

# **Cloud Load Balance**

## **SDK Documents**

### **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



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 Documents

Python SDK

# SDK Documents

## Python SDK

Last updated : 2018-08-21 17:36:12

### CLB SDK Instruction

## Python SDK Introduction (Linux)

### Environment Dependencies

Python 2.x. Python 3 is not supported for now.

Dependent library: requests

How to get Python version (Linux Shell):

```
$python -V  
  
Python 2.7.11
```

### Downloading and Configuring CLB SDK

#### Cloud API Key Instructions

When using the SDK, the user's Cloud API key is required to verify the validity of the user's identity.

How to obtain Cloud API key:

Log in to [Tencent Cloud Console](#), and select **Cloud Products** -> **Cloud API Key**.

Users can create a new Cloud API key or use an existing key. Click the key ID to enter the details page in order to get the secretId of the key and its corresponding secretKey.



## Downloading CLB Python SDK

Download the latest [CLB Python SDK](#).

## Using CLB Python SDK

### 1. Configure Cloud API Key

Specify secretId and secretKey in file `CLB_SDK_0.0.1/src/QcloudApi/qcloudapi.py` of SDK. Here is part of the code in the file:

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

config = {
    'Region': 'gz',
    'secretId': '',
    'secretKey': '',
    'method': 'post'
}

class QcloudApi:
    def __init__(self, module='lb', config=config, region='gz'):
```

### 2. Use Case for a Specific API:

The following code is in `CreateForwardLBFourthLayerListeners.py` (API for creating an application-based load balancer layer-4 listener) under the directory `sample/application` of Python SDK. `region` in the code refers to the region of instance to be operated, which is specified based on the actual situation.

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

import sys
import os
sys.path.insert(0, os.path.dirname(os.path.abspath(__file__)) + '/../..')
```

```

from src.QcloudApi.qcloudapi import QcloudApi

action = 'CreateForwardLBFourthLayerListeners' # Create application-based load balancer layer-4 listeners

"""
loadBalancerId Required Load balancer ID
listeners.n.loadBalancerPort Required Load balancer listener
listeners.n.protocol Required Listening protocol (2: TCP, 3: UDP) of load balancer listener
listeners.n.listenerName Optional Name of load balancer listener
listeners.n.sessionExpire Optional Session persistence duration (in sec) of load balancer listener . Session persistence is not supported for private network load balancer. Default is 0 (Disabled).
listeners.n.healthSwitch Optional Whether to enable health check for load balancer instance listeners (1: On, 0: Off). Default is 1 (On).
listeners.n.timeOut Optional Health check response timeout (in sec) for load balancer listeners. Value range: 2-60. Default is 2. The response timeout must be smaller than the interval between health checks.
listeners.n.intervalTime Optional Health check time interval (in sec) for load balancer listeners. Default is 5. Value range: 5-300.
listeners.n.healthNum Optional Healthy threshold of load balancer listener (in count). Default is 3, which means the forwarding is considered normal if it is detected to be healthy for three times consecutively. Value range: 2-10.
listeners.n.unhealthNum Optional Unhealthy threshold of load balancer listener (in count). Default is 3, which means the forwarding is considered normal if it is detected to be healthy for three times consecutively. Value range: 2-10.

"""
region = 'gz'
params = {
    'loadBalancerId': "lb-j2nvt9hq",
    'listeners.0.loadBalancerPort': 80,
    'listeners.0.protocol': 2,
    'listeners.0.listenerName': "test",
}

try:
    service = QcloudApi(region=region)
    print service.generateUrl(action, params)
    print service.call(action, params)
except Exception, e:
    print 'exception:', e

```