

Serverless Cloud Function Getting Started Product Introduction





Copyright Notice

©2013-2018 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

Getting Started

Overview

Example 1: Creating Hello World Function

Step 1: Create Hello World Function

Step 2: Call Hello World Function

Example 2: Dowloading Files Uploaded to COS Bucket

Step 1: Create DownloadImage Function

Step 2: Call DownloadImage Function

Getting Started Overview

Last updated : 2018-08-28 11:30:33

This chapter provides some examples for users who use Serverless Cloud Function (SCF) for the first time. You can perform the following operations according to this tutorial:

- Create and test simple Hello World function.
- By configuring the action of uploading files to COS as a trigger, create and test a function: When a file is uploaded to COS Bucket, the file is downloaded to the file system of the environment.

With these two examples, users can perform some operations in the SCF console, including:

- Use templates. Each template provides sample codes and code configurations to implement certain logics. You only need to choose a required template to easily create a SCF that comes with sample codes. "Hello World" primary practice uses the Hello World template. "Download File" primary practice uses the COS Put Object template.
- Create and update the configuration information of SCF.
- Call the SCF manually and check the function execution result.

After reading "Getting Started", you can also read documents such as "How Does SCF Work?" and "Creating SCF" for more information about how to use SCF to create a production system.

Example 1: Creating Hello World Function Step 1: Create Hello World Function

Last updated : 2018-08-28 11:31:48

1) Log in to the Tencent Cloud console, and select **SCF**.

2) Click the **Create Function** button under the Guangzhou region to enter the page for creating a new function.

3) Enter hello-world as the function name and leave all other configuration options unchanged.

4) Click **Next** to enter the page for editing function codes, and then select default **Online Edit** and the Hello World template in **Template**. At this time, the default values in the template will be entered for the execution method and the code:

- Execution method is index.main_handler. SCF console stores this code in an index.py file automatically, and compresses and uploads the file to the SCF platform to create a SCF.
- The following code snippets are displayed in the function code:

```
print('Start Hello World function')
def main_handler(event, context):
print("value1 = " + event['key1'])
print("value2 = " + event['key2'])
return event['key1'] #return the value of key "key1"
```

This sample code acquires the following form of data from the parameter event :

```
{
    "key1": "test value 1",
    "key2": "test value 2"
}
```

5) Click **Next** to enter the trigger method page. You don't need to configure any trigger for this sample code, so click **Complete**.

6) At this point, the console generates a code package automatically and uploads it to the SCF platform to create a SCF. Click the hello-world function you just create in the SCF list page to enter the SCF details page.

Step 2: Call Hello World Function

Last updated : 2018-08-28 11:32:29

Use the test template provided by the console to call the Hello World function you just create.

1) Click the Hello World function you just create in the list page to enter the function details page, and click **Test** on the right.

2) In the Test Function pop-up box, select Hello World Test Code from test templates, and the following data will appear in the window. You can modify the value of the JSON data (for example, change test_1 to my_own_data), but cannot change the data structure.

```
{
    "key1": "test value 1",
    "key2": "test value 2"
}
```

3) Click Run, and the code starts running and the test result is displayed. Notes:

- The running result (successful or failed) and the function execution result returned by return statement in the code will appear in the function return value.
- The execution time of the function and memory will appear in the running information.
- Logs generated during the function execution will appear in the log, including print statement in user codes and trace stack for function execution failure, and will be written in the log module.

4) You can run the code several times, and click the **Log** tab to check the log information of each run.

Example 2: Dowloading Files Uploaded to COS Bucket Step 1: Create DownloadImage Function

Last updated : 2018-08-28 11:33:07

1) Log in to the Tencent Cloud console, and select **Cloud Object Storage**.

2) Click the **Create Bucket** button in the **Bucket List** tab to create a new COS Bucket.

3) Configure the name of COS Bucket, such as testbucket, and set the region to South China, access permission to default public read and private write, CDN acceleration to default Disabled, and click **OK** to create a new COS Bucket.

4) Navigate to **SCF** in the console, and click **Create Function** under the region **Guangzhou** to enter the page for creating a new function.

5) Enter DownloadImage as the function name and leave all other configuration options unchanged.

6) Click Next to enter the page for editing function codes, and then select default Online Edit, and the COS Put Object template in Template. At this time, the default values in the template will be entered for the execution method and the code:

Execution method is index.main_handler . SCF console stores this code in an index.py file automatically, and compresses and uploads the file to the SCF platform to create a SCF.

The following code snippets are displayed in the function code: Replace the parameter field appid, secret_id, secret_key, and region with your actual data. Notes:

• appid can be found in Account Information in the console.

Products 👻		13020005 ▼ Bi	
Account Info		Account	
		Information	
		Security Setting	
Account info	Industry Information	Permission	
Account Email 1302000590@qq.com	Industry IoT/IoV - VR devic	e e	
Account Alias 1302000590@cc.com		Settings	
		Cloud API Key	
Account ID (i) 100000624047		Open Port 25	
APPID (i) 1253702919	Contact Info Modify		
	Full Name: 123123	O with the strength of the strength	
	Country: Albania	Switch developers	
		Log out	

• secret_id and secret_key can be obtained from **Cloud API Key** in the console.

Manage API Key				Cloud API Document	
To call the Tencent C	Cloud API, you need to sign, and the Cloud API key and secret is used	to generate the signature, see the generated s	gnature algorith	m.	
API key is an important certificate to request for creating Tencent Cloud API. With the API, you can operate all your Tencent cloud resources. For your property and service security, please manage the key safely and change it regularly. After you've changed to a new key, please delete the old one timely.					
APPID	Key	Creation Time	Status	Operation	
1253702919	SecretId: AKIDoSs42IPVWIp8C2K2B5kBNK2VjiTzgLAk SecretKey: ****** Show	2017-06-01 16:50:28	Enabled	Disable	

 region is the region in which the function and COS Bucket reside. sh, gz, and bj are supported. Note: The function must be in the same region with COS Bucket. The storage bucket created in the first step resides in South China (Guangzhou), so the region value in the code must be gz.

import json
import urllib
import commands
import logging
from qcloud_cos import CosClient
from qcloud_cos import DownloadFileRequest
print('Loading function')
appid = 1251111111 #please change to your appid. Find it in Account Info
secret_id = u'AKIDYDh085xQp48161uOn2CKKVbeebvDu9lb' #please change to your API secret id. Fin



d it in API secret key pair secret key = 'ILkxx40kIfuyqW0IOI0WqyueCYjlgzqE' #please change to your API secret key. Find it in A PI secret key pair region = u'qz'def main handler(event,context): logger = logging.getLogger() bucket = event['Records'][0]['cos']['cosBucket']['name'] key = urllib.unquote_plus(event['Records'][0]['cos']['cosObject']['key'].encode('utf8')) try: cos client = CosClient(appid, secret id, secret key, region) request = DownloadFileRequest(bucket, key, '/tmp'+key) download file ret = cos client.download file(request) if download file ret['code'] == 0: logger.info("Download file [%s] Success" % key) logger.info("find local file:" + commands.getoutput('ls /tmp')) return "download success" else: logger.error("Download file [%s] Failed, err: %s" % (key, download file ret['message'])) return -1 except Exception as e: print(e) print('Error getting object {} from bucket {}. Make sure the object exists and your bucket is in the sam e region as this function.'.format(key, bucket)) raise e

7) Click **Next** to enter the trigger method page. Click **Create Trigger Method** to add a new trigger for the function. Set the trigger method to **COS Trigger**, select the testbucket storage bucket you just create, and the File Upload for the event type, and then click **Save**.

8) Click **Complete** at the bottom. At this point, the console generates a code package automatically and uploads it to the SCF platform to create a SCF. Click the DownloadImage function you just create in the SCF list page to enter the SCF details page.

Step 2: Call DownloadImage Function

Last updated : 2018-08-28 11:33:51

Test the DownloadImage function you just create by following the steps below.

Using COS to Upload Files for Testing

- 1. Log in to the Tencent Cloud console, navigate to **Cloud Object Storage**, select the **TestBucket** Bucket created in Step 1 in the Bucket list, and click the **Upload files** button to upload a sample image: testimage.jpeg. You can download this image to a local machine for testing.
- 2. Navigate to **SCF**, and select DownloadImage you just created.
- 3. Click **Log**, and check whether the function execution log of the image you just update is generated. The log of the image downloaded to the local machine should be included, such as:

Loading function

Uri is http://TestBucket-1251111111.cosgz.myqcloud.com/testimage.jpeg?sign=QlEyq+WH8g5RpD+L 6sPk05XhVQthPTEyNTE3NjlyMjcmaz1BS0lEWURoMDg1eFFwNDgxNjF1T24yQ0tLVmJIZWJ2RHU2ak8m ZT0xNDk2ODM5NDQ3JnQ9MTQ5NjgzOTE0NyZyPTk1NDI3NjgyNCZmPS8xNDcyNjQwNzgwXzg0X3cx NjE0X2g0NDAucG5nJml9ZG9uZ3l1YW50ZXEE Starting new HTTP connection (1): TestBucket-1251111111.cosgz.myqcloud.com http://TestBucket-1251111111.cosgz.myqcloud.com:80 "GET /testimage.jpeg?sign=QlEyq+WH8g5Rp D+L6sPk05XhVQthPTEyNTE3NjlyMjcmaz1BS0lEWURoMDg1eFFwNDgxNjF1T24yQ0tLVmJIZWJ2RHU2a k8mZT0xNDk2ODM5NDQ3JnQ9MTQ5NjgzOTE0NyZyPTk1NDI3NjgyNCZmPS8xNDcyNjQwNzgwXzg0 X3cxNjE0X2g0NDAucG5nJml9ZG9uZ3l1YW50ZXEE HTTP/1.1" 200 62296 Download file [/testimage.jpeg] Success testimage.jpeg

Manual Simulation Test

You can view the function's running status by entering the test data, such as COS trigger.

1) In the Test Function pop-up box, select Upload File to COS/Delete File from COS Test Code from test templates, and the following data will appear in the window. The following modifications are required:

• Replace name in cosBucket with TestBucket you just created.

• Replace key in cosObject with \testimage.jpeg you just uploaded.

As shown below:

```
{
"Records":
{
"event": {
"eventVersion":"1.0",
"eventSource":"qcs::cos",
"eventName": "cos:ObjectCreated:*,
"eventTime":"1970-01-01T00:00:00.000Z",
"eventQueue":"qcs:0:cos:gz:1251111111:cos",
"requestParameters":{
"requestSourceIP": "111.111.111.111",
"requestHeaders":{
"Authorization": "Example"
}
}
},
"cos":{
"cosSchemaVersion":"1.0",
"cosNotificationId":"Configured or returned ID",
"cosBucket":{
"name":"TestBucket", # Notice Here
"appid":"1251111111",
"region":"gz",
},
"cosObject":{
"key":"/testimage.jpg", # Notice Here
"size":"1024",
"meta":{
"Content-Type": "text/plain",
"x-cos-meta-test": "Custom meta",
"x-image-test": "Custom meta"
},
"url": "Origin server URL for accessing files"
}
}
}
1
}
```

2) Click Run, and the code starts running and the test result is displayed. Notes:

- The running result and the function execution result returned by return statement in the code will appear in the function return value.
- The execution time of the function and memory will appear in the running information.
- Logs generated during the function execution will appear in the log, including print statement in user codes and trace stack for function execution failure, and will be written in the log module.

3) You can run the code several times, and click the **Log** tab to check the log information of each run.