API Gateway

Service Overview

Issue	01
Date	2025-03-05

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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.

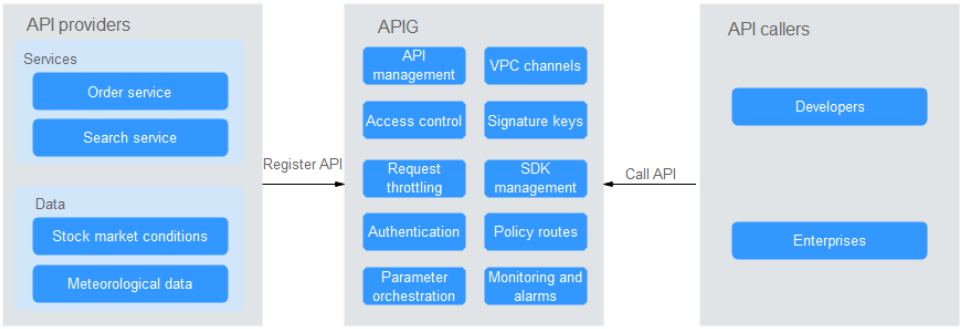
Contents

1 What Is APIG?

API Gateway (APIG) is your fully managed API hosting service. With APIG, you can build, manage, and deploy APIs at any scale to package your capabilities. With just a few clicks, you can integrate internal systems, monetize service capabilities, and selectively expose capabilities with minimal costs and risks.

- To monetize your service and data, you can open them up by creating APIs in APIG. Then you can provide the APIs for API callers using offline channels.
- You can also obtain open APIs from APIG to reduce your development time and costs.

Figure 1-1 APIG architecture



Product Functions

- **API lifecycle management**

  The lifecycle of an API involves creating, publishing, removing, and deleting the API. API lifecycle management enables you to quickly and efficiently expose service capabilities.

- **Built-in debugging tool**

  With the built-in debugging tool, you can debug APIs using different HTTP headers and request bodies. This tool simplifies the API development process and reduces the API development and maintenance costs.

- **Version management**

  An API can be published in different environments. Publishing an API again in the same environment will override the API's previous version. APIG displays the publication history (including the version, description, date and time, and

environment) of each API. You can roll back an API to any historical version to meet dark launch and version upgrade requirements.

- **Environment variables**

  Environment variables are manageable and specific to environments. Variables of an API will be replaced by the values of the variables in the environment where the API will be published. You can create variables in different environments to call different backend services using the same API.

- **Refined request throttling**
  - For different service demands and user levels, you can control the frequency at which an API can be called by a user, app (credential), or IP address, ensuring that backend services can run stably.
  - The throttling can be accurate to the second, minute, hour, or day.
  - Set throttling limits for excluded applications (credentials) and tenants.

- **Monitoring and alarms**

  APIG provides visualized, real-time API monitoring, and displays multiple metrics, including number of requests, invocation latency, and number of errors. The metrics help you understand the API usage, allowing you to identify potential service risks.

- **Security**
  - Domain name access can be authenticated with TLS 1.1 and TLS 1.2.
  - Access control policies limit API access from specific IP addresses or accounts. You can blacklist or whitelist certain IP addresses and accounts to access your APIs.
  - Identity authentication can be based on AK/SK, function-based custom authorizers, and tokens. APIG verifies your backend services via certificates and is verified by your backend services through signature keys.

- **VPC channels (load balance channels)**

  Virtual Private Cloud (VPC) channels (load balance channels) can be created for accessing resources in VPCs and exposing backend services deployed in VPCs. VPC channels balance API requests to backend services.

- **Mock response**

  Mock backends simulate API responses for circuit breakers, service degradation, and redirection.

- **HTTP2.0**

  APIG supports HTTP/2, which is a major revision of HTTP and was originally named HTTP 2.0. It provides binary encoding, request multiplexing over a single connection, and request header compression, improving transmission performance and throughput with a lower latency.

  NOTE

  - HTTP 2.0 strongly depends on network stability. To use HTTP 2.0, ensure that your network is stable and your client supports this protocol.
  - If your gateway does not support HTTP 2.0, contact technical support to upgrade it.
  - To disable HTTP 2.0, turn off **HTTP/2** under the **request_custom_config** parameter on the **Parameters** tab page of the APIG console.

- Binary encoding

  Unlike HTTP 1.x where data is transmitted in text format, data in HTTP 2.0 is split into messages and frames for binary encoding. Compared with string (text) parsing, binary parsing is easier and less error-prone and delivers higher transmission performance.

- Multiplexing

  With binary encoding, HTTP 2.0 no longer relies on multiple connections to process and send requests and responses concurrently.

  For the same domain name, all requests are completed on a single connection, and each connection can process any number of messages. A message consists of one or more frames, which can be sent out of order and finally recombined based on the stream ID in the header of each frame. This shortens the latency and improves the efficiency.

- Header compression

  HTTP 2.0 uses an encoder to reduce the size of the headers to transmit. Both the client and server store a header field table to avoid transmitting same headers repeatedly, achieving high throughput.

2 Product Advantages

Available Out-of-the-Box

You can quickly create APIs by configuring the required settings on the APIG console. APIG provides an inline debugging tool to simplify API development, and allows you to publish an API in multiple environments for easy testing and fast iteration.

Convenient API Lifecycle Management

APIG provides full-lifecycle API management, including design, development, test, publish, and O&M, to help you quickly build, manage, and deploy APIs at any scale.

Refined Request Throttling

APIG combines synchronous and asynchronous traffic control and multiple algorithms to throttle requests at the second level. You can flexibly define request throttling policies to ensure stability and continuity of API services.

Visualized API Monitoring

APIG monitors the number of API calls, data latency, and number of errors, helping you identify potential service risks.

Comprehensive Security Protection

APIG provides multiple measures to secure API calling, such as Secure Sockets Layer (SSL) transfer, strict access control, IP address blacklist/whitelist, authentication, anti-replay, anti-attack, and multiple audit rules. In addition, APIG implements flexible and refined quota management and request throttling to help you flexibly and securely open your backend services.

Flexible Policy Routes

You can configure backends for an API to forward requests according to multiple policies. This facilitates dark launch and environment management.

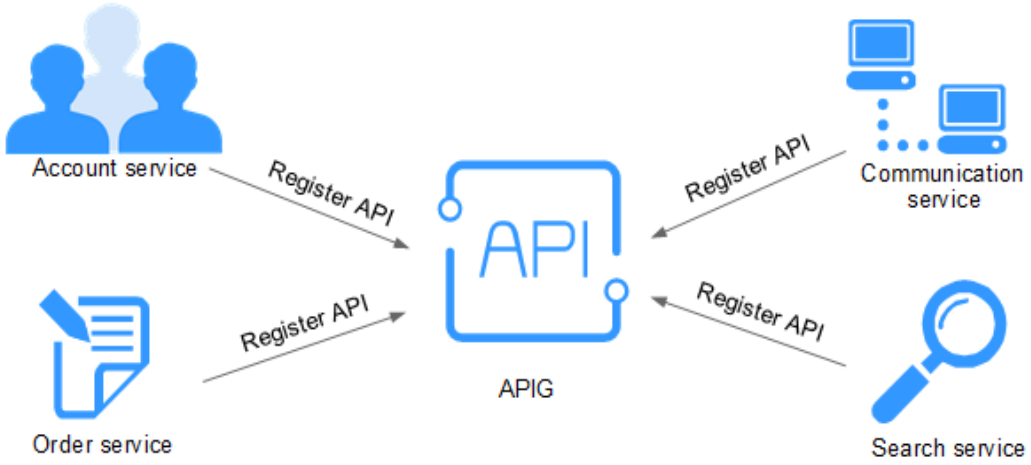
SDKs of Different Programming Languages

SDKs of different programming languages (such as Java, Go, Python, and C) are available for access from clients. Because the backends do not need to be modified, only one system is required to adapt to different service scenarios (such as mobile devices and IoT).

3 Application Scenarios

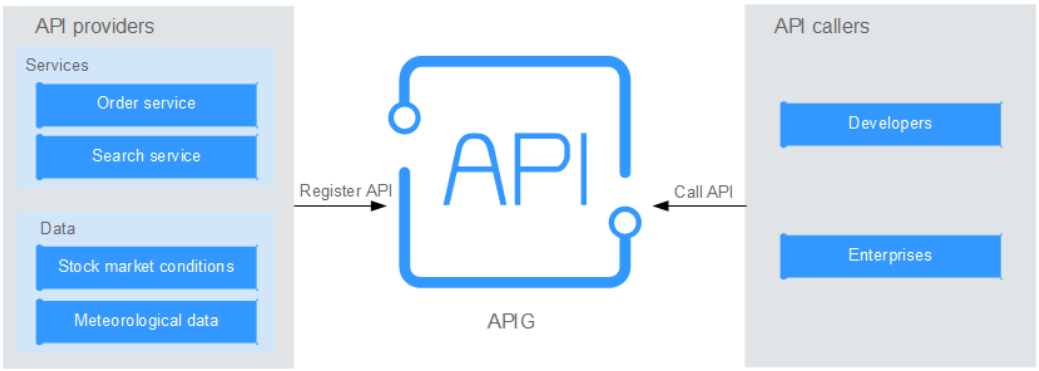
Internal System Decoupling

As enterprises develop rapidly with quick business changes, internal systems of enterprises need to keep pace with the development. However, it is difficult to ensure system universality and stability because internal systems are dependent on each other. APIG uses standard RESTful APIs to simplify the service architecture, decouples internal systems, and separates the frontend from backend. Existing capabilities can be reused to avoid repetitive development.



Enterprise Capabilities Opening

An enterprise cannot develop without partners' capabilities, such as a third-party payment platform and partner account login. APIG enables you to selectively expose capabilities to partners by using standard APIs and share services and data with partners to build a new ecosystem.



4 Specifications

Dedicated Gateway Specifications

The query per second (QPS) throughput of a dedicated gateway is affected by multiple factors, such as the response size, whether HTTPS access is enabled, and whether gzip compression is enabled. The following table lists the APIG QPS reference values at 30% CPU usage in non-authentication and single node scenarios.

The **security watermark** enables APIG to maintain high throughput and low latency even when the burst traffic doubles.

Table 4-1 QPS Reference

Edition				Basic	Professional	Enterprise	Platinum	Platinum 2
Connection Type	Number of Response Bytes (KB)	Whether to Use HTTPS	Whether to Use gzip	QPS Reference at 30% CPU Usage				
Non-persistent connection	1	No	No	1,600	3,600	9,000	55,000	72,000
		Yes	No	1,000	1,100	2,800	16,000	20,000
Persistent connection	1	No	No	2,500	4,200	13,000	79,000	105,000
		Yes	No	2,000	4,000	11,000	67,000	95,000
	10	No	No	2,200	4,000	10,000	67,000	85,000
		Yes	No	1,800	3,800	9,500	65,000	80,000

The **bandwidth** and **private network connections** vary depending on the gateway edition. Refer to the following table for optimal settings.

Table 4-2 Bandwidth and connections

Edition	Bandwidth	Private Network Connections per Second
Basic	Single-AZ: 50 Mbit/s Dual-AZ or more: 100 Mbit/s	1,000
Professional	Single-AZ: 100 Mbit/s Dual-AZ or more: 200 Mbit/s	1,000
Enterprise	Single-AZ: 200 Mbit/s Dual-AZ or more: 400 Mbit/s	1,000
Platinum	Single-AZ: 400 Mbit/s Dual-AZ or more: 800 Mbit/s	1,000

NOTE

- The specifications of dedicated gateways cannot be modified.

5 Notes and Constraints

Gateway

Table 5-1 Gateway notes and constraints

Item	Restrictions
Permissions	• You must be assigned both the **APIG Administrator** and **VPC Administrator** roles so that you can create gateways. • Alternatively, you must be attached the **APIG FullAccess** policy. • For details about how to use custom policies, see **APIG Custom Policies**.
Network	• If you use **192.x.x.x** or **10.x.x.x**, APIG uses **172.31.32.0/19** as the internal subnet. • If you use **172.x.x.x**, APIG uses **192.168.32.0/19** as the internal subnet.
Number of available private IP addresses in the subnet	The basic, professional, enterprise, and platinum editions of APIG require 3, 5, 6, and 7 private IP addresses. Check that the subnet you choose has sufficient private IP addresses on the VPC console.
Load	• VPCs (workloads) where gateways have been deployed cannot be changed.

Item	Restrictions
Specifications	• During the specification change, the persistent connection is intermittently disconnected and needs to be re-established. You are advised to change the specification during off-peak hours. • Specifications can be upgraded but cannot be downgraded. • Changing the gateway edition will also change the private network access IP addresses. Modify your firewall or whitelist configuration if necessary for service continuity. Do not perform any other operations on the gateway. After the change is complete, adjust the firewall or whitelist configuration based on service requirements.

API

Table 5-2 API notes and constraints

Item	Restrictions
API group	Each API can belong to only one group.
SSL Certificate	• Only SSL certificates in PEM format are supported. • SSL certificates support only the RSA and ECDSA encryption algorithms.

Item	Restrictions
Domain name	• By default, the debugging domain name of an API group can only be resolved to a server in the same VPC as the gateway. If you want to resolve the domain name to a public network, bind an EIP to the gateway. • The debugging domain name cannot be used for production services and can be used only for application debugging. • Groups under the same gateway cannot be bound with a same independent domain name. • If a domain name is already bound to a port, it cannot be bound to the same port again. • If different ports are used for the same domain name, all ports take effect no matter whether any of them are bound to, modified, or unbound from the SSL certificate or whether client authentication is enabled or disabled. • If you access backend services through a load balance channel, the port bound to the independent domain name must be the same as the access port of the backend server in the load balance channel. • After an independent domain name is bound to a port, if you use an IP address to access an API in a custom group, you need to add the header parameter **host** to the request. The **host** value should include the port number for access, unless you are using the default ports **80** or **443**, in which case the **host** value is not necessary. • Accessing APIs by IP address is not advised, it requires IP certificates for SSL. Otherwise, the connection may be insecure. • HTTP-to-HTTPS redirection is only suitable for GET and HEAD requests. Redirecting other requests may cause data loss due to browser restrictions. Redirection takes effect only when the API request protocol is **HTTPS** or **HTTP&HTTPS** and an SSL certificate has been bound to the independent domain name.

Item	Restrictions
API policies	• An API can be bound with only one policy of the same type (request throttling, proxy cache, or third-party authorizer) for a given environment, but each policy can be bound to multiple APIs. • Policies are independent of APIs. A policy takes effect for an API only after they are bound to each other. When binding a policy to an API, you must specify an environment where the API has been published. The policy takes effect for the API only in the specified environment. • After you bind a policy to an API, unbind the policy from the API, or update the policy, you do not need to publish the API again. • Taking an API offline does not affect the policies bound to it. The policies are still bound to the API if the API is published again. • Policies that have been bound to APIs cannot be deleted.
Credential	• A credential can be bound to a maximum of 1,000 APIs. • You can create a maximum of five AppCodes for each credential.

Quota Limits

To change the default restrictions, contact technical support to increase the quota. For details about parameter configuration of a dedicated gateway, see **Modifying Configuration Parameters**.

NOTICE

- It takes 5 to 10 seconds for a new or modified APIG resource to take effect.
- The maximum quota may be slightly exceeded in case of high concurrency, but resource usage will not be affected.

Table 5-3 Dedicated API gateway quotas

Item	Default Restriction	Modifiable
Gateways	5	√
API groups	1500	√

Item	Default Restriction	Modifiable
APIs	Number of APIs for each gateway edition: • Basic: 250 • Professional: 800 • Enterprise: 2000 • Platinum: 8000	√
APIs	1000 for each group	x
Backend policies	5	√
Credentials	50. The credential quota includes the apps you have created.	√
Request throttling policies	• You can create a maximum of 300 request throttling policies for each gateway. • The call limit for a single user cannot exceed that for the target API. • The call limit for a single app (credential) cannot exceed that for a single user. • The call limit for a single IP address cannot exceed that for the target API.	√
Environments	10	√
Signature keys	200	√
Access control policies	100	√
VPC channels (load balance channels)	200	√
Variables	You can create a maximum of 50 variables for an API group in each environment.	√
Independent domain names	A maximum of five independent domain names can be bound to an API group.	√
ECSs	A maximum of 10 ECSs can be added to a VPC channel.	√
Parameters	A maximum of 50 parameters can be created for an API.	√

Item	Default Restriction	Modifiable
API publication records	A maximum of 10 publication records of an API can be retained for each environment.	√
API access rate	Up to 6000 times per second	√
Excluded applications (Credentials)	A maximum of 30 excluded apps can be added to a request throttling policy.	√
Excluded tenants	A maximum of 30 excluded tenants can be added to a request throttling policy.	√
Access to a subdomain name (debugging domain name)	A subdomain name can be accessed up to 1000 times a day.	x
Maximum size of an API request package	12 MB	√
TLS protocol	TLS 1.1 and TLS 1.2 are supported. TLS 1.2 is recommended.	√
Custom authorizers	50	x
Plug-ins	500	√
HTTP protocol	When the HTTP protocol is used, the maximum size of URL+Header is 32 KB.	x

6 Permissions Management

If you need to assign different permissions to personnel in your enterprise to access your APIG resources, Identity and Access Management (IAM) is a good choice for fine-grained permissions management. IAM provides identity authentication, permissions management, and access control, helping you securely access your resources.

With IAM, you can use your account to create IAM users for your employees, and assign permissions to the employees to control their access to specific resources.

If your account does not require individual IAM users for permissions management, skip this chapter.

APIG Permissions

By default, new IAM users do not have any permissions assigned. You need to add a user to one or more groups, and attach policies or roles to these groups. The user then inherits permissions from the groups to which the user belongs, and can perform specified operations on cloud services based on the permissions.

APIG is a project-level service deployed and accessed in specific physical regions. To assign APIG permissions to a user group, you need to specify region-specific projects for which the permissions will take effect. If you select **All projects**, the permissions will be granted for both the global service project and all region-specific projects. When accessing APIG, the users need to switch to a region where they have been authorized to use this service.

You can grant permissions by using roles and policies.

- Roles: A type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. This mechanism provides only a limited number of service-level roles for authorization. When using roles to grant permissions, you need to also assign other dependent roles for permissions to take effect. However, roles are not an ideal choice for fine-grained authorization and secure access control.
- Policies: A fine-grained authorization strategy that defines permissions required to perform operations on specific cloud resources under certain conditions. This mechanism allows for more flexible policy-based authorization and meets requirements for secure access control. For example, you can grant APIG users only the permissions for performing specific

operations. Most policies define permissions based on APIs. For the API actions supported by APIG, see **Permissions Policies and Supported Actions**

Table 6-1 lists all the system-defined roles and policies supported by APIG.

Table 6-1 System-defined roles and policies supported by APIG

Role/ Policy Name	Description	Type	Dependency
APIG Administrator	Administrator permissions for APIG. Users with this permission can use all functions.	System-defined role	If a user needs to create, delete, or change resources of other services, the user must also be granted administrator permissions of the corresponding services in the same project.
APIG FullAccess	Full permissions for APIG. Users granted these permissions can use all functions of gateways.	System-defined policy	None
APIG ReadOnly Access	Read-only permissions for APIG. Users granted these permissions can only view gateways.	System-defined policy	None

You can view the content of the preceding roles and policies on the IAM console. For example, the content of the **APIG FullAccess** policy is as follows:

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "apig:*:*",
                "vpc:*:get*",
                "vpc:*:list*",
                "vpc:ports:create",
                "vpc:ports:update",
                "vpc:ports:delete",
                "vpc:publicIps:update",
                "FunctionGraph:function:listVersion",
                "FunctionGraph:function:list",
                "FunctionGraph:function:getConfig",
                "ecs:servers:list",
                "lts:groups:list",
                "lts:logs:list",
                "lts:topics:list"
            ],
            "Effect": "Allow"
        }
    ]
}
```


Related Documents

- Section "Service Overview" in the *Identity and Access Management User Guide*
- **Creating a User and Granting APIG Permissions**

7 Basic Concepts

API

A set of predefined functions that encapsulates application capabilities. You can create APIs and make them accessible to users.

When creating an API, you need to configure the basic information and the frontend and backend request paths, parameters, and protocols.

API Group

A collection of APIs used for the same service. API groups facilitate API management.

Environment

A stage in the lifecycle of an API. An environment, such as API testing or development environment, specifies the usage scope of APIs, facilitating API lifecycle management. The same API can be published in different environments.

To call an API in different environments, you need to add the **x-stage** header parameter to the request sent to call the API. The value of this parameter is an environment name.

Environment Variable

A variable that is manageable and specific to an environment. You can create variables in different environments to call different backend services using the same API.

Request Throttling

Controls the number of times APIs can be called by a user, app (credential), or IP address during a specific period to protect backend services.

Request throttling can be accurate to the minute and second.

Access Control

Access control policies are one of the security measures provided by APIG. They allow or deny API access from specific IP addresses or accounts.

App (Credential)

An entity that requests for APIs. An app can be authorized to access multiple APIs, and multiple apps can be authorized to access the same API.

Signature Key

Consists of a key and secret, which are used by backend services to verify the identity of API Gateway and ensure secure access.

When an API bound with a signature key is called, API Gateway adds signature information to the API requests. The backend service of the API signs the requests in the same way, and verifies the identity of API Gateway by checking whether the signature is consistent with that in the **Authorization** header sent by API Gateway.

VPC Channel (Load Balance Channel)

A method for accessing VPC resources from API Gateway, allowing you to selectively expose backend services deployed in VPCs to third-party users.

Custom Authentication

A mechanism defined with custom rules for API Gateway to verify the validity and integrity of requests initiated by API callers. The mechanism is also used for backend services to verify the requests forwarded by API Gateway.

The following two types of custom authentication are provided:

- Frontend custom authentication: A custom authorizer is configured with a function to authenticate requests for an API.
- Backend custom authentication: A custom authorizer can be configured to authenticate requests for different backend services, eliminating the need to customize APIs for different authentication systems and simplifying API development. You only need to create a function-based custom authorizer in API Gateway to connect to the backend authentication system.

Simple Authentication

Simple authentication facilitates quick response for API requests by adding the **X-Apig-AppCode** parameter (whose value is an AppCode) to the HTTP request header. API Gateway verifies only the AppCode and does not verify the request signature.

Gateway Response

Gateway responses are returned if API Gateway fails to process API requests. API Gateway provides default responses for multiple scenarios and allows you to

customize response status codes and content. You can add a gateway response in JSON format on the **API Groups** page.

8 Billing

APIG helps you build, manage, and deploy APIs at any scale.

To learn about the pricing of APIG and calculate the prices for using this service, go to the **Product Pricing Details** page.

Billing

Gateways are billed based on the **gateway edition** and **bandwidth**.

Billing for the Gateway Edition

Gateways are available in four editions: basic, professional, enterprise, and platinum. You need to pay the corresponding prices when purchasing these editions.

APIG provides two billing modes: pay-per-use and yearly/monthly. The pay-per-use mode is recommended if you cannot accurately predict your future service needs and want to avoid paying for unused resources. However, if you can accurately predict your future service needs, the yearly/monthly mode is more cost-effective.

- Yearly/Monthly: Provides a larger discount than the pay-per-use mode and is recommended for long-term users.
- Pay-per-use (hourly): You can start and stop gateways as needed. You will be billed based on the duration for which you use the gateways. Billing starts when a gateway is purchased and ends when the gateway is stopped due to arrears or is deleted. The minimum time unit is one second.
- Change of the billing mode: You can switch between the yearly/monthly and pay-per-use modes.

Billing for Bandwidth

If your API backend service is deployed on the public network, you will be charged for the bandwidth for forwarding API requests to the public network. The prices are calculated based on the **bandwidth** and the **duration** for which you use the gateway.

NOTE

- If your backend service is deployed in the same VPC as your gateway, the backend service can be accessed using a private IP address, and you do not need to purchase bandwidth for the gateway.
- If your gateway contains APIs that will be called from public networks, you need to purchase an EIP and bind it to the gateway.
- If the APIs in your gateway will be called within a VPC, you do not need to purchase or bind an EIP to the gateway.

Expiration and Overdue Payment

If your account is in arrears, you can view the arrears details in the Billing Center. To prevent related resources from being stopped or released, top up your account at the earliest. For details, see **Top-Up and Repayment**.

Unsubscription

To stop using yearly/monthly gateways, unsubscribe from them on the **Cloud Service Unsubscriptions** page of the Billing Center or in the gateway list of the APIG console.

