API Gateway

User Guide for Consumers

User Guide for Consumers

You can use API Gateway to call the API services enabled by other Alibaba Cloud users or third-party service providers. API Gateway provides for you a series of management services and support.

Call example

Based on the SDKs provided by API Gateway, you can write codes to call an API. You can also edit an HTTP request to call an API. The request structure of the API is as follows:

//If the domain name is a13db7999e494a90819cce500130034d.com.

//If the path is /web/cloudapi/mapping/service.

//If the query content is a=name, b=12.

//Then the URL of the request is as follows:

```
http://a13db7999e494a90819cce500130034d.com/web/cloudapi/mapping/service ? a=name&amp;amp;b=12
```

//Requesting method.

```
POST HttpMethod: POST
```

//Headers shall include signature information and certain parameters.

//For details about the methods of calculating and passing the encrypted signature, refer to Portal and Protocol.

```
X-Ca-Version:  1      // API version
X-Ca-Signature-Headers: X-Ca-Version,X-Ca-Key,X-Ca-Stage,X-Ca-Timestamp // Headers involved in signature calculation
X-Ca-Key: 60028305 //AppKey
X-Ca-Stage: test //Stage
X-Ca-Timestamp: 1456905123049 //Time stamp
X-Ca-Signature: UAaH/qteir4G9UK4YR+NWdyq+c1rjl0PvtO/C1Qo68U= // Signature
```

//Standard HTTP header.

```
Host:  a13db7999e494a90819cce500130034d.com        //Service address
Date: Wed 02 Mar 2016 07:52:02 GMT
User-Agent: Apache-HttpClient/4.1.2 (java 1.6)
```

```
Content-Type: application/x-www-form-urlencoded; charset=utf-8
```

//Body content.

```
Amount=11&InstanceId=ClientInstanceId&InstanceName=ClientInstanceName
```

An API request is constructed through the above content and the inputted parameters of the API. At the public beta stage, you need to obtain API documentation and details, such as the service address and path, in the deprecation environment from the API service provider. The AppKey is the key for the created app, which is used for identity verification. The app is your identity to call an API. For details, see subsequent content.

You need to create an app as your identity to call an API. Each app has a key pair consisting of an AppKey and AppSecret. These are used as the encrypted signature in your request and is verified by the gateway verifies.

> In API Gateway, create an app as your requester identity. During app creation, the system automatically assigns an AppKey and AppSecret. The AppKey indicates your identity. The AppSecret is the key used to encrypt the signature string and to verify the signature string on the server. When calling an API, you need to include the AppKey and AppSecret into the request. API Gateway verifies your identity through symmetric encryption. For details about the methods of calculating and passing the encrypted signature, refer to Portal and Protocol.
>
> The AppKey and AppSecret have all of the permissions on the APP, and therefore, should be kept secure. If any of the keys are released, you need to reset them on the API Gateway console.
>
> You can own multiple apps, to which different APIs are assigned based on your service requirements. Note that the API authorization is specific to an app, but not the Alibaba Cloud user account.
>
> On the API Gateway console, you can manage apps, view details, manage keys, and view authorized apps.

Authorization grants an app the permission to call an API. Your app needs authorization for an API before calling it. At the public beta stage, the API service provider establishes the permission relationship between an app and API.

> At the public beta stage, the API service provider establishes the authorization. You need to provide the API service provider with your AppID or Alibaba Mail account to indicate that an

APP is given for authorization. After authorization, you can use this app to call the API.

At the public beta stage, you do not have permission to establish or revoke authorization. You can only view the authorized APIs under an app on the console. If you need to revoke the authorization for an API, contact the API service provider.

When you call an API, API Gateway uses the AppKey and AppSecret to calculate the encrypted signature for identity verification.

In API Gateway, you need to use an app as your identity to call an API. During app creation, the system automatically assigns an AppKey and AppSecret which is used for the server to verify your identity.

Either the HTTP or HTTPS request must include signature information. The AppKey indicates your identity. The AppSecret is used to encrypt and verify the signature string on the server. For details about the methods of calculating and passing the encrypted signature, refer to Portal and Protocol.

There can be up to 10 apps under each account, and each app name must be unique.

At the public beta stage, you do not have permission to authorize an app or revoke authorization. Only the API service provider has such permissions.

Your request shall include the signature information. For details, refer to Portal and Protocol.

Domain name

- Each API belongs to an API group, and each API group has a unique domain name. These independent domain names are bound by the service provider. API Gateway uses a domain name to locate an API group.
- The domain name is in the format of “www.[Independent domain name].com/[Path]?[HTTPMethod].” At the public beta stage, the API user needs to obtain this domain name offline from the API service provider.
- Alibaba Cloud API Gateway uses the domain name to locate a unique API group, and then locate the unique API through Path+HTTPMethod.
- You need to obtain API documentation in the deprecation environment from the API service provider. This documentation must include necessary parameter information, such as the

domain name and path.

System headers

- [Required] X-Ca-Key: AppKey
- [Required] X-Ca-Signature: Signature string
- [Optional] X-Ca-Timestamp: The time stamp in milliseconds passed by the API caller, that is, the milliseconds of the time from January 1, 1970 until now. By default, it is valid within 15 minutes.
- [Optional] X-Ca-Nonce: The UUID generated by the API caller. This header is used with the time stamp to prevent replay.
- [Optional] Content-MD5: When the request body is not a Form, calculate the MD5 value of the body and send that value to the cloud gateway for checking.
- [Optional] X-Ca-Stage: The stage where the requested API belongs. Only test and release are supported, and the default value is release.

Signature verification

For details about the demo (Java) of signature calculation, refer to here.

The signature calculation procedure is as follows:

Organize the strings involved in signature calculation

```
String stringToSign=
HTTPMethod + "\n" +
Accept + "\n" +
Content-MD5 + "\n"
Content-Type + "\n" +
Date + "\n" +
Headers +
Url
```

Each letter of the HTTPMethod value should be capitalized.

If “Accept”, “Content-MD5”, “Content-Type”, and “Date” are empty, add a linefeed “\n.” If “Headers” is empty, “\n” is not required. The specified “Headers” includes “\n.” For details, refer to the headers organization method described below.

Content-MD5

Content-MD5 indicates the MD5 value of the body. MD5 is only calculated when the body is not a Form. The calculation method is as follows:

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

The “bodyStream” indicates the byte array.

Headers

Headers indicates the keys and values of the headers involved in signature calculation. Note that X-Ca-Signature and X-Ca-Signature-Headers are excluded in Headers signature calculation.

Headers organization method:

Rank the keys of all Headers involved in signature calculation in lexicographic order and then splice them in the following method:

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

URL

URL indicates the Form parameter in the Path+Query+Body. The organization method is as follows:

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, and “?” does not need to be added.

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

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

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"));
```

The “secret” indicates the key corresponding to an app.

Pass the signature

Put the calculated signature in the Header of the Request. The key is X-Ca-Signature.

Separate the keys of all Headers involved in signature calculation by commas and put them in the Header of the Request regardless of the order. The key is X-Ca-Signature-Headers.

For details about the demo of signature calculation, refer to here.