

Serverless Cloud Function

SDK Documentation

Product Documentation



Copyright Notice

©2013-2019 Tencent Cloud. All rights reserved.

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

Trademark Notice

 Tencent Cloud

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

Service Statement

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

Contents

SDK Documentation

Node.js SDK

PHP SDK

Python SDK

SDK Documentation

Node.js SDK

Last updated : 2020-07-07 17:07:06

Preparations for Development

Before installing the SDK for Node.js and using TencentCloud API for the first time, you need to apply for security credentials in the Tencent Cloud Console, which consists of `SecretId` and `SecretKey`. `SecretId` is used to identify the API requester, while `SecretKey` is a key used for signature string encryption and authentication by the server. Please keep your `SecretKey` private and do not disclose it to others.

Development environment

Node.js v8.9

Installation through npm

Installation through npm is the recommended way to use the SDK for Node.js. npm is a dependency manager for Node.js that supports the dependencies your project requires and installs them into your project. For more information, please visit [npm's official website](#).

1. Run the following installation command:

```
npm install tencentcloud-sdk-nodejs --save
```

2. Refer to the corresponding module code in your code. For more information, please see the sample.

Installation through source package

1. Go to the [GitHub code hosting page](#) to download the source code package.
2. Decompress the source package to an appropriate location in your project.
3. Import the corresponding module code into your code. For more information, please see the sample.

API List

API Name	Description
CreateFunction	Creates function
DeleteFunction	Deletes function
GetFunction	Gets function details
GetFunctionLogs	Gets function execution logs
Invoke	Executes function
ListFunctions	Gets function list
UpdateFunctionCode	Updates function code
UpdateFunctionConfiguration	Updates function configuration

Samples

```
'use strict';

const tencentcloud = require("/var/user/tencentcloud-sdk-nodejs");
const Credential = tencentcloud.common.Credential;

// Import the client models of the corresponding product module
const ScfClient = tencentcloud.scf.v20180416.Client;
const models = tencentcloud.scf.v20180416.Models;

exports.main_handler = (event, context, callback) => {
  console.log("Hello World")
  console.log(event)
  // console.log(context)
  callback(null, event);

  // Instantiate an authentication object. The Tencent Cloud account `secretId` and `secretKey` need to be passed in as the input parameters
  let cred = new Credential("AKIxxxxxxPDpqj3C", "75rxxxxxyJSODrMkx");

  // Instantiate the client object to request the product and the region where the function is located
  let client = new ScfClient(cred, "ap-shanghai");

  // Instantiate a request object to get the function list
```

```
console.log("Start ListFunctions")
let req = new models.ListFunctionsRequest();
// Call the API you want to access through the client object; you need to pass in the request object and the response callback function
client.ListFunctions(req, function(err, response) {
// The request returns an exception and the exception information is printed
if (err) {
console.log(err);
return;
}
// The request is returned normally, and the `response` object is printed
console.log(response.toJsonString());
});

// Instantiate a request object and call `invoke()`
console.log("Start Invoke")
let request = new models.InvokeRequest();
// API parameter. Enter the name of the function to be invoked, `RequestResponse` (sync), and `Event` (async)
let params = '{"FunctionName":"test_python", "InvocationType":"RequestResponse"}'
request.fromJsonString(params);
// Call the API you want to access through the client object; you need to pass in the request object and the response callback function
client.Invoke(request, function(err, response) {
// The request returns an exception and the exception information is printed
if (err) {
console.log(err);
return;
}
// The request is returned normally, and the `response` object is printed
console.log(response.toJsonString());
}, "test_python", "RequestResponse");

};
```

Packaging and Deployment

If you need to deploy a function in the SCF Console and use the SDK to invoke other functions, you need to package the `tencentcloud` library and function code together into a zip file.

- Please note that the execution method specified when the function is created in the console must correspond to the code file and execution function in the zip file.
- If the generated zip package is larger than 50 MB, it should be uploaded through COS.

- The default call rate limit for TencentCloud API is 20 calls per second. If you need to increase the limit for high concurrence, please [submit a ticket](#) for application.

Related Information

You can also use Tencent SCF SDK (Tencentserverless SDK), which integrates SCF business flow APIs to simplify the function invocation method and eliminates your need to encapsulate public TencentCloud APIs. For more information, please see [Calling SDK Across Functions](#).

PHP SDK

Last updated : 2020-06-24 16:22:13

Preparations for Development

Before installing the SDK for PHP and using TencentCloud API for the first time, you need to apply for security credentials in the Tencent Cloud Console, which consists of `SecretId` and `SecretKey`.

`SecretId` is used to identify the API requester, while `SecretKey` is a key used for signature string encryption and authentication by the server. Please keep your `SecretKey` private and do not disclose it to others.

Development environment

PHP 7.2

Installation through Composer (recommended)

Installation through Composer is the recommended way to use the SDK for PHP. Composer is a dependency manager for PHP that supports the dependencies your project requires and installs them into your project. For more information, please visit [Composer official website](#).

1. Install Composer:

- For Windows, go to [Composer official website](#) to download the installation package.
- For Unix, install it by running the following command on the command line.

```
curl -sS https://getcomposer.org/installer | php
```

2. Add dependencies to the `require` structure of `composer.json`. v3.0.6 is used here as an example, and you can view the latest version number in the Composer repository:

```
"tencentcloud/tencentcloud-sdk-php": "3.0.6"
```

3. Run the `composer install` command to download and install the SDK for PHP.

4. Add the following import code. For importing methods, please see the sample.

```
require 'vendor/autoload.php';
```


Installation through source package

1. Go to the [GitHub code hosting page](#) to download the source code package.
2. Decompress the source package to an appropriate location in your project.
3. Add the following import code. For importing methods, please see the sample.

```
require_once '../TCloudAutoLoader.php';
```

API List

API name	Description
CreateFunction	Creates function
DeleteFunction	Deletes function
GetFunction	Gets function details
GetFunctionLogs	Gets function execution logs
Invoke	Executes function
ListFunctions	Gets function list
UpdateFunctionCode	Updates function code
UpdateFunctionConfiguration	Updates function configuration

Samples

```
<?php
require_once '/var/user/tencentcloud-sdk-php/TCloudAutoLoader.php'; # Pay attention to the import
path
use TencentCloud\Common\Credential;
use TencentCloud\Common\Profile\ClientProfile;
use TencentCloud\Common\Profile\HttpProfile;
use TencentCloud\Common\Exception\TencentCloudSDKException;
use TencentCloud\Scf\V20180416\ScfClient;
use TencentCloud\Scf\V20180416\Models\InvokeRequest;
function main_handler($event, $context) {
    print "good";
    print "\n";
}
```

```
var_dump($event);
var_dump($context);
try {
    // Instantiate an authentication object. The Tencent Cloud account `secretId` and `secretKey` need to be passed in as the input parameters
    $cred = new Credential("your secretId", "your secretKey");
    $httpProfile = new HttpProfile();
    $httpProfile->setEndpoint("scf.tencentcloudapi.com");

    $clientProfile = new ClientProfile();
    $clientProfile->setHttpProfile($httpProfile);
    // Instantiate the client object to request the product and the region where the function is located
    $client = new ScfClient($cred, "ap-shanghai", $clientProfile);
    $req = new InvokeRequest();
    // API parameter. Enter the name of the function to be invoked, `RequestResponse` (sync), and `Event` (async)
    $params = '{"FunctionName":"test_python", "InvocationType":"RequestResponse"}';
    $req->fromJsonString($params);
    $resp = $client->invoke($req);
    print_r($resp->toJsonString());
}
catch(TencentCloudSDKException $e) {
    echo $e;
}
return "hello";
}
```

Packaging and Deployment

If you need to deploy a function in the SCF Console and use the SDK to invoke other functions, you need to package the `tencentcloud` library and function code together into a zip file.

- Please note that the execution method specified when the function is created in the console must correspond to the code file and execution function in the zip file.
- If the generated zip package is larger than 50 MB, it should be uploaded through COS.
- The default call rate limit for TencentCloud API is 20 calls per second. If you need to increase the limit for high concurrency, please [submit a ticket](#) for application.

Related Information

You can also use Tencent SCF SDK (Tencentserverless SDK), which integrates SCF business flow APIs to simplify the function invocation method and eliminates your need to encapsulate public TencentCloud APIs. For more information, please see [Calling SDK Across Functions](#).

Python SDK

Last updated : 2020-06-24 16:22:13

Preparations for Development

Before installing the SDK for Python and using TencentCloud API for the first time, you need to apply for security credentials in the Tencent Cloud Console, which consists of `SecretID` and `SecretKey`. `SecretID` is used to identify the API requester, while `SecretKey` is a key used for signature string encryption and authentication by the server. Please keep your `SecretKey` private and do not disclose it to others.

Development environment

Python v2.7 or v3.6

Installation through Pip (recommended)

You can install the TencentCloud API SDK for Python into your project through Pip. If you haven't installed Pip in your project environment yet, install it first as instructed at [Pip official website](#). To install through Pip, run the following command on the command line:

```
pip install tencentcloud-sdk-python
```

Installation through source package

Go to the [GitHub code hosting page](#) to download the latest code, decompress it, and run the following command:

```
$ cd tencentcloud-sdk-python
$ python setup.py install
```

API List

API name	Description
CreateFunction	Creates function
DeleteFunction	Deletes function

API name	Description
GetFunction	Gets function details
GetFunctionLogs	Gets function execution logs
Invoke	Executes function
ListFunctions	Gets function list
UpdateFunctionCode	Updates function code
UpdateFunctionConfiguration	Updates function configuration

Samples

```
# -*- coding: utf8 -*-
import json
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import TencentCloudSDKException
# Import the client models of the corresponding product module
from tencentcloud.scf.v20180416 import scf_client, models

# API name of the corresponding API
action = 'Invoke'

# API parameter. Enter the name of the function to be invoked, `RequestResponse` (sync), and `Event` (async)
action_params = {
    'FunctionName': "test",
    'InvocationType': "Event"
}

print('Start Hello World function')
```

```
def main_handler(event, context):
    try:
        # Instantiate an authentication object. The Tencent Cloud account `secretId` and `secretKey` need
        # to be passed in as the input parameters
        cred = credential.Credential("your secretId", "your secretKey")

        # Instantiate the client object to request the product and the region where the function is located
```

```
client = scf_client.ScfClient(cred, "ap-guangzhou")

# Call the API, initiate the request, and print the returned result
ret = client.call(action, action_params)

print(json.loads(ret)["Response"]["Result"]["RetMsg"])

except TencentCloudSDKException as err:
    print(err)
```

Packaging and Deployment

If you need to deploy a function in the SCF Console and use the SDK to invoke other functions, you need to package the `tencentcloud` library and function code together into a zip file. You can also run the following command in the root directory of the function to download and install the SDK to the function directory.

```
pip install tencentcloud-sdk-python -t .
```

- Please note that the execution method specified when the function is created in the console must correspond to the code file and execution function in the zip file.
- If the generated zip package is larger than 50 MB, it should be uploaded through COS.
- The default call rate limit for TencentCloud API is 20 calls per second. If you need to increase the limit for high concurrency, please [submit a ticket](#) for application.

Related Information

You can also use Tencent SCF SDK (Tencentserverless SDK), which integrates SCF business flow APIs to simplify the function invocation method and eliminates your need to encapsulate public TencentCloud APIs. For more information, please see [Calling SDK Across Functions](#).