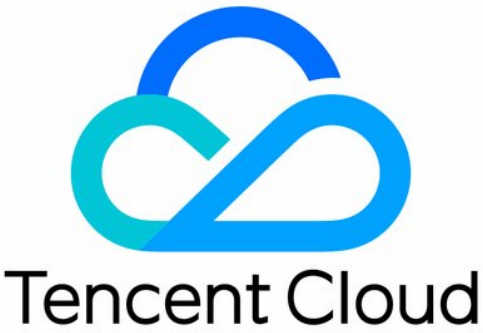
Serverless Cloud Function

API Documentation

Product Introduction



Contents

API Documentation
- Introduction
- API Category
  - Making API Requests
    - Request Structure
    - Common Params
    - Signature
    - Responses
- History
  - Function APIs
    - CreateFunction
    - DeleteFunction
    - GetFunction
    - GetFunctionLogs
    - Invoke
    - ListFunctions
    - UpdateFunctionCode
    - UpdateFunctionConfiguration
  - Trigger APIs
    - CreateTrigger
    - DeleteTrigger
- Data Types
  - SCF API 2017
    - API Overview
    - API Overview
    - Update History
      - Calling Mode
        - Request Body
          - Introduction of Request Structure
          - Common Request Parameters
          - API-specific Request Parameters
          - Final Request Format
        - Returning Results
          - 正确返回结果
          - 错误返回结果

Error Codes
签名方法
Function API
Acquire Function List
Acquire Function Details
Create Function
Delete Function
Update Function
Acquire Operation Log of Functions
Obtaining Function Monitoring Data
Run Function
Trigger API
Configure Function Trigger
Delete Function Trigger

API Documentation Introduction

Last updated：2018-09-28 17:01:27

Thank you for choosing Tencent Cloud Serverless Cloud Function (SCF). By hosting your code and using user-configured triggers, SCF automatically executes your applications based on the trigger event of the triggers, providing you with FaaS (Function as a Service) type services.

SCF is Tencent Cloud's serverless execution environment. With SCF, you don't need to purchase and manage servers. Instead, all you need to do is write the core code using the languages supported by the platform and set the conditions for code execution. After that, your code can be ran on the Tencent Cloud infrastructure in an elastic and secure manner. The underlying computing resources and tasks are fully managed by Tencent Cloud, including maintenance of server CPUs, memories, networks and other configurations/resources, code deployment, elastic scaling and load balancing. The code is executed on demand and there is no charge when it is idle.

Glossary

Term	Full Name	Description
SCF	Serverless Cloud Function	By hosting your code and using user-configured triggers, SCF automatically executes your applications based on the trigger event of the triggers, providing you with FaaS (Function as a Service) type services
Trigger	Function Trigger	SCF functions are triggered by events. Event sources serve as the triggers, which are usually resource objects of other products such as a COS bucket, a CMQ topic or a timer

API Category

Last updated：2018-09-28 17:01:27

Function-related APIs

API Name	API Description
CreateFunction	Create a function
DeleteFunction	Delete a function
GetFunction	Get function details
GetFunctionLogs	Get function logs
Invoke	Run a function
ListFunctions	Get function list
UpdateFunctionCode	Update function code
UpdateFunctionConfiguration	Update function configuration

Trigger-related APIs

API Name	API Description
CreateTrigger	Set function triggering mode
DeleteTrigger	Delete a trigger

Making API Requests
Request Structure

Last updated：2018-09-28 17:01:28

1. Service Address

The Tencent Cloud API is divided into different functional modules by function and each module is accessed via a unique domain name. The API supports access from either a nearby region or a specified region. For example, Cloud Virtual Machine's domain name for the nearby region is cvm.tencentcloudapi.com, and its domain name for the Guangzhou region is cvm.ap-guangzhou.tencentcloudapi.com.

Below lists the currently supported regions:

Access region	Domain name
Nearby region (recommended)	*.tencentcloudapi.com
South China (Guangzhou)	*.ap-guangzhou.tencentcloudapi.com
East China (Shanghai)	*.ap-shanghai.tencentcloudapi.com
North China (Beijing)	*.ap-beijing.tencentcloudapi.com
Southwest China (Chengdu)	*.ap-chengdu.tencentcloudapi.com
Southwest China (Chongqing)	*.ap-chongqing.tencentcloudapi.com
Southeast Asia (Seoul)	*.ap-seoul.tencentcloudapi.com
East China (Shanghai Financial)	*.ap-shanghai-fsi.tencentcloudapi.com
South China (Shenzhen Financial)	*.ap-shenzhen-fsi.tencentcloudapi.com
Southeast Asia (Singapore)	*.ap-singapore.tencentcloudapi.com
Asia Pacific (Mumbai)	*.ap-mumbai.tencentcloudapi.com
Western America (Silicon Valley)	*.na-siliconvalley.tencentcloudapi.com
Eastern America (Virginia)	*.na-ashburn.tencentcloudapi.com

Note: As financial and non-financial availability zones are isolated, when accessing the services in a financial availability zone (with the common parameter Region specifying a financial availability zone), it is necessary to specify a domain name with the financial availability zone, preferably in the same region as specified in Region.

2. Communications Protocol

All the interfaces of the Tencent Cloud API communicate via HTTPS, providing highly secure communications tunnels.

3. Request Method

POST and GET requests are supported. Currently, POST requests only support Content-Type application/x-www-form-urlencoded.

4. Character Encoding

Only UTF-8 encoding is used.

Common Params

Last updated：2018-09-28 17:01:28

The common parameters are used to authenticate the user and API. If not necessary, these parameters are not described in individual API documents. However, they have to be carried by each request to initiate properly.

Parameter name	Type	Required	Description
Action	String	Yes	The name of the command API for the specific operation. For example, if you want to call the instance list query API of Cloud Virtual Machine, then the Action parameter is DescribeInstances.
Region	String	Yes	The Region parameter used to identify the region whose data you want to operate on.
Timestamp	Integer	Yes	The current UNIX timestamp which records when an API request is initiated. For example, 1529223702. If it is too different from the current time, it will cause a signature expiry error.
Nonce	Integer	Yes	A random positive integer used to prevent replay attacks along with Timestamp.
SecretId	String	Yes	The identifying SecretId obtained on the Cloud API Key page. A SecretId corresponds to a unique SecretKey which is used to generate the request signature (Signature).
Signature	String	Yes	Request signature used to verify the validity of this request. This is generated based on the actual input parameters. For details on how to generate, see the API authentication document.
Version	String	Yes	The version of the API. For example, 2017-03-12
SignatureMethod	String	No	Signature mode. Currently, HmacSHA256 and HmacSHA1 are supported. The HmacSHA256 algorithm is used to verify the signature only when this parameter is specified as HmacSHA256. In other cases, the signature is verified with HmacSHA1.

Parameter name	Type	Required	Description
Token	String	No	The token used by the temporary certificate, which needs to be used in conjunction with the temporary key. The temporary key and token need to be obtained through the access management service call API. Long-term keys do not require a token.

Assuming you want to query the list of Cloud Virtual Machine instances in the Guangzhou region, the form of the request link may be as follows:

```
https://cvm.tencentcloudapi.com/?Action=DescribeInstances
&SecretId=xxxxxxx
&Region=ap-guangzhou
&Timestamp=1402992826
&Nonce=345122
&Signature=xxxxxxxx
&Version=2017-03-12
```

Region List

The possible values for the Region field in all APIs of this product are as shown below. If an API does not support any of the listed regions, the detail will be described separately in the API document.

Region	Value
North China (Beijing)	ap-beijing
Southwest China (Chengdu)	ap-chengdu
South China (Guangzhou)	ap-guangzhou
South China (Guangzhou Open)	ap-guangzhou-open
Asia Pacific (Mumbai)	ap-mumbai
East China (Shanghai)	ap-shanghai

Signature

Last updated：2018-09-28 17:01:29

Tencent Cloud API authenticates each access request, i.e. each request needs to include signature information (Signature) in the common request parameters to verify the identity of the requester. The Signature is generated by the security credentials which include SecretId and SecretKey. If you don't have the security credentials yet, please go to the Cloud API Key page to apply; otherwise, you cannot call the Cloud API.

1. Applying for Security Credentials

Before using the Cloud API for the first time, go to the Cloud API Key page to apply for security credentials. Security credentials include SecretId and SecretKey:

> SecretId is used to identify the API caller. SecretKey is used to encrypt the signature string and verify it on the server. **You must keep your security credentials private and avoid disclosure.**

You can apply for the security credentials in the following steps:

1) Log in to Tencent Cloud Management Center Console.

2) Go to the Cloud API Key page

3) On the Cloud API Key page, click [New] to create a pair of SecretId/SecretKey

> **A developer account can have up to two pairs of SecretId/SecretKey.**

2. Generating Signature String

After obtaining the security credentials (SecretId and SecretKey), you can generate a signature string. Below describes how to generating the signature string in details:

Assume that the SecretId and SecretKey are:

SecretId: AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE SecretKey: Gu5t9xGARNpq86cd98joQYCN3EXAMPLE

Note: This is just an sample here. For actual operations, use your real SecretId and SecretKey!

Take the Cloud Virtual Machine's request to view the instance list (DescribeInstances) as an example. When you invoke this interface, the request parameters may be as follows:

Parameter name	English	Parameter value
Action	Method name	DescribeInstances
SecretId	Key ID	AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE
Timestamp	Current timestamp	1465185768
Nonce	Random positive integer	11886
Region	Region where the instance is located	ap-guangzhou
InstanceIds.0	ID of the to-be-queried instance	ins-09dx96dg
Offset	Offset	0
Limit	Allowed maximum output	20
Version	Interface version number	2017-03-12

2.1. Sorting Parameters

First, sort all request parameters by parameter name in ascending lexicographical order (ASCII code). Notes: 1) Sort the parameters only by parameter name and keep the parameter values corresponding which don't participate in the ranking; 2) Rank the parameters by ASCII code, for example, InstanceIds.2 should be ranked after InstanceIds.12 (not by alphabet nor by value). You can do this with the aid of related sorting functions in the programming language, such as the ksort function in PHP. The sorting results of the sample parameters above are as below:

```
{
'Action' : 'DescribeInstances',
'InstanceIds.0' : 'ins-09dx96dg',
'Limit' : 20,
```

```
'Nonce' : 11886,
'Offset' : 0,
'Region' : 'ap-guangzhou',
'SecretId' : 'AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE',
'Timestamp' : 1465185768,
'Version': '2017-03-12',
}
```

When developing in another programming language, you can sort these sample parameters and it would work as long as you get the same results.

2.2. Generating Request String

This step generates a request string. Format the request parameters sorted in the previous step into the form of "parameter name"="parameter value". For example, for the Action parameter, its parameter name is "Action" and its parameter value is "DescribeInstances", so it will become Action=DescribeInstances after formatted. **Note: The "parameter value" is the original value but not the value after URL encoding.**

Then, splice together the formatted parameters with "&". The resulting request string is as below:

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region
=ap-guangzhou&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768
&Version=2017-03-12
```

2.3. Generating Signature Original String

This step generates a signature original string, which consists of the following parameters:

1) Request method: POST and GET modes are supported, and GET is used here for the request. Please note that the method name should be in all capital letters. 2) Request host: The domain name of the request to view the list of instances (DescribeInstances) is cvm.tencentcloudapi.com. The actual request domain name varies for different modules to which the interface belongs. For details, see the instructions of the specific interface. 3) Request path: The request path in the current version of Cloud API is fixed to / . 4) Request string: That is the request string generated in the previous step.

The splicing rule of the signature original string is:

```
Request method + request host + request path + ? + request string
```

The splicing result of the sample is:

```
GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12
```

2.4. Generating Signature String

This step generates a signature string. First, use the HMAC-SHA1 algorithm to sign the **signature original string** obtained in the previous step, and then encode the generated signature string using Base64 to obtain the final signature string.

The specific code is as below with the PHP language as an example:

```
$secretKey = 'Gu5t9xGARNpq86cd98joQYCN3EXAMPLE';
$srcStr = 'GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

The final signature string is:

```
EliP9YW3pW28FpsEdkXt/+WcGeI=
```

When developing in another programming language, you can sign and verify the original in the sample above and it would work as long as you get the same results.

3. Encoding Signature String

The generated signature string cannot be directly used as a request parameter and needs to be URL encoded. **Note: If your request method is GET, or the request method is POST and the Content-Type is application/x-www-form-urlencoded, then all the request parameter values need to be URL encoded. Non-ASCII characters need to be encoded with UTF-8 before URL encoding.** For example, if the signature string generated in the previous step is EliP9YW3pW28FpsEdkXt/+WcGeI=, the final signature string request parameter (Signature) is EliP9YW3pW28FpsEdkXt/+WcGeI=, which will be used to generate the final request URL.

4. Authentication Failure

The following error codes for authentication failure exist based on the actual conditions.. Please cope with the errors accordingly.

Error code	Error description
AuthFailure.SignatureExpire	Signature expired
AuthFailure.UnauthorizedOperation	Request not authorized through CAM
AuthFailure.SecretIdNotFound	Key does not exist
AuthFailure.SignatureFailure	Signature error
AuthFailure.TokenFailure	Token error
AuthFailure.MFAFailure	MFA error
AuthFailure.InvalidSecretId	Invalid key (not Cloud API key type)

5. Signature Demo

When calling API 3.0, it is recommended to use the corresponding Tencent Cloud SDK 3.0 which encapsulates the signature process, enabling you to focus on only the specific interfaces provided by the product when developing. See SDK Center for more information. Currently, the following programming languages are supported:

- Python
- Java
- PHP
- Go
- JavaScript

In order to explain the signing process more clearly, the process described above is implemented below with Java as an example. The request domain name, called interface and parameter values in the sample are used here.

Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
```

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPIDemo {
private final static String CHARSET = "UTF-8";

public static String sign(String s, String key, String method) throws Exception {
Mac mac = Mac.getInstance(method);
SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
mac.init(secretKeySpec);
byte[] hash = mac.doFinal(s.getBytes(CHARSET));
return DatatypeConverter.printBase64Binary(hash);
}

public static String getStringToSign(TreeMap<String, Object> params) {
StringBuilder s2s = new StringBuilder("GETcvm.tencentcloudapi.com/?" );
// When signing, the parameters need to be sorted in lexicographical order. TreeMap is used here to
guarantee the correct order
for (String k : params.keySet()) {
s2s.append(k).append("=").append(params.get(k).toString()).append("&");
}
return s2s.toString().substring(0, s2s.length() - 1);
}

public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException
{
StringBuilder url = new StringBuilder("https://cvm.tencentcloudapi.com/?" );
// There is no requirement for the order of the parameters in the actual request URL
for (String k : params.keySet()) {
// The request string needs to be URL encoded. As the Key is all in English letters, only the value is UR
L encoded here
url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).append
("&");
}
return url.toString().substring(0, url.length() - 1);
}

public static void main(String[] args) throws Exception {
TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap can enable automati
c sorting
// A random numbers should be used when actually calling, for example: params.put("Nonce", new R
andom().nextInt(java.lang.Integer.MAX_VALUE));
params.put("Nonce", 11886); // Common parameter
// The current time of the system should be used when actually calling, for example: params.put("Tim
```

```
estamp", System.currentTimeMillis() / 1000);
params.put("Timestamp", 1465185768); // Common parameter
params.put("SecretId", "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"); // Common parameter
params.put("Action", "DescribeInstances"); // Common parameter
params.put("Version", "2017-03-12"); // Common parameter
params.put("Region", "ap-guangzhou"); // Common parameter
params.put("Limit", 20); // Business parameter
params.put("Offset", 0); // Business parameter
params.put("InstanceIds.0", "ins-09dx96dg"); // Business parameter
params.put("Signature", sign(getStringToSign(params), "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE",
"HmacSHA1")); // Common parameter
System.out.println(getUrl(params));
}
}
```

The final output URL: https://cvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Signature=EliP9YW3pW28FpsEdkXt/+WcGeI=&Timestamp=1465185768&Version=2017-03-12

Notes: The key in the example is fictitious, and the timestamp is not the current time of the system, so if this URL is opened in the browser or called using commands such as curl, an authentication error will be returned: Signature expired. In order to get a URL that can work properly, you need to replace the SecretId and SecretKey in the example with your real credentials and use the current time of the system as the Timestamp.

Responses

Last updated：2018-09-28 17:01:30

Returned Result for Success

Take viewing the instance status list (DescribeInstancesStatus) version 2017-03-12 through the Cloud Virtual Machine interface as an example. If the call succeeds, the possible returned result is as follows:

```
{
"Response": {
"TotalCount": 0,
"InstanceStatusSet": [],
"RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
}
}
```

- `Response` and its internal `RequestId` are fixed fields and will be returned as long as processed by the API no matter whether the request succeeds.
- `RequestId` is used to uniquely identify an API request. If the API is abnormal, you can contact us and provide the ID for troubleshooting.
- Except for the fixed fields, all the fields are defined by the specific API. For the fields returned by different APIs, see the definitions in the API documentation. In this example, TotalCount and InstanceStatusSet are the fields defined by the DescribeInstancesStatus API. As the user who calls the request does not have a Cloud Virtual Machine instance yet, TotalCount returns a value of 0 in this case and the InstanceStatusSet list is empty.

Returned Result for Error

If the call fails, the returned result may look like the example below:

```
{
"Response": {
"Error": {
"Code": "AuthFailure.SignatureFailure",
"Message": "The provided credentials could not be validated. Please check your signature is correct."
},
"RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
```

```
}
}
```

- The presence of the Error field indicates that the request call failed. The Error field and its internal Code and Message fields, must be returned when the call fails.
- Code indicates the error code of the specific error. When the request goes wrong, you can use this error code to locate the cause and solution in the common error code list and the error code list corresponding to the current API.
- Message shows the specific cause of this error. The message text is subject to change or update as the business develops or the experience gets optimized, so you should not rely on this return value.
- RequestId is used to uniquely identify an API request. If the API is abnormal, you can contact us and provide the ID for troubleshooting.

Common Error Codes

If there is an Error field in the returned result, it means that the API call failed. The Code field in Error indicates the error code. The error codes that may appear for all Tencent Cloud services are common error codes, which are listed below:

Error code	Error description
InvalidParameter	Wrong parameter (including errors with parameter format, type, etc.)
InvalidParameterValue	Wrong parameter value
MissingParameter	Missing parameter; a required parameter is missing
UnknownParameter	Unknown parameter; an undefined parameter passed in by the user will cause this error
AuthFailure	Error with CAM signature/authentication
InternalError	Internal error
InvalidAction	API does not exist
UnauthorizedOperation	Unauthorized operation
RequestLimitExceeded	The number of requests exceeds the frequency limit
NoSuchVersion	API version does not exist
UnsupportedRegion	API does not support the passing region

Error code	Error description
UnsupportedOperation	Unsupported operation
ResourceNotFound	Resource does not exist
LimitExceeded	Quota limit is exceeded
ResourceUnavailable	Resource not available
ResourceInsufficient	Insufficient resource
FailedOperation	Operation failed
ResourceInUse	Resource is in use
DryRunOperation	DryRun operation, which means the request will succeed, but an unnecessary DryRun parameter is passed in

History

Last updated：2018-09-28 17:01:30

The Second Release

Release time: 19:15:23, July 26, 2018

This release contains:

Improvement to existing documentation.

New interfaces:

- CreateFunction
- CreateTrigger
- DeleteFunction
- DeleteTrigger
- GetFunction
- GetFunctionLogs
- ListFunctions
- UpdateFunctionCode
- UpdateFunctionConfiguration

Modified interfaces:

- Invoke
  - New input parameters: Namespace

New data structures:

- Code
- Environment
- Filter
- Function
- FunctionLog
- Trigger
- Variable
- VpcConfig

The First Release

Release time: 15:58:40, May 17, 2018

This release contains:

Improvement to existing documentation.

New interfaces:

- Invoke

New data structures:

- Result

Function APIs
CreateFunction

Last updated：2018-09-28 17:01:35

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API creates a new function based on the parameters passed in.

API request frequency limit: 10 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: CreateFunction
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	Yes	String	Name of the new function. It can contain 2 to 60 characters, including lower-case and upper-case English letters, numbers, dashes and underscores, must begin with a letter and cannot end with a dash or underscore
Code	Yes	Code	Code of the function. Note: COS and ZipFile cannot be specified at the same time

Parameter name	Required	Type	Description
Handler	No	String	Name of the function handler. This name supports the "file name.handler name" form where the file name and handler name are separated with a ".". File and handler names must be of 2-60 characters and start and end with letters, and can contain letters, numbers, underscores and dashes in the middle
Description	No	String	Description of the function. It can contain up to 1,000 characters including English letters, numbers, spaces, commas, line breaks, periods and Chinese characters
MemorySize	No	Integer	Memory size available for the function during execution between 128MB and 1,536MB in an increment of 128MB; 128MB by default
Timeout	No	Integer	Maximum execution duration of the function in seconds; the value can be between 1 and 300 seconds; 3 seconds by default
Environment	No	Environment	Environment variable of the function
Runtime	No	String	Runtime environment of the function; currently, only the following ones are supported: Python2.7, Python3.6, Nodejs6.10, PHP5, PHP7, Golang1 and Java8; Python2.7 by default
VpcConfig	No	VpcConfig	VPC configuration of the function

3. Output Parameters

Parameter name	Type	Description
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError	Internal error
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.Code	Wrong Code parameter passed in.
InvalidParameterValue.Description	Wrong Description parameter passed in.
InvalidParameterValue.Environment	Wrong Environment parameter passed in.
InvalidParameterValue.FunctionName	Function does not exist.
InvalidParameterValue.Handler	Wrong Handler parameter passed in.
InvalidParameterValue.Runtime	Wrong Runtime parameter passed in.
LimitExceeded.Function	The number of functions exceeds the upper limit.
LimitExceeded.Memory	Memory exceeds the upper limit.
LimitExceeded.Timeout	Timeout exceeds the upper limit.
MissingParameter.Code	Code parameter missing.
ResourceInUse.FunctionName	FunctionName already exists.
UnauthorizedOperation.CAM	CAM authentication failed.
UnauthorizedOperation.Region	Error with Region.

5. Examples

Example 1. Creating Function

Input Example

```
https://scf.tencentcloudapi.com/?Action=CreateFunction
&FunctionName=<FunctionName>
&Handler=<function.handler>
```

```
&Code.CosBucketName=<CosBucketName>
&Code.CosObjectName=<CosObjectName>
&<Common request parameter>
```

Output Example

```
{
"Response": {
"RequestId": "eac6b301-a322-493a-8e36-83b295459397"
}
}
```

6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

DeleteFunction

Last updated：2018-09-28 17:01:34

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API deletes the function based on the parameters passed in.

API request frequency limit: 20 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: DeleteFunction
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	Yes	String	Name of the function to be deleted

3. Output Parameters

Parameter name	Type	Description
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
ResourceNotFound.FunctionName	Function does not exist.
UnauthorizedOperation.CAM	CAM authentication failed.

5. Examples

Example 1. Deleting Function

Input Example

```
https://scf.tencentcloudapi.com/?Action=DeleteFunction
&FunctionName=<FunctionName>
&<Common request parameter>
```

Output Example

```
{
"Response": {
"RequestId": "eac6b301-a322-493a-8e36-83b295459397"
}
}
```

6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go

- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

GetFunction

Last updated：2018-09-28 17:01:34

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API gets the details of a function, including fields such as the name, code, processing method, associated trigger and timeout.

API request frequency limit: 20 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: GetFunction
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	Yes	String	Name of the function whose details are to be obtained
Qualifier	No	String	Version number of the function
ShowCode	No	String	This indicates whether to display the code; TRUE means displaying the code, while FALSE means hiding the code; the code will not be displayed for entry files of more than 1MB

3. Output Parameters

Parameter name	Type	Description
ModTime	Timestamp	Last modified time of the function
CodeInfo	String	Code of the function
Description	String	Description of the function
Triggers	Array of Trigger	Trigger list of the function
Handler	String	Entry of the function
CodeSize	Integer	Code size of the function
Timeout	Integer	Timeout of the function
FunctionVersion	String	Version of the function
MemorySize	Integer	Maximum available memory for the function
Runtime	String	Runtime environment of the function
FunctionName	String	Name of the function
VpcConfig	VpcConfig	VPC of the function
UseGpu	String	This indicates whether to use GPU
Environment	Environment	Environment variable of the function
CodeResult	String	This indicates whether the code is correct
CodeError	String	Error message of the code
ErrNo	Integer	Error code of the code
Namespace	String	Namespace of the function
Role	String	Role bound with the function
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError	Internal error
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
ResourceNotFound.FunctionName	Function does not exist.
UnauthorizedOperation	Unauthorized operation
UnauthorizedOperation.CAM	CAM authentication failed.

5. Examples

Example 1. Getting Function Details

Input Example

```
https://scf.tencentcloudapi.com/?Action=GetFunction
&FunctionName=<FunctionName>
&<Common request parameter>
```

Output Example

```
{
"Response": {
"CodeError": "",
"CodeInfo": "",
"CodeResult": "failed",
"CodeSize": 0,
"Description": "",
"Environment": {
"Variables": []
},
"ErrNo": 0,
"FunctionName": "ledDummyAPITest",
"FunctionVersion": "$LATEST",
"Handler": "scfredis.main_handler",
```

```
"MemorySize": 128,
"ModTime": "2018-06-07 09:52:23",
"Namespace": "default",
"RequestId": "a1ffbba5-5489-45bc-89c5-453e50d5386e",
"Role": "",
"Runtime": "Python2.7",
"Timeout": 3,
"Triggers": [],
"UseGpu": "FALSE",
"VpcConfig": {
"SubnetId": "",
"VpcId": ""
}
}
}
```


6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

GetFunctionLogs

Last updated：2018-09-28 17:01:33

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API returns the function logs according to the configured log query criteria.

API request frequency limit: 20 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: GetFunctionLogs
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	No	String	Name of the function
Offset	No	Integer	Offset of the data; Offset+Limit cannot be greater than 10000
Limit	No	Integer	Length of the returned data; Offset+Limit cannot be greater than 10000
Order	No	String	This indicates whether the logs are sorted in ascending or descending order; possible values: desc and acs

Parameter name	Required	Type	Description
OrderBy	No	String	This is to sort logs by a specific field; the following fields are supported: startTime, functionName, requestId, duration and memUsage
Filter	No	Filter	Log filtering condition. This can be used to distinguish between logs for successes and logs for errors. filter.retCode=not0 indicates that only the logs for errors are returned, while filter.retCode=is0 indicates that only the logs for successes are returned; if this parameter is blank, all logs are returned
Qualifier	No	String	Version of the function
FunctionRequestId	No	String	The requestId that executes this function
StartTime	No	Timestamp	The specific start time of the query, for example, 2017-05-16 20:00:00. It can be only less than one day before the endTime
EndTime	No	Timestamp	The specific end time of the query, for example, 2017-05-16 20:59:59. It can be only less than one day after the startTime

3. Output Parameters

Parameter name	Type	Description
TotalCount	Integer	Total number of function logs
Data	Array of FunctionLog	Function log information
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError	Internal error
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.DateTime	Wrong DateTime parameter passed in.
LimitExceeded.Offset	Offset is out of range.
UnauthorizedOperation.CAM	CAM authentication failed.

5. Examples

Example 1. Getting Function Logs

Input Example

```
https://scf.tencentcloudapi.com/?Action=GetFunctionLogs
&FunctionName=<FunctionName>
&<Common request parameter>
```

Output Example

```
{
"Response": {
"Data": [
{
"BillDuration": 100,
"Duration": 0.532,
"FunctionName": "APITest",
"InvokeFinished": 1,
"Log": "",
"MemUsage": 3174400,
"RequestId": "bc309eaa-6d64-11e8-a7fe-5254000b4175",
"RetCode": 1,
"RetMsg": "Success",
"StartTime": "2018-06-11 18:46:45"
```

```
}
],
"RequestId": "e2571ff3-da04-4c53-8438-f58bf057ce4a",
"TotalCount": 1
}
}
```


6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

Invoke

Last updated：2018-09-28 17:01:32

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API is used to run the function.

API request frequency limit: 20 times/second.

2. Input Parameters

The following list of request parameters lists only the interface request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this interface: Invoke
Version	Yes	String	Common parameter; the value for this interface: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	Yes	String	Name of the function
InvocationType	No	String	RequestResponse (sync) and Event (asynchronous); sync by default
Qualifier	No	String	Version number that triggers the function
ClientContext	No	String	Parameters when running the function; input in JSON format; the maximum supported parameter length is 1MB
LogType	No	String	If this field is specified when calling synchronously, the return value will contain 4K logs; possible values: None and Tail; None by default. When the value is Tail, the logMsg field in the return parameter will contain the corresponding function execution log

Parameter name	Required	Type	Description
Namespace	No	String	Namespace

3. Output Parameters

Parameter name	Type	Description
Result	Result	Execution result of the function
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the interface business logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError	Internal error
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.Param	The input parameter is not in standard JSON format.
ResourceNotFound.FunctionName	Function does not exist.
ResourceUnavailable.InsufficientBalance	The balance is insufficient. Please top up first.
UnauthorizedOperation.CAM	CAM authentication failed.

5. Examples

Example 1. Running Function

Input Example

```
https://scf.tencentcloudapi.com/?Action=Invoke
&FunctionName=xxxx
&<Common request parameter>
```

Output Example

```
{
"Response": {
"RequestId": "c2af8a64-c922-4d55-aee0-bd86a5c2cd12",
"Result": {
"BillDuration": 100,
"Duration": 0.826,
"ErrMsg": "",
"FunctionRequestId": "6add56fa-58f1-11e8-89a9-5254005d5fdb",
"InvokeResult": 0,
"Log": "",
"MemUsage": 3207168,
"RetMsg": "hello from scf"
}
}
}
```

6. Other Resources

Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

ListFunctions

Last updated：2018-09-28 17:01:32

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API returns related function information based on the query parameters passed in.

API request frequency limit: 20 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: ListFunctions
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
Order	No	String	This indicates whether the returned results are sorted in ascending or descending order; possible values: ASC and DESC
Orderby	No	String	This indicates by which field to sort the returned results; the following fields are supported: AddTime, ModTime, FunctionName
Offset	No	Integer	Data offset; 0 by default
Limit	No	Integer	Length of the returned data; 20 by default
SearchKey	No	String	Support for fuzzy match with FunctionName

3. Output Parameters

Parameter name	Type	Description
Functions	Array of Function	Function list
TotalCount	Integer	Total number
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.Order	Wrong Order parameter passed in.
InvalidParameterValue.Orderby	Wrong Orderby parameter passed in.
UnauthorizedOperation.CAM	CAM authentication failed.
UnauthorizedOperation.Region	Error with Region.

5. Examples

Example 1. Getting Function List

Input Example

```
https://scf.tencentcloudapi.com/?Action=ListFunctions
&Limit=2
&Order=ASC
&<Common request parameter>
```

Output Example

```
{
"Response": {
"Functions": [
{
"AddTime": "2018-04-08 15:18:49",
"FunctionId": "lam-xxxxxxx",
"FunctionName": "test",
"ModTime": "2018-04-08 19:02:20",
"Namespace": "default",
"Runtime": "Python2.7"
}
],
"RequestID": "3c140219-cfe9-470e-b241-907877d6fb03",
"TotalCount": 1
}
}
```


6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

UpdateFunctionCode

Last updated：2018-09-28 17:01:31

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API updates the function code based on the parameters passed in.

API request frequency limit: 20 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: UpdateFunctionCode
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
Handler	Yes	String	Name of the function handler. This name supports the "file name.function name" format. File and function names must be of 2-60 characters and start and end with letters, and can contain letters, numbers, underscores and dashes
FunctionName	Yes	String	Name of the function to be modified
CosBucketName	No	String	Name of the object bucket
CosObjectName	No	String	Path of the COS object
ZipFile	No	String	This contains a .zip file of the function code file and its dependencies. When using this API, the content of the .zip file needs to be encoded with Base64. It can be up to 20MB

Parameter name	Required	Type	Description
CosBucketRegion	No	String	Region of the COS. For the Beijing region, you need to pass in ap-beijing; for the Beijing Zone 1, you need to pass in ap-beijing-1; for other regions, this parameter can be left blank

3. Output Parameters

Parameter name	Type	Description
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.Code	Wrong Code parameter passed in.
InvalidParameterValue.Handler	Wrong Handler parameter passed in.
ResourceNotFound.FunctionName	Function does not exist.
UnauthorizedOperation.CAM	CAM authentication failed.

5. Examples

Example 1. Updating Function Code

Input Example

```
https://scf.tencentcloudapi.com/?Action=UpdateFunctionCode
&Handler=index.main
&CosBucketName=<CosBucketName>
&CosObjectName=<CosObjectName>
&<Common request parameter>
```

Output Example

```
{
"Response": {
"RequestId": "eac6b301-a322-493a-8e36-83b295459397"
}
}
```

6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

UpdateFunctionConfiguration

Last updated：2018-09-28 17:01:30

1. API Description

API request domain name: scf.tencentcloudapi.com.

This API updates the function configuration based on the parameters passed in.

API request frequency limit: 20 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: UpdateFunctionConfiguration
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	Yes	String	Name of the function to be modified
Description	No	String	Description of the function. It can contain up to 1,000 characters including English letters, numbers, spaces, commas, periods and Chinese characters
MemorySize	No	Integer	Memory size of the function runtime between 128MB and 1,536MB; 128MB by default
Timeout	No	Integer	Maximum execution duration of the function in seconds; the value can be between 1 and 300 seconds; 3 seconds by default

Parameter name	Required	Type	Description
Runtime	No	String	Runtime environment of the function; currently, only the following ones are supported: Python2.7, Python3.6, Nodejs6.10, PHP5, PHP7, Golang1 and Java8
Environment	No	Environment	Environment variable of the function
VpcConfig	No	VpcConfig	VPC configuration of the function

3. Output Parameters

Parameter name	Type	Description
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.Environment	Wrong Environment parameter passed in.
InvalidParameterValue.FunctionName	Function does not exist.
InvalidParameterValue.Handler	Wrong Handler parameter passed in.
LimitExceeded.Memory	Memory exceeds the upper limit.
LimitExceeded.Timeout	Timeout exceeds the upper limit.
ResourceNotFound.FunctionName	Function does not exist.

Error Code	Description
UnauthorizedOperation.CAM	CAM authentication failed.

5. Examples

Example 1. Updating Function Configuration

Input Example

```
https://scf.tencentcloudapi.com/?Action=UpdateFunctionConfiguration
&FunctionName=<FunctionName>
&Description=<Description>
&<Common request parameter>
```

Output Example

```
{
"Response": {
"RequestId": "eac6b301-a322-493a-8e36-83b295459397"
}
}
```

6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

Trigger APIs
CreateTrigger

Last updated：2018-09-28 17:01:36

1. API Description

API request domain name: scf.tencentcloudapi.com.

The API is used to create a new trigger based on the input parameters.

API request frequency limit: 100 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: CreateTrigger
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	Yes	String	Name of the function bound with the new trigger
TriggerName	Yes	String	Name of the new trigger. For a timer trigger, the name supports up to 100 characters including English letters, numbers, dashes and underscores; for other triggers, see the descriptions of parameters bound with the specific trigger
Type	Yes	String	Type of the trigger; currently, cos, cmq, timer and ckafka types are supported
TriggerDesc	No	String	The parameter corresponding to the trigger. For a timer trigger, its content is a Linux cron expression; for other triggers, see the description of the specific trigger

Parameter name	Required	Type	Description
Qualifier	No	String	Version of the function

3. Output Parameters

Parameter name	Type	Description
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.Cdn	Wrong CDN parameter passed in.
InvalidParameterValue.Ckafka	Wrong CKafka parameter passed in.
InvalidParameterValue.Cos	Wrong COS parameter passed in.
InvalidParameterValue.TriggerDesc	Wrong TriggerDesc parameter passed in.
InvalidParameterValue.TriggerName	Wrong TriggerName parameter passed in.
InvalidParameterValue.Type	Wrong Type parameter passed in.
LimitExceeded.Cdn	CDN usage exceeds the upper limit.
LimitExceeded.FunctionOnTopic	The number of functions under the same topic exceeds the upper limit.
LimitExceeded.Trigger	The number of triggers exceeds the upper limit.

Error Code	Description
ResourceInUse.Cdn	CDN is in use.
ResourceInUse.Cmq	CMQ is in use.
ResourceInUse.Cos	COS is in use.
ResourceNotFound.Cdn	CDN does not exist.
ResourceNotFound.Ckafka	CKafka does not exist.
ResourceNotFound.Cmq	CMQ does not exist.
ResourceNotFound.Cos	COS does not exist.
ResourceNotFound.FunctionName	Function does not exist.
ResourceNotFound.FunctionVersion	Function version does not exist.
UnauthorizedOperation.CAM	CAM authentication failed.
UnsupportedOperation.Cdn	CDN is not supported.
UnsupportedOperation.Cos	COS operation is not supported.
UnsupportedOperation.Trigger	Trigger operation is not supported.

5. Examples

Example 1. Creating New Trigger

Input

```
https://scf.tencentcloudapi.com/?Action=CreateTrigger
&FunctionName=<FunctionName>
&TriggleName=<TriggerName>
&Type=timer
&TriggerDesc=*/2****
&<Common request parameter>
```

Output

```
{
"Response": {
```

```
"RequestId": "eac6b301-a322-493a-8e36-83b295459397"
}
}
```


6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

DeleteTrigger

Last updated：2018-09-28 17:01:36

1. API Description

API request domain name: scf.tencentcloudapi.com.

The API is used to delete the existing trigger based on the parameters passed in.

API request frequency limit: 100 times/second.

2. Input Parameters

The following list of request parameters lists only the API request parameters and some common parameters. For the complete list of common parameters, see Common Request Parameters.

Parameter name	Required	Type	Description
Action	Yes	String	Common parameter; the value for this API: DeleteTrigger
Version	Yes	String	Common parameter; the value for this API: 2018-04-16
Region	Yes	String	Common parameters; for details, see the Region List supported by the product.
FunctionName	Yes	String	Name of the function
TriggerName	Yes	String	Name of the trigger to be deleted
Type	Yes	String	Type of the trigger to be deleted. The value can be `cos`, `cmq`, `timer` and `ckafka`
TriggerDesc	No	String	If the to-be-deleted trigger is a COS trigger, this field is required and holds the data {"event":"cos:ObjectCreated:*"} in JSON format, and the data content is in the same format as this field in the SetTrigger API; if the to-be-deleted trigger is a timer or CMQ trigger, this field is optional
Qualifier	No	String	Version information of the function

3. Output Parameters

Parameter name	Type	Description
RequestId	String	The unique request ID which is returned for each request. The RequestId for the current request needs to be provided when troubleshooting.

4. Error Codes

Only the error codes related to the API logic are listed below. For other error codes, see Common Error Codes.

Error Code	Description
InternalError	Internal error
InternalError.System	Internal system error.
InvalidParameterValue	Wrong parameter value
InvalidParameterValue.Cdn	Wrong CDN parameter passed in.
InvalidParameterValue.Cmq	Wrong CMQ parameter passed in.
InvalidParameterValue.Cos	Wrong COS parameter passed in.
ResourceInUse.Cdn	The CDN resource is in use.
ResourceInUse.Cmq	The CMQ resource is in use.
ResourceNotFound.Cdn	CDN resource does not exist.
ResourceNotFound.Cmq	CMQ resource does not exist.
ResourceNotFound.FunctionName	Function does not exist.
UnauthorizedOperation.CAM	CAM authentication failed.
UnsupportedOperation.Cdn	CDN is not supported.

5. Examples

Example 1. Deleting Trigger

Input

```
https://scf.tencentcloudapi.com/?Action=DeleteTrigger
&FunctionName=ledDummyAPITest
&TriggerName=test3
&Type=timer
&<Common request parameter>
```

Output

```
{
"Response": {
"RequestId": "eac6b301-a322-493a-8e36-83b295459397"
}
}
```

6. Other Resources

Tencent Cloud API 3.0 comes with a set of complementary development tools that make it easier to call the API.

- Tencent Cloud SDK 3.0 for Python
- Tencent Cloud SDK 3.0 for Java
- Tencent Cloud SDK 3.0 for PHP
- Tencent Cloud SDK 3.0 for Go
- Tencent Cloud SDK 3.0 for NodeJS
- Tencent Cloud SDK 3.0 for .NET
- Tencent Cloud CLI 3.0

Data Types

Last updated：2018-09-28 17:01:37

Code

Function code

Referenced by the following API: CreateFunction.

Name	Type	Required	Description
CosBucketName	String	No	Name of the object bucket
CosObjectName	String	No	Path of the COS object
ZipFile	String	No	This contains a .zip file of the function code file and its dependencies. When using this API, the content of the .zip file needs to be encoded with Base64. It can be up to 20MB
CosBucketRegion	String	No	Region of the COS. For the Beijing region, you need to pass in ap-beijing; for the Beijing Region One, you need to pass in ap-beijing-1; for other regions, this parameter can be left blank

Environment

Environment variable of the function

Referenced by the following APIs: CreateFunction, GetFunction, UpdateFunctionConfiguration.

Name	Type	Required	Description
Variables	Array of Variable	No	Environment variable array

Filter

Log filtering conditions for distinguishing between logs for successes and logs for errors

Referenced by the following API: GetFunctionLogs.

Name	Type	Required	Description
RetCode	String	No	filter.retCode=not0 indicates that only the logs for errors are returned, while filter.retCode=is0 indicates that only the logs for successes are returned; if this parameter is not passed in, all logs are returned

Function

Function list

Referenced by the following API: ListFunctions.

Name	Type	Description
ModTime	String	Modified time
AddTime	String	Created time
Runtime	String	Runtime
FunctionName	String	Name of the function
FunctionId	String	Function ID
Namespace	String	Namespace

FunctionLog

Log information

Referenced by the following API: GetFunctionLogs.

Name	Type	Description
FunctionName	String	Name of the function
RetMsg	String	Return value after function execution
RequestId	String	The requestId that executes this function
StartTime	Timestamp	Time point when the function starts executing

Name	Type	Description
RetCode	Integer	Function execution result; 0 for success and other values for failure
InvokeFinished	Integer	This indicates whether the function call is ended; 1 for ended call and other values for exceptional call
Duration	Float	Duration of function execution in ms
BillDuration	Integer	Duration for function billing in ms, rounded upwards to the nearest 100 ms
MemUsage	Integer	The actual memory size consumed by function execution in bytes
Log	String	Log output during function execution

Result

Return of the executed function

Referenced by the following API: Invoke.

Name	Type	Description
Log	String	Log output during function execution; null for async call
RetMsg	String	This indicates the return of the executed function; null for async call
ErrMsg	String	This indicates the error message returned for the executed function; null for async call
MemUsage	Integer	The memory size consumed by function execution in bytes; null for async call
Duration	Float	This indicates the duration of function execution in ms; null for async call
BillDuration	Integer	This indicates the duration for function billing in ms; null for async call
FunctionRequestId	String	ID of this function execution
InvokeResult	Integer	0 for successes; null for async call

Trigger

Trigger type

Referenced by the following API: GetFunction.

Name	Type	Description
ModTime	Timestamp	Last modified time of the trigger
Type	String	Type of the trigger
TriggerDesc	String	Detailed configuration of the trigger
TriggerName	String	Name of the trigger
AddTime	Timestamp	Created time of the trigger

Variable

Variable parameters

Referenced by the following APIs: CreateFunction, GetFunction, UpdateFunctionConfiguration.

Name	Type	Required	Description
Key	String	No	Name of the variable
Value	String	No	Value of the variable

VpcConfig

VPC parameter configuration

Referenced by the following APIs: CreateFunction, GetFunction, UpdateFunctionConfiguration.

Name	Type	Required	Description
VpcId	String	No	ID of the VPC
SubnetId	String	No	ID of the subnet

SCF API 2017
API Overview

Last updated：2018-08-16 13:17:01

Welcome to Tencent Cloud Serverless Cloud Function. Serverless Cloud Function hosts user code, uses user-configured triggers, and automatically executes user applications based on triggering events, providing users with FaaS (Function as a Service).

Serverless Cloud Function is a serverless execution environment provided by Tencent Cloud. You can simply make your code run in a secure and elastic way on Tencent Cloud infrastructure by writing key code in the language supported by the platform and setting the running conditions without purchasing and managing servers. Tencent Cloud completely manages the underlying computing resources, such as server CPU, memory, network, and performs other configuration/resource maintenance, code deployment, auto scaling, load balance and so on. Code is run on demand, with no service fee charged when idle.

Glossary

Term	Full Name	Description
SCF	Serverless Cloud Function	Serverless Cloud Function hosts user code, uses user-configured triggers, and automatically executes user applications based on triggering events, providing users with FaaS (Function as a Service).
Trigger	SCF Trigger	The SCF cloud function is triggered by an event. The event source is a trigger, which is usually the resource object of other products, such as COS Bucket, CMQ Topic queue, timer.

How to Use

To use Serverless Cloud Function via API, you need to complete the configuration as follows:

1. Create a function

You can use the API CreateFunction to create a cloud function. The function name is the unique ID of the function and cannot conflict with others in the same region.

2. Create a function trigger

After the function is created, you can set the function to respond to a triggering event by using the API SetTrigger.

3. Test a function

You can call the function directly using the API InvokeFunction, or manipulate the trigger object to generate an event and use the API GetFunctionLogs to verify whether the function executes properly. After the above three steps are completed, your function can be put in use.

API Overview

Last updated：2018-08-16 13:17:21

Function-related APIs

Feature	Action ID	Description
Obtain Function List	ListFunctions	This API is used to obtain the function list in the current region.
Obtain Function Details	GetFunction	This API is used to obtain function details.
Create Function	CreateFunction	This API is used to create functions.
Delete Function	DeleteFunction	This API is used to delete a specified function.
Update Function	UpdateFunction	This API is used to update the configuration information or the code of a specified function.
Obtain Function Operation Log	GetFunctionLogs	This API is used to obtain the operation log of a specified function.
Obtain Function Monitoring Data	GetMonitorData	This API is used to obtain monitoring data of a specified function.
Run Function	InvokeFunction	This API is used to run a specified function.

Trigger-related APIs

Feature	Action ID	Description
Set Function Trigger	SetTrigger	This API is used to set a trigger for a specified function.
Delete Function Trigger	DeleteTrigger	This API is used to delete a trigger for a specified function.

Update History

Last updated：2018-09-26 17:52:34

The Second Release

Release time: 19:15:23, July 26, 2018

This release contains:

Improvement to existing documentation.

New interfaces:

- CreateFunction
- CreateTrigger
- DeleteFunction
- DeleteTrigger
- GetFunction
- GetFunctionLogs
- ListFunctions
- UpdateFunctionCode
- UpdateFunctionConfiguration

Modified interfaces:

- Invoke
  - New input parameters: Namespace

New data structures:

- Code
- Environment
- Filter
- Function
- FunctionLog
- Trigger
- Variable
- VpcConfig

The First Release

Release time: 15:58:40, May 17, 2018

This release contains:

Improvement to existing documentation.

New interfaces:

- Invoke

New data structures:

- Result

Calling Mode
Request Body
Introduction of Request Structure

Last updated：2018-08-16 13:17:43

A Tencent Cloud API call is completed by sending a request towards Tencent Cloud API server address while adding corresponding request parameters into the request according to API description. A Tencent Cloud API request is composed of server address, communication protocol, request method, request parameters and character encoding. Details are described below:

Service Address

The service connection address of Tencent Cloud APIs depends on the modules. For more information, please see the descriptions of each API.

Communication Protocol

Most Tencent Cloud APIs communicate over HTTPS to provide high-security channels.

Request Method

Tencent Cloud APIs support both POST and GET requests.

Note:

1. POST and GET requests cannot be used together. If you use GET method, parameters are obtained from Querystring.If you use POST method, parameters are obtained from Request Body, in which case parameters in Querystring will be ignored. The parameter formats in these request methods are the same. We use GET requests generally. But it is recommended to use POST if the parameter strings are too long.
2. If GET method is used, all request parameters need to be encoded with URL encoding. This is not required if POST method is used.

3. The maximum length for GET requests depends on browsers and different server configurations. For example, the length limit is 2K for traditional IE browser, 8K for Firefox. For long API requests with a large number of parameters, it is recommended to use POST method to avoid request failure due to exceeded string length.
4. For POST requests, you need to pass parameters in the form of `x-www-form-urlencoded`, since the API on the cloud obtains request parameters from $_POST.

Request Parameters

Two types of parameters are required for each Tencent Cloud API request: common request parameters and API request parameters. Common request parameters are required for all APIs (for more information, please see Common Request Parameters section), while API request parameters are specific to each API. For more information, please see the "Request Parameters" description of each API.

Character Encoding

All requests for Tencent Cloud APIs and their returned results are encoded using UTF-8 character set.

Common Request Parameters

Last updated：2018-08-16 13:17:53

A complete Tencent Cloud API request requires two types of request parameters: common request parameters and API request parameters. This document describes 6 common request parameters used in Tencent Cloud API requests. For more information about API request parameters, please see API Request Parameters.

Common request parameters are required in every API. When developers use Tencent Cloud APIs to send requests, they should make sure that the requests carry these common request parameters. Otherwise, the requests will fail. The initial letter of each common request parameter is in uppercase so that it can be differentiated from API request parameters.

Common request parameters are as follows:

Note:
This document illustrates APIs specific to Tencent Cloud CVMs. For APIs specific to other Tencent Cloud products, please see the relevant documents.

Parameter Name	Description	Type	Required
Action	The name of the API for the desired operation. For example, when a Tencent Cloud CVM user calls the API Query Instance List, the Action parameter is DescribeInstances.	String	Yes
Region	Region parameter, which is used to identify the region to which the instance you want to work with belongs. For more information, please see Regions and Availability Zones, or use the API Query Region List. **Note:** 1. Unless otherwise specified in the API document, this parameter is required generally. 2. Some of the regions are under internal trial and only available to certain users.	String	No
Timestamp	The current UNIX timestamp that records the time at which the API request was initiated.	UInt	Yes
Nonce	A random positive integer that is used in conjunction with Timestamp to prevent replay attacks.	UInt	Yes

Parameter Name	Description	Type	Required
SecretId	The SecretId that is applied for under Cloud API Key to identify identity. A SecretId corresponds to a unique SecretKey, which is used to generate the request Signature. For more information, please see Signature Method.	String	Yes
Signature	Request signature, which is used to verify the validity of the request. The signature must be computed based on input parameters. For more information, please see Signature Method.	String	Yes
SignatureMethod	Signature method. Supported methods include HmacSHA256 and HmacSHA1. The HmacSHA256 method is used to verify signatures only when the parameter is specified as HmacSHA256. Otherwise, HmacSHA1 is used. For more information, please see Signature Method.	String	No
Token	The token used for the temporary certificate, which must be used together with a temporary key. No token is required for a long-term key.	String	No

Use Case

The following example shows how common request parameters look like in an API request link for a Tencent Cloud product. If, for example, you want to query the list of Tencent Cloud CVM instances in the Guangzhou region, the request link should look like this:

```
https://cvm.api.qcloud.com/v2/index.php?
Action=DescribeInstances
&SecretId=xxxxxxx
&Region=ap-guangzhou
&Timestamp=1465055529
&Nonce=59485
&Signature=mysignature
&SignatureMethod=HmacSHA256
&<API request parameters>
```

API-specific Request Parameters

Last updated：2018-08-16 13:18:02

A complete Tencent Cloud API request requires two types of request parameters: common request parameter and API request parameter. This document will describe API request parameters used in Tencent Cloud API requests. For more information about common request parameters, please see Common Request Parameters.
API request parameters depend on specific APIs. Different APIs support different API request parameters. In order to differentiate from common request parameters, the initial letters of API request parameters are all lowercases.

Note:
The parameters in this document use Tencent Cloud CVM as example. For actual parameters for other Tencent Cloud products, refer to their API parameter instructions accordingly.

The following parameter list uses the Tencent Cloud CVM API DescribeInstances as example. This API supports the following API request parameters:

Parameter	Description	Type	Required
instanceIds.n	Array of IDs of CVMs to be queried. The array subscript starts from 0. Both instanceId and unInstanceId are supported, but it is recommended to use unified resource ID: unInstanceId.	String	No
lanIps.n	Array of private IPs of the CVMs to be queried.	String	No
searchWord	CVM alias set by the user.	String	No
offset	Offset. Default is 0.	Int	No
limit	The maximum number of servers allowed to be queried at a time. Default is 20, and the maximum is 100.	Int	No
status	Status of the CVM to be queried.	Int	No
projectId	Project ID. CVM instances of all projects are queried if this is left empty. 0 indicates default project. Call the API DescribeProject to look for the IDs of other projects.	String	No
simplify	Obtain non-real time data if simplify=1 is added when passing parameters.	Int	No

Parameter	Description	Type	Required
zoneId	Availability zone ID. CVM instances of all availability zones are queried if this is left empty. Call the API DescribeAvailabilityZones to look for the IDs of other availability zones.	Int	No

The fields are described as below:

Parameter name: Request parameter name supported by the API. Users can use this as API request parameter when using this API. A parameter name which ends with ".n" indicates that the parameter is an array, and array parameters need to be passed when using this parameter.
Required: Indicate whether this parameter is required. "Yes" means the parameter is required when you call the API. "No" means the parameter can be left empty.
Type: Data type of the API parameter.
Description: A brief description of the API request parameter.

Example

The format of API request parameters in API request links for Tencent Cloud products are shown below. Take Tencent Cloud CVM as example, suppose a user needs to query the list of scaling groups, the request link should be:

```
https://cvm.api.qcloud.com/v2/index.php?
&<Common request parameters>
&instanceIds.0=ins-0hm4gvho
&instanceIds.1=ins-8oby8q00
&offset=0
&limit=20
&status=2
&zoneId=100003
```

Final Request Format

Last updated：2018-08-16 13:18:11

Construction Rule

The construction rule for Tencent Cloud API request URLs:

https:// + request domain name + request path + ? +final request parameter string

Component description:

- **Request domain name:** The request domain name is determined by the product or the product module to which the API belongs. This domain name is different for different products or product modules. For example, the request domain name for Tencent Cloud CVM API for querying instance lists (DescribeInstances) is: `cvm.api.qcloud.com` . For more information of product request domain names, please see the description for each API.
- **Request path:** This is the request path for the corresponding Tencent Cloud API product. Each product usually corresponds to one fixed path. For example, the request path for Tencent Cloud CVM is always `/v2/index.php` .
- **Final request parameter string:** The API request parameter string includes common request parameters and API request parameters.

Example

The format of a final request URL for Tencent Cloud API is as follows:
Take the Tencent Cloud CVM API DescribeInstances as example, the first 6 parameters are common request parameters, while the last 6 ones are API request parameters.

```
https://cvm.api.qcloud.com/v2/index.php?
Action=DescribeInstances
&SecretId=xxxxxxx
&Region=gz
&Timestamp=1465055529
&Nonce=59485
&Signature=mysignature //Common request parameters
&instanceIds.0=ins-0hm4gvho
&instanceIds.1=ins-8oby8q00
&offset=0
&limit=20
```

```
&status=2
&zoneId=100003 //API request parameters
```

Returning Results
正确返回结果

Last updated：2018-08-16 13:18:21

If the API is successfully called, the error code is 0, the error message is empty, and the returned result data is displayed.
Example:

```
{
"code": 0,
"message": "",
<Returned result data>
}
```

错误返回结果

Last updated：2018-08-16 13:18:30

If the API call fails, the error code is not 0, and the message field displays error details. You can query specific error information based on the codes and message fields on the Error codes page.
Example of a returned error:

```
{
"code": 5100,
"message": "(100004) projectId is incorrect",
}
```

Error Codes

Last updated：2018-08-16 13:18:42

Error Codes

"message" field indicates errors related to modules.
Example:
"message": "(100004) incorrect projectId"
It consists of two parts - the string within () indicates the module error code, and the string following () is the error description.
Different modules may produce different errors. You can identify the cause of error based on error description.The common error codes are listed in the following table. Other error codes not listed here can be found in the specific API description of API document.

Common Error Codes

Error Code	Module Error Code	Error Message	Description
4000	10000	invalid request parameters	Request parameter is invalid. This is generally because the parameter is not specified according to the CMQ Cloud API description.
4000	10010	lacked of required parameters	Required parameter is missing. Find the missing required parameters by referring to the CMQ Cloud API description.
4100	10030	authentication failed	Authentication failed. The possible reasons are: (1) The secret key used in the generation of signature algorithm is incorrect; (2) Signature algorithm is incorrect (the signature algorithm implemented by users is used instead of official SDK).
4300	10040	charge overdue	Account is in arrears.
6000	10050	server internal error	Server internal error. Generally, requestId is returned to users, based on which users can submit a ticket to the technical support staff to locate the problem.

4100	10080	secret id status error	"secret id" status error. "secret id" may be disabled.
4000	10110	request parameters error	Incorrect request parameter. This is generally because the parameter is not specified according to the CMQ Cloud API description.
4420	10250	qps throttling	QPS is limited because it has reached the upper limit. To ensure backend stability and calling latency, the maximum QPS is 5k. Users with special requirements can submit a ticket to apply for more quota.
4100	10270	secret id is not existed	The secret id does not exist.
4000	10280	action is not existed	The parameter Action does not exist.
4000	10310	error: parameter key format error"	Incorrect parameter format.
4000	10320	no such parameter	Such parameter does not exist.
4000	10330	parameter is NOT a repeatable parameter	The parameter is unique. In a batch API, such as BatchSendMessage, the parameter msgBody is repeatable. The parameter receiptHandle in BatchDeleteMessage is also repeatable. An error may occur if any parameter other than those specified to be repeatable in each API is passed as a repeatable parameter.
4000	10350	parameter value or length is out of range	The parameter value (Integer) or parameter length (String) exceeds the limit.
4000	10360	parameter error type	Incorrect parameter type. For example, a string is specified instead of an integer.
4000	10370	parameter batch size is more than 16	The batch value of batch API exceeds 16.
4000	10380	parameter is not consequent	The subscripts of repeatable parameters are not in sequential order.

4000	10390	lacked of required parameter	Required parameter is missing.
4000	10400	cannot find parameter in uri	Parameter cannot be found in URI.
4000	10410	unexpected http method, only GET or POST is supported	HTTP method is not supported. Currently, only GET and POST methods are supported. POST is recommended due to a limit on the length of GET request.
4000	10420	cannot parse	HTTP message cannot be resolved.
4000	10430	action name is not existed	The API name specified by Action does not exist.
4000	10440	account illegal, it may be an assistant account	Invalid account. It may be an assistant account. KMS does not support operations on an assistant account.
4000	10450	secret id doesn't begin with AKID account	The secret id does not start with AKID.
4480	10460	exceed interface frequency limit, please slow down	To protect the backend system, the call frequency limit of control class APIs (such as CreateQueue, DeleteQueue) is much lower than QPS. Therefore, when this error occurs, reduce the call frequency of control class APIs.
4200	10461	no cam authorization	The sub-account is not authorized by root account to access this API or resource.

##SCF Error Code##

Error Code	Module Error Code	Error Message	Description
4400	9003	InvalidParameter	Parameter Error

5100	9000	SystemError	System Error
4102	9002	SecretidNotAuthAccessResource	Unauthorized access to the resource
4000	9305	InsufficientBalance	Insufficient Balance

签名方法

Last updated：2018-08-16 13:18:52

Tencent Cloud API authenticates each access request, so each request is required to include signature information in the common request parameters for user authentication. The signature is generated with user's security credentials, which consist of a SecretId and a SecretKey. If you don't have security credentials, apply for the credentials from the Tencent Cloud official website. Otherwise, you will not be able to call the cloud APIs.

Applying for Security Credentials

Before using Tencent Cloud's APIs for the first time, you need to apply for security credentials on **Tencent Cloud Console** -> **API Key Management**. Security credentials consist of a SecretId and a SecretKey, where:

- **SecretId**: Used to identify the API caller;
- **SecretKey:** Used for signature string encryption, and signature string verification by server.

> **Note:**
> API key is very important for building Tencent Cloud API requests. With Tencent Cloud APIs, you can work with all of your Tencent Cloud resources under your account. For the security of your property and services, keep the key well and change it regularly (after changing the key, be sure to delete the old one as soon as possible).

How to apply for security credentials:

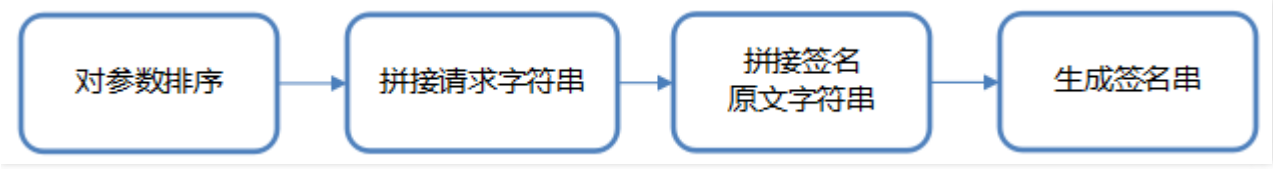
1. Log in to the Tencent Cloud Console.
2. Click **Cloud Products**, and select **Cloud API Key** under **Management Tools** to go to the cloud API key management page.

3. On the API Key Management page, click **New Key** to create a pair of SecretId/SecretKey.

> **Note:**
>
> - A developer account can have two pairs of SecretId/SecretKey at most.
> - The QQ accounts added as sub-users by a developer can apply for different security credentials on different developer consoles.
> - The security credentials of a sub-user can only be used to call some of cloud APIs.

Generating a Signature String

With the Secret ID and Secret Key, a signature string can be generated. The following is the process for generating a signature string:



Suppose that you have the following SecretId and SecretKey:
SecretId: AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA

SecretKey: Gu5t9xGARNpq86cd98joQYCN3Cozk1qA

Note:
This information is only for demonstration purpose. Make sure you proceed with your actual SecretId, SecretKey and request parameters.

For example, when you call Tencent Cloud CVM's API View Instance List (DescribeInstances), the request parameters are as follows:

Parameter Name	Description	Parameter Value
Action	Method name	DescribeInstances
SecretId	Key ID	AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA
Timestamp	Current timestamp	1465185768
Nonce	A random positive integer	11886
Region	The region where the instance resides	ap-guangzhou
SignatureMethod	Signature method	HmacSHA256
InstanceIds.0	ID of the instance to be queried	ins-09dx96dg

1. Sort parameters

First, sort all the request parameters in an ascending lexicographical order by their names. (This is like sorting words in a dictionary in an ascending alphabetical order or numerical order. That is to say, sort the parameters by their first letters, then by their second letters if their first letters are the same, and so on). You can complete the sorting process using relevant sorting functions in the programming language, such as the ksort function in PHP. The parameters in the example are sorted as follows:

```
{
"Action" : "DescribeInstances",
"Nonce" : 11886,
"Region" : "ap-guangzhou",
"SecretId" : "AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA",
"SignatureMethod" : "HmacSHA256",
"Timestamp" : 1465185768,
```

```
"InstanceIds.0" : "ins-09dx96dg"
}
```

Any other programming language can be used to sort these parameters as long as the same result is produced.

2. Generate a request string

This step is to generate a request string.
Format the request parameters sorted in the previous step as `"parameter name"="parameter value"` . For example, if the parameter value of `"Action"` is `"DescribeInstances"` , the resulting format is `Action=DescribeInstances` .

Note:

1. "Parameter value" is the original value, instead of the URL encoded value.
2. Any underscore in the Key of input parameter needs to be replaced with ".". But the underscore in Value does not. For example, `Placement_Zone=CN_GUANGZHOU` needs to be converted to `Placement.Zone=CN_GUANGZHOU` .

Then, join the formatted parameters together with `"&"` to generate the final request string (ignore the line breaks in the text):

```
Action=DescribeInstances
&InstanceIds.0=ins-09dx96dg
&Nonce=11886
&Region=ap-guangzhou
&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA
&SignatureMethod=HmacSHA256
&Timestamp=1465185768
```

3. Constructing the original signature string

This step is used to generate the original signature string.
The original signature string is constructed as follows:

Request method + Request CVM + Request path + ? + Request string

Description of parameters:

- **Request method:** The POST and GET methods are supported. In this case, a GET request is used. The methods must be in upper-case.
- **Request CVM:** The CVM domain name. The domain name for request varies with the product or product module to which the API belongs. For example, for Tencent Cloud CVM's API for querying instance list (DescribeInstances), the domain name for request is: `cvm.api.qcloud.com` . For more information on the domain names for requests in different products, please see the description of each API.
- **Request path:** This is the request path for the Tencent Cloud product corresponding to the API. Each product has a fixed path. For example, the request path for Tencent Cloud CVM is always `/v2/index.php` .
- **Request string:** The request string generated in the previous step.

The resulting original signature string in the above example is as follows (ignore the line breaks in the text):

```
GETcvm.api.qcloud.com/v2/index.php?Action=DescribeInstances
&InstanceIds.0=ins-09dx96dg
&Nonce=11886
&Region=ap-guangzhou
&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA
&SignatureMethod=HmacSHA256
&Timestamp=1465185768
```

4. Generating the signature string

This step is to generate the signature string.

> **Note:**
> You can compute a signature using two methods: HmacSHA256 and HmacSHA1. The signature string is generated with the signature algorithm you specified (parameter SignatureMethod). If you specify HmacSHA256 as SignatureMethod, the signature is computed using HmacSHA256. But in other cases, HmacSHA1 is used.

Sign the **original signature string** obtained in the previous step with signature algorithm (HmacSHA256 or HmacSHA1), and then encode the generated signature string with Base64 to obtain the final signature string.

In this example, PHP language is used and the signature algorithm is **HmacSHA256**. Therefore, the code for generating the signature string is as follows (If another programming language is used, the original

signature string in the above example can be used for verification, as long as the signature string generated is the same as the one in the example):

```
$secretKey = 'Gu5t9xGARNpq86cd98joQYCN3Cozk1qA';
$srcStr = 'GETcvm.api.qcloud.com/v2/index.php?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Nonce=11886&Region=ap-guangzhou&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA&SignatureMethod=HmacSHA256&Timestamp=1465185768';
$signStr = base64_encode(hash_hmac('sha256', $srcStr, $secretKey, true));
echo $signStr;
```

The resulting signature string is as follows:

```
0EEm/HtGRr/VJXTAD9tYMth1Bzm3lLHz5RCDv1GdM8s=
```

Similarly, if you specify **HmacSHA1** as the signature algorithm, the code for generating the signature string is as follows:

```
$secretKey = 'Gu5t9xGARNpq86cd98joQYCN3Cozk1qA';
$srcStr = 'GETcvm.api.qcloud.com/v2/index.php?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Nonce=11886&Region=ap-guangzhou&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3gnPhESA&SignatureMethod=HmacSHA1&Timestamp=1465185768';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

The resulting signature string is as follows:

```
nPVnY6njQmwQ8ciqbPl5Qe+Oru4=
```

Encoding the Signature String

The generated signature string cannot be directly used as the request parameter, and needs to be URL encoded.

For example, the signature string generated in the previous step `0EEm/HtGRr/VJXTAD9tYMth1Bzm3lLHz5RCDv1GdM8s=` is converted to `0EEm/HtGRr/VJXTAD9tYMth1Bzm3lLHz5RCDv1GdM8s=` after being encoded. The resulting request parameter for signature string (Signature) is `0EEm/HtGRr/VJXTAD9tYMth1Bzm3lLHz5RCDv1GdM8s=`, which will be used to generate the final request URL.

Note:
If GET method is used, all request parameters need to be URL encoded. In addition, some language libraries can encode URLs automatically. Repeated encoding will cause the failure of signature verification.

Failure of Authentication

The following errors may occur when the authentication fails:

Error Code	Error Type	Error Description
4100	Authentication failed	Authentication failed. Make sure the Signature in your request parameters is computed correctly as described in the above steps. Be sure to encode the Signature with URL encoding before initiating the request.
4101	No access to this API	The sub-user is not authorized by the developer to call the API. Contact the developer for authorization. For more information, please see Authorization Policy.
4102	No access to the resources operated in the API	The user is not authorized by the developer to access some resource in the requested resources. Check the message field for the ID of the resource the user does not have access to. Contact the developer for authorization. For more information, please see Authorization Policy.
4103	This API is unavailable for non-developer's SecretId	The sub-user with this SecretID cannot call this API. Only the developer has the access to this API.
4104	SecretId does not exist	The SecretId used for the signature does not exist, or the key status is incorrect. Make sure the API key is valid and enabled.
4110	Authentication failed	Permission verification failed. Make sure you have the access to this resource.
4500	Replay attack error	The parameter Nonce cannot be the same in two requests, and the difference between the Timestamp and Tencent server time should not be greater than 2 hours.

Function API
Acquire Function List

Last updated：2018-08-16 13:19:02

1. API Description

This API returns relevant function information according to the query parameters you passed.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters.
The Action field for this API is ListFunctions.

Parameter Name	Required	Type	Description
order	No	String	Whether the result is returned in ascending order or descending order. Available values: asc, desc.
orderby	No	String	This determines the field based on which the returned result is sorted. Available fields: addtime, modtime, functionname.
offset	No	Int	Data offset. Default: 0.
limit	No	Int	Length of returned data. Default: 20.
searchKey	No	String	Result is returned based on fuzzy match with the functionName field.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.

Parameter Name	Type	Description
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.
data	String	JSON data which contains function list information for the user.

The "data" field contains function list information for the user, where the "total" field indicates the total number of functions for the user, in the current region. The data structures for the functions are as follows:

Parameter Name	Type	Description
modTime	String	The last time when the function was modified.
functionName	String	Function name.
addTime	String	Function creation time.

4. Example

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=ListFunctions
&<Common request parameters>
&limit=2
&order=asc
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success",
"data": {
"functions": [
{
"modTime": "2017-05-04 19:40:38",
"addTime": "2017-05-04 19:40:38",
```

```
"functionName": "test6"
},
{
"modTime": "2017-05-02 09:39:49",
"addTime": "2017-05-02 09:39:49",
"functionName": "test4"
}
],
"total": 6
}
}
```

Acquire Function Details

Last updated：2018-08-16 13:19:14

1. API Description

This API is used to acquire the details of a certain function, including its name, code, handling method, associated triggers, timeout and other fields.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters. The Action field for this API is GetFunction.

Parameter Name	Required	Type	Description
functionName	Yes	String	Name of the function whose details are to be acquired.
code	No	Int	0: The code field is not included in the returned details. 1: The code field is included in the returned details. Default is 0.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.

Parameter Name	Type	Description
data	String	JSON data which contains function information, such as trigger name, runtime memorySize, entry point function, function description, version number, code size, code and so on.

Apart from certain fields such as function description upon creation, handler name and runtime memory size, the "data" field also contains the list of triggers that are associated with this function. Data structure for each trigger entry is as follows:

Parameter Name	Type	Description
modtime	String	The last time when the trigger was modified.
type	String	Trigger type. Two types are currently supported: cos and timer.
triggerDesc	String	Trigger parameter. For a timer trigger, this field is the cron expression; for a COS trigger, this field is the COS trigger event.
triggerName	String	Trigger name
addtime	String	The time when the trigger was bound to the function.

4. Example

When Code is not returned

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=GetFunction
&<Common request parameters>
&functionName=helljin89
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success",
"data": {
"modTime": "2017-05-18 16:38:38",
```

```
"functionName": "helljin89",
"triggers": [
{
"modTime": "2017-05-18 17:28:36",
"type": "timer",
"triggerDesc": {
"cron": "*/1 * * * *"
},
"triggerName": "abc",
"addTime": "2017-05-18 17:28:36"
}
],
"handler": "lambda_function.lambda_function",
"codeSize": 3225,
"memorySize": 128,
"version": "LATEST",
"timeout": 300,
"description": "abc\nefg"
}
}
```


When Code is returned

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=GetFunction
&<Common request parameters>
&functionName=helljin89
&code=1
```

Output:


```
{
"code": 0,
"message": "",
"codeDesc": "Success",
"data": {
"modTime": "2017-05-18 16:38:38",
"codeError": "",
"code": "import mymath.myadd as myadd\nimport mymath.mysub as mysub\nimport mymath.test.my
multiply as mymultiply\nimport json\nimport os\n\ndef lambda_function(event, context):\n a = even
t['a']\n b = event['b']\n print \"+\", myadd.myadd(a, b)\n print \"-\", mysub.mysub(a, b)\n print \"*\",
mymultiply.mymult(a, b)\n return \"hello\"\n\n",
"description": "abc\nefg",
"triggers": [
```

```
{
"modTime": "2017-05-18 17:28:36",
"type": "timer",
"triggerDesc": {
"cron": "*/1 * * * *"
},
"triggerName": "abc",
"addTime": "2017-05-18 17:28:36"
}
],
"handler": "lambda_function.lambda_function",
"codeSize": 3225,
"memorySize": 128,
"timeout": 300,
"version": "LATEST",
"codeResult": "success",
"functionName": "helljin89"
}
}
```

Create Function

Last updated：2018-08-16 13:19:23

1. API Description

This API is used to create functions. You must specify function name, code and handling method when calling this API. The other parameters (function description, runtime memory size, time out and so on) are optional.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters. The Action field for this API is CreateFunction.

Parameter Name	Required	Type	Description
functionName	Yes	String	Name of the created Lambda function, which can only include letters (upper case or lower case), numbers, en dashes (-) and underscores. The name must start with a letter and cannot end with en dash or underscore. Length: 2-60 characters.
code	Yes	String	This includes the function and zip files containing the dependent elements. When using the API, you need to encode the content of the zip files using base64 and prefix the encoded characters with `@` . Max size: `5 MB` . Note: this string will not be used for authentication.
handler	Yes	String	Name of function handling method. Format: ".", file name and function name are separated with ".". File name and function name may contain letters, numbers, underscores and en dashes (-) but must start and end with letters. Length for file name and function name: 2-60 characters.

Parameter Name	Required	Type	Description
description	No	String	Function description, which can contain a maximum of 1,000 characters (letters, numbers, spaces, commas, line breaks and periods).
memorySize	No	Int	Memory size when the function runs. Default: 128 MB. Available range: 128 MB - 1536 MB
timeout	No	Int	Maximum function operation time (in seconds). Available range: 1 - 300 seconds. Default is 3 seconds.

Note: an account can have at most 20 Lambda functions in a region, and each Lambda function can have up to 2 timer triggers and 2 COS triggers.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed. For more information, please see Common Error Codes in the Error Codes page.
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.

4. Example

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=CreateFunction
&<Common request parameters>
&functionName=myFunction
&code=Corresponding string when the zip file containing the code is encoded using base64
&hander=lambda_function.lambda_handler
&description=helloWorld Lambda function
&memorySize=128
&timeout=2
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success"
}
```

Delete Function

Last updated：2018-08-16 13:19:32

1. API Description

This API is used to delete function, along with the triggers that are bound to this function.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters.
The Action field for this API is DeleteFunction.

Parameter Name	Required	Type	Description
functionName	Yes	String	Name of the function to be deleted.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.

4. Example

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=DeleteFunction
&<Common request parameters>
```

```
&functionName=hell
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success"
}
```

Update Function

Last updated：2018-08-16 13:19:41

1. API Description

This API is used to update relevant fields of a function, including code, handling method, description, runtime memory size, time out and so on.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters. The Action field for this API is UpdateFunction.

Parameter Name	Required	Type	Description
functionName	Yes	String	Name of the function to be modified.
code	No	String	This includes the code files for the function and zip files containing the dependent elements. When using the API, you need to encode the content of the zip files using base64 and prefix the encoded characters with `@`. Max size: `5 MB`. Note: this string will not be used for authentication.
handler	No	String	Name of function handling method. Format: ".", file name and function name are separated with ".". File name and function name may contain letters, numbers, underscores and en dashes (-) but must start and end with letters. Length for file name and function name: 2-60 characters.
description	No	String	Function description. Description can contain a maximum of 1,000 characters (letters, numbers, spaces, commas and periods).
memorySize	No	Int	Memory size when the function runs. Default: 128 MB. Available range: 128 MB - 1536 MB

Parameter Name	Required	Type	Description
timeout	No	Int	Maximum function operation time (in seconds). Available range: 1 - 300 seconds. Default is 3 seconds.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.

4. Example

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=UpdateFunction
&<Common request parameters>
&functionName=hell
&memorySize=130
&timeout=2
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success"
}
```

Acquire Operation Log of Functions

Last updated：2018-08-16 13:19:50

1. API Description

This API is used to acquire function log according to the configured query conditions.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters. The Action field for this API is GetFunctionLogs.

Parameter Name	Required	Type	Description
functionName	No	String	Function name.
offset	No	Int	Data offset. Default: 0.
limit	No	Int	Length of returned data. Default: 20.
order	No	String	Whether the logs are sorted in ascending order or descending order. Available values: desc, asc.
orderby	No	String	This determines the field based on which the logs are sorted. Available fields: start_time, function_name, request_id, duration, mem_usage.
requestId	No	String	The generated request_id when the trigger is called.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.

Parameter Name	Type	Description
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.
total	Int	Total number of logs.
data	Array	List of log information returned.

The data structures of returned log entries are as follows:

Parameter Name	Type	Description
function_name	String	Name of the corresponding function for this log.
ret_msg	String	Returned value when function execution was finished.
request_id	String	Corresponding request_id when the function was executed.
start_time	String	The time when function execution started.
ret_code	Int	Function execution result. 0: Successful; other values: Failed.
duration	Float	Function execution time cost (in ms).
bill_duration	Int	Billable duration for the function, rounded up to the latest 100 ms based on "duration" (in ms).
mem_usage	Int	Actual memory used during function execution (in bytes).
log	String	Log output during function execution.

4. Example

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=GetFunctionLogs
&<Common request parameters>
&offset=1
&limit=2
```

```
&orderby=duration
&functionName=helljin89
```

Output


```
{
"code": 0,
"message": "",
"codeDesc": "Success",
"total": 189,
"data": [
{
"functionName": "helljin89",
"retMsg": "'module' object has no attribute 'lambda_handler'",
"requestId": "b684ef85-250a-11e7-839d-5254007d2563",
"startTime": "2017-04-19 22:16:00",
"retCode": 1,
"duration": 0.241,
"billDuration": 100,
"memUsage": 131072,
"log": ""
},
{
"functionName": "helljin89",
"retMsg": "'module' object has no attribute 'lambda_handler'",
"requestId": "b6842e1a-250a-11e7-aaa8-525400edfec1",
"startTime": "2017-04-19 22:16:00",
"retCode": 1,
"duration": 0.269,
"billDuration": 100,
"memUsage": 131072,
"log": ""
}
]
}
```

Obtaining Function Monitoring Data

Last updated：2018-08-16 13:20:04

API Description

Description

This API (GetMonitorData) is used to obtain monitoring data of serverless cloud functions. Monitoring data can be obtained based on the function name specified by the user.

Domain name for API

Domain name for API request: `monitor.api.qcloud.com`

Request

Syntax:

```
GET https://monitor.api.qcloud.com/v2/index.php?Action=GetMonitorData
&<Common request parameters>
&namespace=qce/scf
&metricName=duration
&dimensions.0.name=functionName
&dimensions.0.value=test
&startTime=2017-07-24 14:00:00
&endTime=2017-07-24 14:05:00
&period=300
```

Input Parameters

The following request parameter list only provides the API request parameters. Common request parameters are required when the API is called. For more information, see Common Request Parameters. The Action field for this API is GetMonitorData.

Parameter	Description	Type	Required
namespace	Namespace. Every Tencent Cloud product has a namespace. The namespace of the serverless cloud function is `qce/scf` .	String	Yes

Parameter	Description	Type	Required
metricName	Metric name. The specific monitor metrics to be obtained, such as duration for the run time, and invocation for the number of calls	String	Yes
dimensions.n.name	Dimension name. The serverless cloud function has only one dimension: functionName, which is used in combination with dimensions.n.value.	String	Yes
dimensions.n.value	The corresponding value of a dimension name.	String	Yes
startTime	Start time, such as "2017-01-01 00:00:00". Default is "00:00:00" of the current day.	Datetime	No
endTime	End time, such as "2017-01-01 10:00:00". Default is the current time. **Note: startTime must be earlier than endTime. It is recommended that endTime and startTime be on the same day.	Datetime	No
period	Interval for collecting monitoring data. The supported granularities are 60s and 300s. Default is 300s if it is not specified.	Int	No

`metricName` list for the serverless cloud function is as follows:

Metric name	Description	Unit
duration	Run time	Milliseconds (ms)
invocation	The number of calls	-
error	The number of error calls	-

The serverless cloud function only supports the dimension of cloud function name, as described below: This dimension reflects the monitoring metric of a cloud function. The dimension (dimensions.n.name) to be specified is as follows:

Dimension	Description	Format
functionName	The name of a serverless cloud function	string

If you want to query the monitoring data of a cloud function named "test" in a specified run time (duration), see the following API request example:

```
GET https://monitor.api.qcloud.com/v2/index.php?Action=GetMonitorData
&<Common request parameters>
&namespace=qce/scf
&metricName=duration
&dimensions.0.name=functionName
&dimensions.0.value=test
```

Response

Response Example:

```
{
"code": 0,
"message": "",
"metricName": "duration",
"startTime": "2017-07-24 14:00:00",
"endTime": "2017-07-24 14:05:00",
"period": 300,
"dataPoints": [
0.65213,
0.5586
]
}
```

Response Parameters

Parameter	Description	Type
code	Common error code. 0: Successful; other values: Failed. For more information, please see Common Error Codes on the Error Codes page.	Int
codeDesc	Description	String
message	Error message details	String
startTime	Start time	Datetime

Parameter	Description	Type
endTime	End Time	Datetime
metricName	Metric name	String
period	Interval for collecting monitoring data	Int
dataPoints	Monitoring data list. Each element of the array stands for the data read at the monitoring time point.	Object

Practical Case

Request

```
GET https://monitor.api.qcloud.com/v2/index.php?
Action=GetMonitorData
&SecretId=AKIDutrojKI3CKQZNAr763UXks05898Lmciu
&Nonce=62089
&Timestamp=1505804102
&Region=gz
&Signature=LWI+dudHPe56OYcsvKWiXImdO5s=
&namespace=qce/scf
&metricName=duration
&dimensions.0.name=functionName
&dimensions.0.value=test5
&startTime=2017-09-19+12:00:00
&endTime=2017-09-19+14:05:00
&period=300
```

Response

```
{
"code": 0,
"message": "",
"codeDesc": "Success",
"metricName": "duration",
"startTime": "2017-09-19 12:00:00",
"endTime": "2017-09-19 14:05:00",
"period": "300",
"dataPoints": [
2.983,
1.131,
```

```
0.732,
0.773,
0.921,
0.901,
0.775,
0.931,
0.812,
0.757,
1.06,
0.805,
0.834,
0.812,
0.881,
0.807,
0.831,
0.992,
0.793,
1.968,
0.868,
0.93,
6.365,
0.732,
0.836,
1.433
]
}
```

Run Function

Last updated：2018-08-16 13:20:13

1. API Description

This API is used to trigger and execute specified function. Two trigger methods are currently supported: synchronous and asynchronous. For synchronous calls, the returned values will contain an additional field `payLoad` which indicates the returned values of the function.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters.
The Action field for this API is InvokeFunction.

Parameter Name	Required	Type	Description
functionName	Yes	String	Name of the function to be executed.
invokeType	No	String	Trigger method. Available values: `RequestResponse` (synchronous) , `Event` (asynchronous) . Default is synchronous.
param	No	String	Function execution parameter, which is passed in JSON format. Max parameter length: `1 MB` .
logType	No	String	This field is specified when using synchronous call. Returned values will include `4K` log. Available values: `None` (default), `Tail` . When the value is `Tail` , the `logMsg` field in the returned parameters will contain the corresponding function execution log.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.
data	String	This field contains relevant information of the function execution (in JSON format).

Data structure of the "data" field is as follows:

Parameter Name	Type	Description
requestId	String	request_id of the function execution process.
payLoad	String	Returned values of the function execution process. This is only returned for synchronous calls.
logMsg	String	Log output during the function execution process. This is only returned for synchronous calls.
memSize	Int	Memory size during the function execution process (in bytes). This is only returned for synchronous calls.
duration	Float	Time cost for the function execution process (in milliseconds). This is only available for synchronous calls.

4. Example

Asynchronous Request

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=InvokeFunction
&<Common request parameters>
&functionName=hell
&param={"aa":3}
&invokeType=Event
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success",
"data": {
"requestId": "141c9b47-2b19-11e7-9a9b-5254007d2563",
}
}
```

Synchronous Request

Input:

```
https://scf.api.qcloud.com/v2/index.php?Action=InvokeFunction
&<Common request parameters>
&functionName=hell
&param={"aa":3}
&invokeType=RequestResponse
&logType=Tail
```

Output:

```
{
"code": 0,
"message": "",
"codeDesc": "Success",
"data": {
"duration": 0.422,
"memUsage": 126976,
"log": "log in func\n",
"requestId": "c471ac84-2f15-11e7-8501-5254007d2563",
"retMsg": "Hello from Lambda"
"billDuration": 100
}
}
```

Trigger API
Configure Function Trigger

Last updated：2018-08-16 13:20:25

1. API Description

This API is used to create or modify function trigger. Two trigger types are currently supported: cos and timer. Each function can have a maximum of 2 timer triggers and 2 COS triggers.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters. The Action field for this API is SetTrigger.

Parameter Name	Required	Type	Description
functionName	Yes	String	Name of the function to which the newly created trigger is bound.
triggerName	Yes	String	Name of the created trigger. For a timer trigger, the name can contain letters, numbers, en dashes (-) and underscores, up to 100 characters. For a COS trigger, the name must be in the format of `-..myqcloud.com
type	Yes	String	Trigger type. Two types are currently supported: cos and timer.
triggerDesc	Yes	String	Trigger parameter. For a timer trigger, this is Linux cron expression; for a COS trigger, this is JSON data `{"event": "cos:ObjectCreated:*"}` .
newTriggerName	No	String	This parameter is used when you update the name of a timer trigger. The new name may contain letters, numbers, en dashes (-) and underscores, up to 100 characters.

Note:

- An account can have at most 20 functions in a region, and each function can have up to 2 timer triggers and 2 COS triggers.
- A bucket can only be bound with triggers in the same region. You cannot bind bucket with triggers in different regions.
- When using a COS trigger, if you need to save function execution result to the bucket, it is recommended that you configure two different buckets, one for trigger source, one for the output result, in order to prevent function from running continuously due to circular dependency.
- Update is not supported for COS triggers.
- Name format for COS triggers should always be: `<bucketName>-<UID>.<Region>.myqcloud.com` .
- For COS triggers, you need to pass JSON data containing the following contents for triggerDesc:

```
{
"event": "cos:ObjectCreated:*"
}
```

Currently supported event values are listed in the table below. There are certain restrictions when you bind COS event to a specific trigger:

1. If you bind the `cos:ObjectCreated:\*` event first, all subsequent operations to bind events that start with `cos:ObjectCreated` will fail.
2. If you bind an event that starts with `cos:ObjectCreated` (except `cos:ObjectCreated:\*`), subsequent operations to bind the `cos:ObjectCreated:\*` event will fail.
3. If you bind the `cos:ObjectRemove:\*` event first, all subsequent operations to bind events that start with `cos:ObjectRemove` will fail.
4. If you bind an event that starts with `cos:ObjectRemove` (except `cos:ObjectRemove:\*`), subsequent operations to bind the `cos:ObjectRemove:\*` event will fail.
5. If COS is already bound, operations to create COS triggers will fail.
6. You cannot update the names for COS triggers.

Event	Description
cos:ObjectCreated:Put	Use the Put Object API to create file.
cos:ObjectCreated:Post	Use the Post Object API to create file.
cos:ObjectCreated:Copy	Use the Put Object - Copy API to create file.
cos:ObjectCreated:Append	Use the Append Object API to create file.

Event	Description
cos:ObjectCreated:CompleteMultipartUpload	Use the CompleteMultipartUploadt API to create file.
cos:ObjectCreated:*	Use "ObjectCreated" APIs mentioned above to create file.
cos:ObjectRemove:Delete	Use the Delete Object API to delete Object under a Bucket that doesn't have version management enabled, or use versionid to delete Object of specified version.
cos:ObjectRemove:DeleteMarkerCreated	Use the Delete Object API to delete Object under a Bucket that has version management enabled or paused.
cos:ObjectRemove:*	Use "ObjectRemove" APIs mentioned above to delete file.

3. Response Parameters

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.

4. Example

Configure Timer Trigger

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=SetTrigger
&<Common request parameters>
&functionName=hell
&triggerName=test1
```

```
&type=timer
&triggerDesc=*/2 * * * *
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success"
}
```

Configure COS Trigger

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=SetTrigger
&<Common request parameters>
&functionName=hell
&triggerName=lambdatest3-1251664966.cn-south.myqcloud.com
&type=cos
&triggerDesc={"event": "cos:ObjectCreated:Put"}
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success"
}
```

Delete Function Trigger

Last updated：2018-08-16 13:20:34

1. API Description

This API is used to delete trigger according to specified function name and trigger name.

Domain name for API access: scf.api.qcloud.com

2. Request Parameter

The following request parameter list only provides API request parameters. Common request parameters are also needed when the API is called. For more information, please see Common Request Parameters. The Action field for this API is DeleteTrigger.

Parameter Name	Required	Type	Description
functionName	Yes	String	Function name.
triggerName	Yes	String	Name of the trigger to be deleted. Name for a COS trigger should always be in the format of `<bucketName>-<UID>.<Region>.myqcloud.com`, while name for a timer trigger is a user-defined name.
type	Yes	String	Type of the trigger to be deleted. Two types are currently supported: cos and timer.
triggerDesc	Yes	String	This field is mandatory if the trigger to be deleted is a COS trigger, in which case this field shares the same format in `{"event":"cos:ObjectCreated:*"}` where JSON data is stored, in the data content and in the SetTrigger API. This field is optional if the trigger to be deleted is a timer trigger.

3. Response Parameters

Parameter Name	Type	Description

Parameter Name	Type	Description
code	Int	Common error code. 0: Successful; other values: Failed.
message	String	Module error message description depending on API
codeDesc	String	Error code. For a successful operation, "Success" will be returned. For a failed operation, a message describing the failure will be returned.

4. Example

Input

```
https://scf.api.qcloud.com/v2/index.php?Action=DeleteTrigger
&<Common request parameters>
&functionName=hell
&type=timer
&triggerName=test2
```

Output

```
{
"code": 0,
"message": "",
"codeDesc": "Success"
}
```