. For details, see "Prerequisites" in [Binding a Domain Name](#).

----End

## 2.4 Creating an API

Procedure:

1. [Configuring Frontend Settings](#)

## 2. Configuring Backend Settings

### Configuring Frontend Settings

**Step 1** In the navigation pane, choose **API Management > APIs**.

**Step 2** Click **Create API** and configure the frontend definition.

**Frontend Definition**

\* API Name

\* Group  [Create API Group](#)  
APIs in the group: 0, Available for creation: 1000

\* URL

Method	Protocol	Subdomain Name	Path
<input type="text" value="POST"/>	<input type="text" value="HTTPS"/>	<input type="text" value=""/>	<input type="text" value="/v2/testabc"/>

\* Gateway Response

Matching  Exact match  Prefix match  
API requests will be forwarded to the specified path.

Tags

Description

0/255

Content Format Type

**Table 2-2** Frontend definition

Parameter	Description
Name	API name. It is recommended that you enter a name based on naming rules to facilitate search.
API Group	By default, the group created in <a href="#">Creating an API Group</a> is selected.
URL	<b>Method:</b> Request method of the API. Set this parameter to <b>POST</b> . <b>Protocol:</b> Set this parameter to <b>HTTPS</b> . Subdomain name: The subdomain automatically allocated to the API group created in <a href="#">Creating an API Group</a> . <b>Path:</b> Path for requesting the API.
Gateway Response	Select a response to be displayed if API Gateway fails to process an API request. The default gateway response is <b>default</b> .
Matching	By default, <b>Exact match</b> is selected.
Tags	Classification attribute used to quickly identify the API from other APIs.
Description	Description of the API.



**Step 3** Configure security settings based on the following table.

**Table 2-3** API request definition

Parameter	Description
Authentication Mode	API authentication mode. Set this parameter to <b>App</b> .
Simple Authentication	If you enable this option, API Gateway verifies only the AppCode and the request signature does not need to be verified. For this example, enable simple authentication.

**Step 4** Click **Next**.

----End

## Configuring Backend Settings

**Step 1** On the **Backend Configuration** page, set the backend service information.

**Step 2** Select a backend service type. For this example, select **HTTP&HTTPS**.

The screenshot shows the 'Backend Configuration' interface. At the top, 'Backend Type' is set to 'HTTP&HTTPS'. Below this, there are tabs for 'Default Backend' and 'Backend Policies'. The 'Basic Information' section is active, showing a 'Load Balance Channel' dropdown set to 'Skip'. Below that, there are input fields for 'URL' (with sub-fields for Method: POST, Protocol: HTTP, Backend Address, and Path: /v2/testabc), 'Timeout (ms)' (5000), and 'Retry Times' (-1). At the bottom, there is a checkbox for 'Backend Authentication' which is unchecked.

**Table 2-4** HTTP&HTTPS backend service definition

Parameter	Description
Load Balance Channel	Determine whether the backend service will be accessed using a load balance channel. For this example, select <b>Skip</b> .
URL	<b>Method:</b> Request method of the API. Set this parameter to <b>POST</b> . <b>Protocol:</b> Set this parameter to <b>HTTP</b> . <b>Backend Address:</b> Address of the backend service. <b>Path:</b> Path of the backend service.
Timeout	Backend service request timeout. Default value: 5000 ms.

**Step 3** On the **Define Response** page, set the responses.

**Define Response**

Example Success Response

pass

4/20,480

Example Failure Response

fail

4/20,480

**Table 2-5** Defining responses

Parameter	Description
Example Success Response	An example of a response returned when the API is called successfully.
Example Failure Response	An example of a response returned when the API fails to be called.

**Step 4** Click **Finish**.

----End

## 2.5 Debugging an API

**Step 1** On the **APIs** tab page, select an API from **Creating an API** and click **Debug**.

**Step 2** Configure the URL.

**Step 3** Click **Debug**. The API request and response information are displayed at the bottom of the page.

If the API is called successfully, the status code **200** is displayed.

----End

## 2.6 (Optional) Creating an Environment

**Step 1** In the navigation pane, choose **API Management > API Policies**. Then click the **Environments** tab.

**Step 2** Click **Create Environment** and set the environment information.

### Create Environment

\* Name

Enter 3 to 64 characters, starting with a letter. Only letters, digits, and underscores ( \_ ) are allowed.

Description

0/255

**Table 2-6** Environment information

Parameter	Description
Name	Environment name. It is recommended that you enter a name based on naming rules to facilitate search.
Description	Description of the environment.

**Step 3** Click **OK**.

----End

## 2.7 Publishing an API

**Step 1** In the navigation pane, choose **API Management > APIs**.

**Step 2** Locate the API created in [Creating an API](#), and click **Publish**.

**Step 3** Select the environment where the API will be published.

API Name

Environment

If you publish the API, any existing configuration of the same API in the selected environment will be overwritten.

Description

0/255

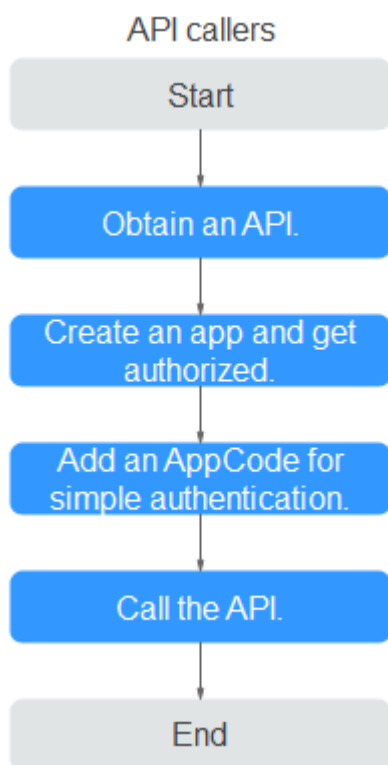
**Step 4** Click **OK**.

----End

# 3 Calling APIs

## 3.1 Process Flow

The following figure shows the process of calling an API.



1. Obtaining an API  
Obtain an API and its documentation from an API provider.
2. **Creating a Credential and Getting Authorized**  
APIs that use app authentication can only be called using credentials bound to them.
3. **Adding an AppCode for Simple Authentication**

API Gateway only verifies the AppCode during simple authentication.

4. **Calling an API**

Use an API test tool to call the API with the simple authentication AppCode.

## 3.2 Creating a Credential and Getting Authorized

### Creating a Credential

**Step 1** In the navigation pane, choose **API Management > Credentials**.

**Step 2** Click **Create Credential** and set credential information.

**Table 3-1** Credential information

Parameter	Description
Name	Credential name. It is recommended that you enter a name based on naming rules to facilitate search.
Description	Description about the credential.

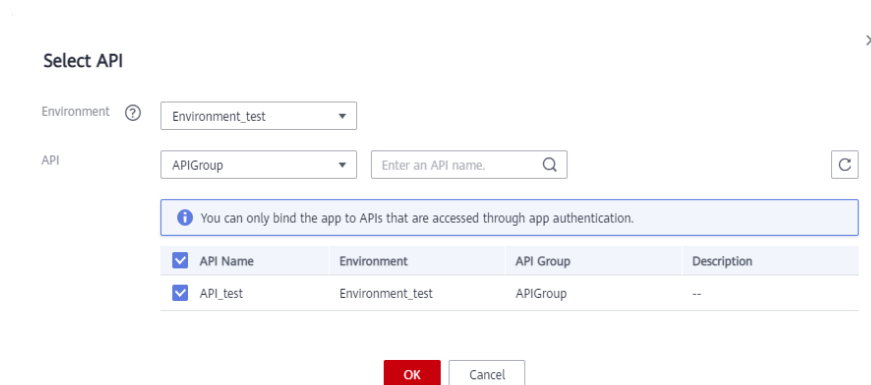
**Step 3** Click **OK**.

----End

### Binding to an API

**Step 1** Locate the target row in **Credentials** and click **Bind to APIs**.

**Step 2** Select the environment, API group, and API created in **Opening APIs**, and click **OK**.



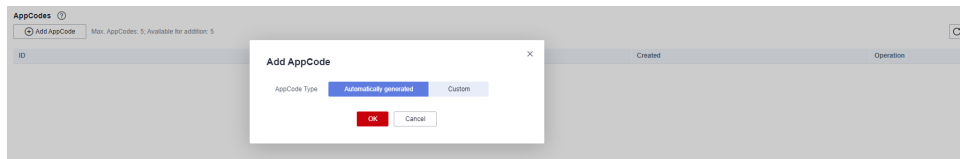
----End

## 3.3 Adding an AppCode for Simple Authentication

**Step 1** In the credential list, click the credential created in **Creating a Credential** to go to the credential details page.

**Step 2** Click **Add AppCode** in the **AppCodes** area.

**Step 3** Select **Automatically generated**.



**Step 4** Click **OK**.

----End

## 3.4 Calling an API

Use the API test tool to configure the API request and authentication. For details about how to call an API, see [Calling APIs](#).

**Step 1** Obtain the API request information and construct the API URL.

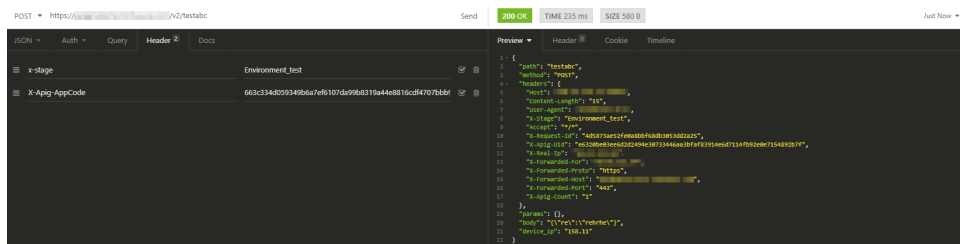
For illustration purposes, an API and its documentation are obtained through offline channels. You can also obtain the authentication mode, request method, request path, and other information about the API.

**Step 2** Add the header parameter **X-Apig-AppCode** and set the parameter value to the **generated AppCode**.

**Step 3** Add the header parameter **x-stage** and set the parameter value to the **running environment**. Skip this step if the API has been published in the RELEASE environment.

**Step 4** Click **Send** to send a request.

If the API is called successfully, the message **200 OK** is displayed.



----End

# 4 Getting Started with Common Practices

---

You can use the common practices provided by APIG to meet your service requirements.

**Table 4-1** Common practices

Practice	Description
<a href="#">Developing a Custom Authorizer with FunctionGraph</a>	<p>In addition to IAM and app authentication, APIG also supports custom authentication with your own authentication system, which can better adapt to your business capabilities.</p> <p>Custom authentication is implemented using the FunctionGraph service. You can create a FunctionGraph function so that APIG can invoke it to authenticate requests for your API.</p>



Practice	Description
<b>Request Throttling 2.0</b>	<p>If the number of requests initiated from public networks for open APIs on APIG is not limited, the continuous increase in users will deteriorate the backend performance. And what's worse, the website or program will break down due to a large number of requests sent by malicious users. The traditional request throttling policies of APIG throttle requests by API, user, credential, and source IP address.</p> <p>However, as users and their demands become more diversified, these traditional policies cannot meet the requirements for more refined rate limiting. To resolve this issue, APIG has launched request throttling 2.0, which is a type of plug-in policy. The 2.0 policies enable you to configure more refined throttling, for example, to throttle requests based on a certain request parameter or tenant.</p>