

SDK

SDK

Product Documentation



Tencent Cloud

Copyright Notice

©2013-2024 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

PHP

Python

Java

Go

.NET

Node.js

C++

SDK

PHP

Last updated : 2019-11-01 10:32:51

Overview

Welcome to Tencent Cloud Software Development Kit (SDK) 3.0, a companion tool for the Cloud API 3.0 platform. Currently, it supports products such as CVM, VPC and CBS. All cloud services and products will be integrated here for access in the future. The new version of SDK is unified and features the same SDK usage, API call methods, error codes and return packet formats for different languages.

To make it easier for PHP developers to debug and access the APIs of Tencent Cloud products, this document describes the Tencent Cloud SDK for PHP and provides a simple example of using the SDK for the first time, helping you quickly get the SDK and start calling.

Dependent Environment

1. PHP version 5.6.33 or higher.
2. Activate the corresponding product in the Tencent Cloud [Console](#).
3. Get the SecretID, SecretKey and call address (endpoint). The general format of endpoint is *.tencentcloudapi.com. For example, the call address of CVM is cvm.tencentcloudapi.com. For details, see the documentation of the specific product.

Installation

Obtain the security credentials before installing the SDK for PHP. Before using the Cloud API for the first time, you need to first apply for security credentials in the Tencent Cloud Console, including SecretID and SecretKey. SecretID is used to identify the API caller, while SecretKey is used to encrypt the signature string and verify it on the server. You must keep the SecretKey private and avoid disclosure.

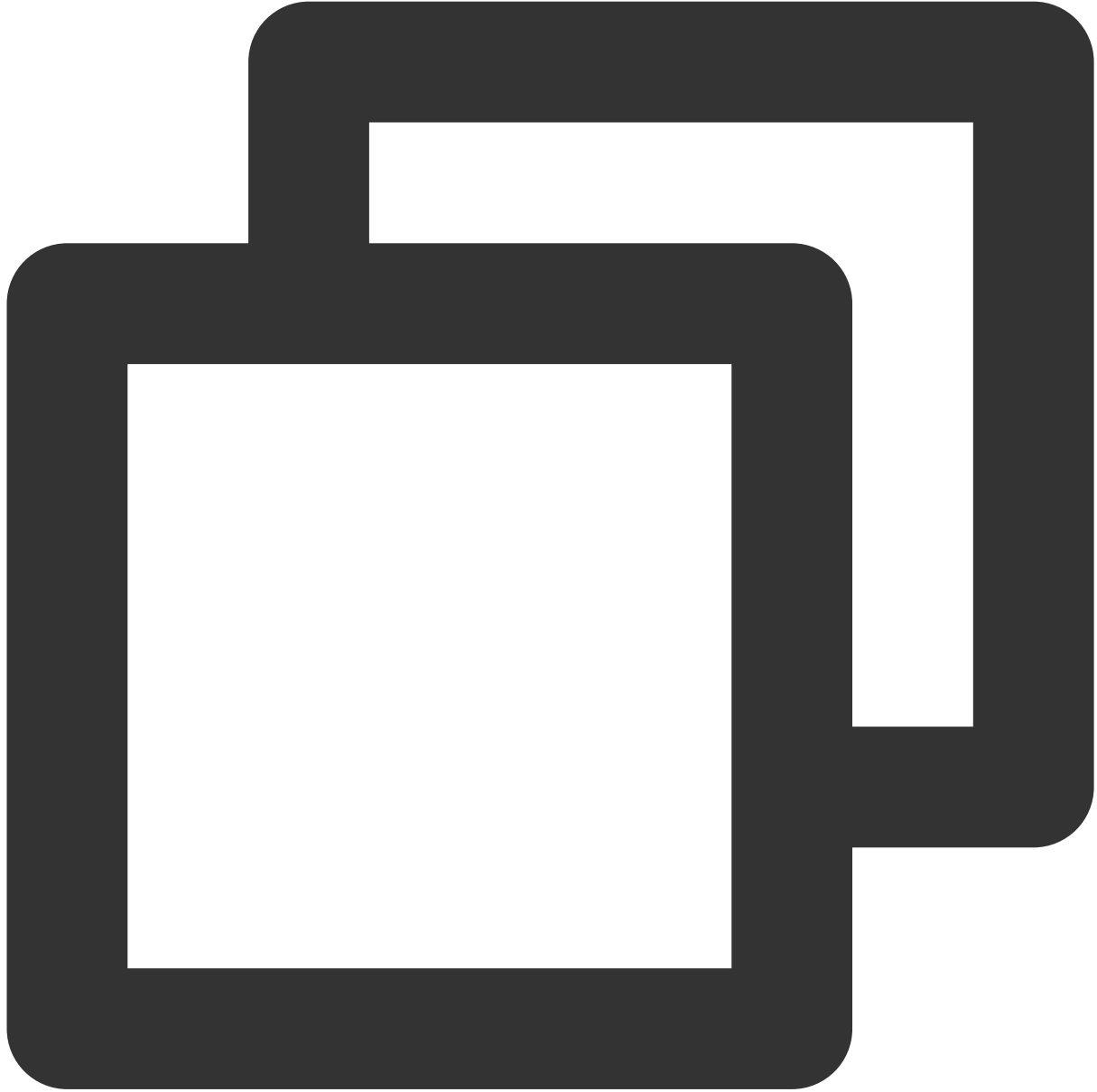
Installing via Composer

Installing via Composer is the recommended way to use the SDK for PHP. Composer is a dependency management tool for PHP that supports the dependencies your project requires and installs them into your project. For more information on Composer, see [Composer's official website](#).

1. Install Composer:

For Windows, go to [Composer's official website](#) to download the installation package.

For Unix, install by executing the following command in command line.



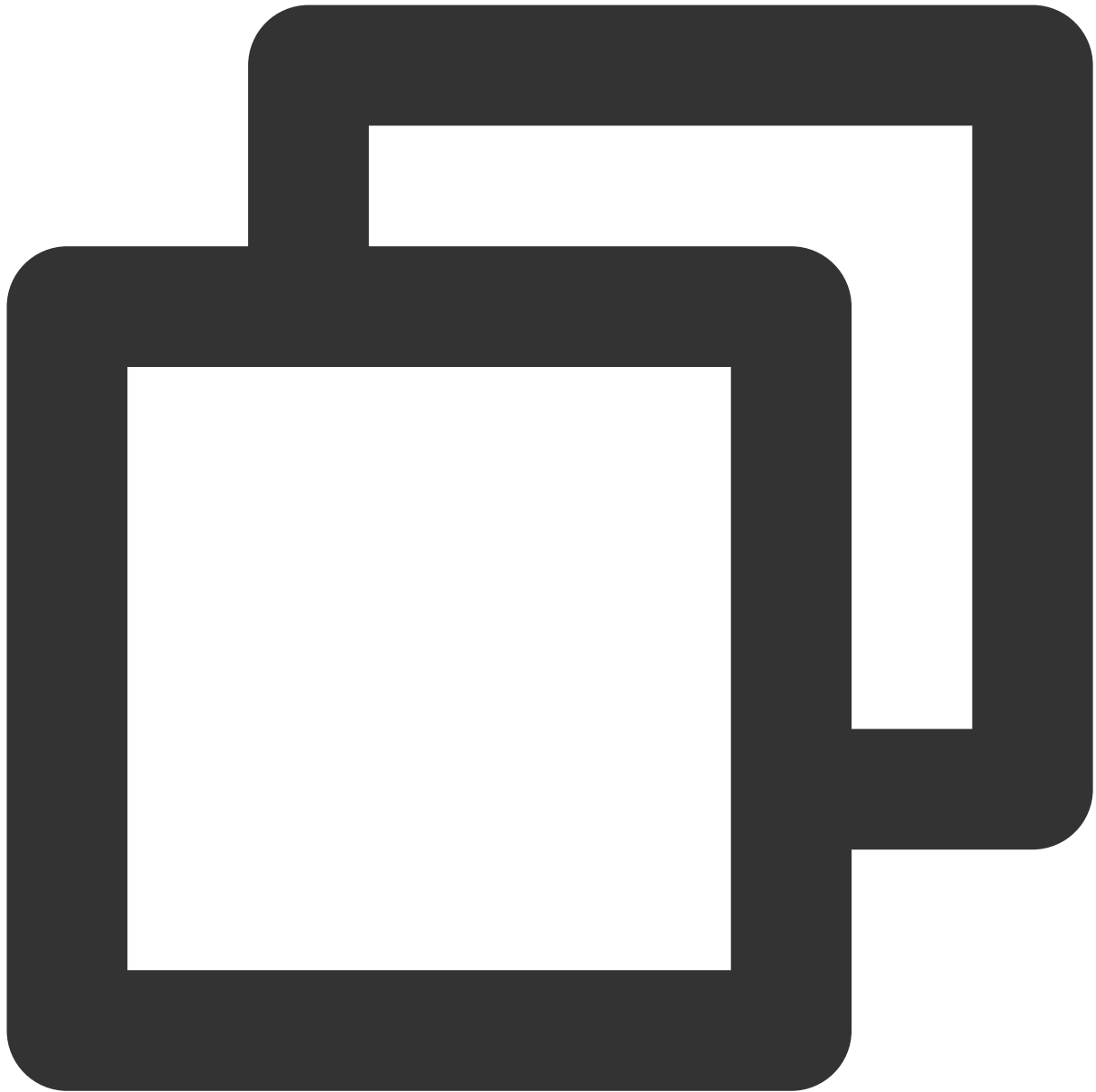
```
curl -sS https://getcomposer.org/installer | php
```

2. Add dependencies to the "require" structure of composer.json. **Please note that the version number here is just an example, and you can view the latest version number on the Composer repository.**



```
"tencentcloud/tencentcloud-sdk-php-intl": "3.0.0"
```

3. Execute the "composer install" command to download and install the SDK for PHP.
4. Add the following reference code. For reference methods, see the example.



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

Example

Take the API for querying available zones as an example:



```
<?php
require_once '../.../TCloudAutoLoader.php';
// Import the client of the corresponding product module.
use TencentCloud\Cvm\V20170312\CvmClient;
// Import the Request class corresponding to the request API.
use TencentCloud\Cvm\V20170312\Models\DescribeZonesRequest;
use TencentCloud\Common\Exception\TencentCloudSDKException;
use TencentCloud\Common\Credential;
try {
    // Instantiate a certificate object. The Tencent Cloud account secretId and secretKey
    $cred = new Credential("secretId", "secretKey");
```



```
// # Instantiate the client object to request the product (with CVM as an example)
$client = new CvmClient($cred, "ap-guangzhou");

// Instantiate a request object.
$req = new DescribeZonesRequest();

// Call the API you want to access through the client object; you need to pass
$res = $client->DescribeZones($req);

print_r($res->toJsonString());
}
catch(TencentCloudSDKException $e) {
    echo $e;
}
```

More Examples

You can find more detailed examples in the examples directory of the GitHub repository.

Python

Last updated : 2020-07-03 16:20:16

Overview

Welcome to Tencent Cloud Software Development Kit (SDK) 3.0, a companion tool for the Cloud API 3.0 platform. Currently, it supports products such as CVM, VPC and CBS. All cloud services and products will be integrated here for access in the future. The new version of SDK is unified and features the same SDK usage, API call methods, error codes and return packet formats for different languages.

To make it easier for Python developers to debug and access the APIs of Tencent Cloud products, this document describes the Tencent Cloud SDK for Python and provides a simple example of using the SDK for the first time, helping you quickly get the SDK and start calling.

Dependent Environment

1. Dependent environment: Python version 2.7 to 3.6.
2. Activate the corresponding product in the [Tencent Cloud Console](#).
3. Get the SecretID, SecretKey and call address (endpoint). The general format of endpoint is *.tencentcloudapi.com. For example, the call address of CVM is cvm.tencentcloudapi.com. For details, see the documentation of the specific product.

Installation

Obtain the security credentials before installing the SDK for Python. Before using the Cloud API for the first time, you need to first apply for security credentials in the Tencent Cloud Console, including SecretID and SecretKey. SecretID is used to identify the API caller, while SecretKey is used to encrypt the signature string and verify it on the server. You must keep the SecretKey private and avoid disclosure.

Installing via Pip (Recommended)

You can install the Tencent Cloud API SDK for Python into your project via Pip. If you haven't installed Pip in your project environment yet, install it first by following the instruction at [Pip's official website](#).

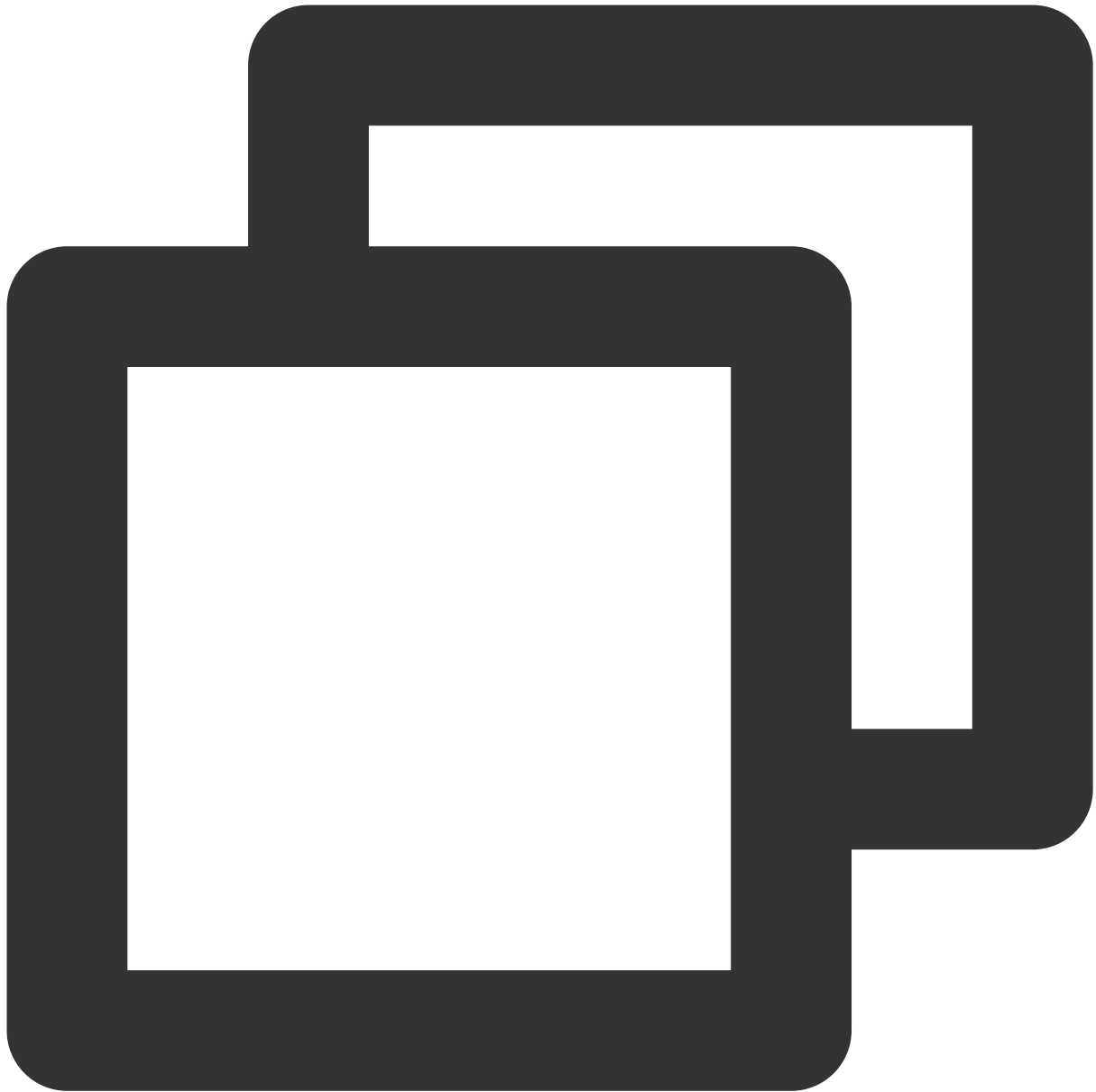
To install via Pip, execute the following command in command line:



```
pip install tencentcloud-sdk-python-intl-en
```

Example

Take the API for querying available zones as an example:



```
# -*- coding: utf-8 -*-
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import TencentCloudS
# Import the client models of the corresponding product module.
from tencentcloud.cvm.v20170312 import cvm_client, models
try:
    # Instantiate an authentication object. The Tencent Cloud account secretId and
    cred = credential.Credential("secretId", "secretKey")

    # Instantiate the client object to request the product (with CVM as an example)
    client = cvm_client.CvmClient(cred, "ap-shanghai")
```

```
# Instantiate a request object.
req = models.DescribeZonesRequest()

# Call the API you want to access through the client object; you need to pass i
resp = client.DescribeZones(req)
# A string return packet in json format is output.
print(resp.to_json_string())

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

More Examples

You can find more detailed examples in the examples directory of the GitHub repository.

Java

Last updated : 2019-11-01 10:39:05

Overview

Welcome to Tencent Cloud Software Development Kit (SDK) 3.0, a companion tool for the TencentCloud API 3.0 platform. Currently, it supports products such as CVM, VPC and CBS. All cloud services and products will be integrated here for access in the future. The new version of SDK is unified and features the same SDK usage, API call methods, error codes and return packet formats for different languages.

To make it easier for Java developers to debug and access the APIs of Tencent Cloud products, this document describes the Tencent Cloud SDK for Java and provides a simple example of using the SDK for the first time, helping you quickly get the SDK and start calling.

Dependent Environment

1. Dependent environment: JDK version 7 or higher.
2. Activate the corresponding product in the [Tencent Cloud Console](#).
3. Get the SecretID, SecretKey and call address (endpoint). The general format of endpoint is *.tencentcloudapi.com. For example, the call address of CVM is cvm.tencentcloudapi.com. For details, see the documentation of the specific product.

Installation

Obtain the security credentials before installing the SDK for Java. Before using the TencentCloud API for the first time, you need to first apply for security credentials in the Tencent Cloud Console, including SecretID and SecretKey. SecretID is used to identify the API caller, while SecretKey is used to encrypt the signature string and verify it on the server. You must keep the SecretKey private and avoid disclosure.

Installing via Maven (Recommended)

Installing via Maven is the recommended way to use the SDK for Java. Maven is a dependency management tool for Java that supports the dependencies your project requires and installs them into your project. For more information on Maven, see Maven's official website.

1. Go to [Maven's official website](#) to download the corresponding Maven installation package for your system and install.
2. Add Maven dependencies to your project by adding the following dependencies in Maven's pom.xml:

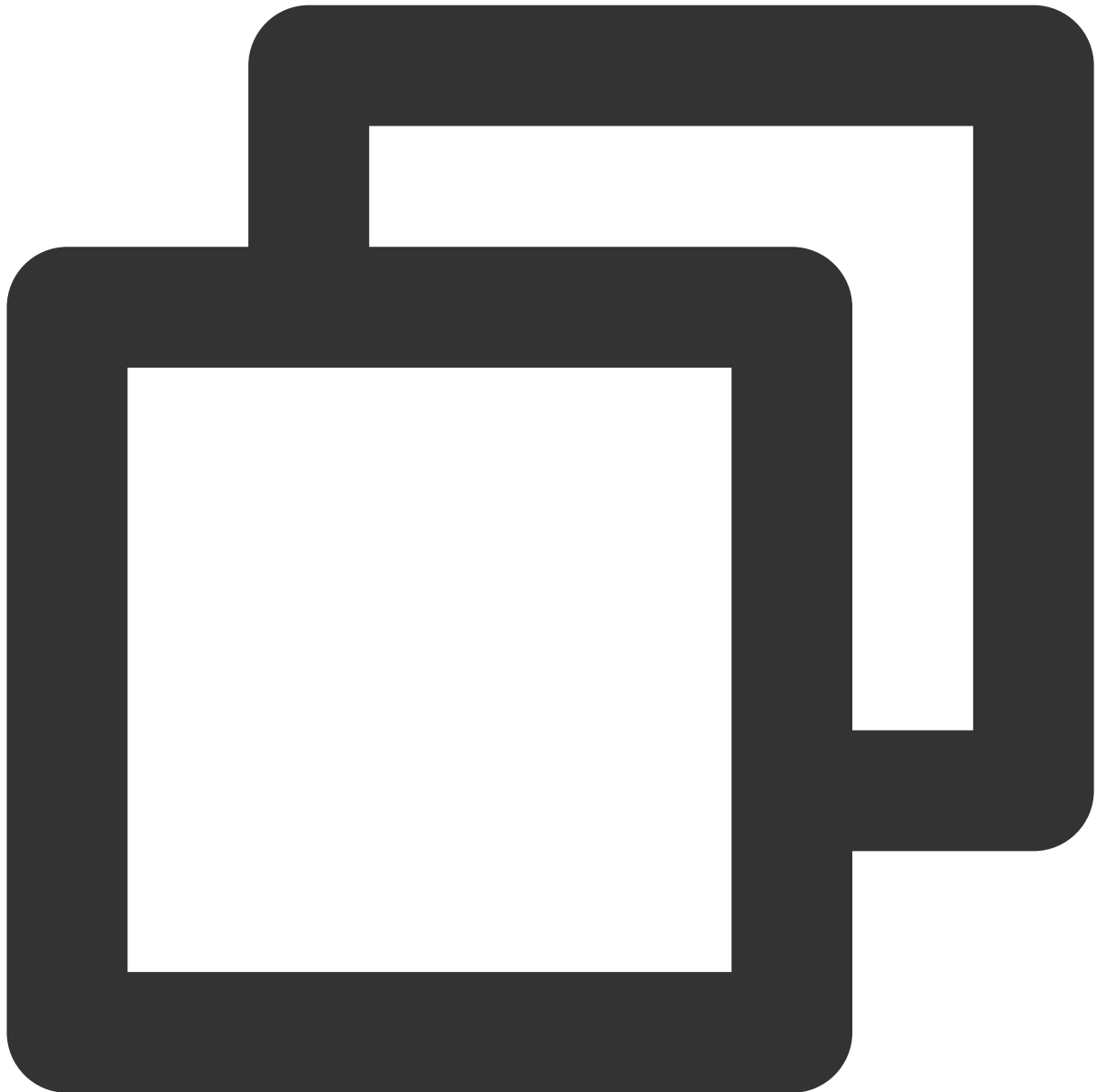


```
<dependency>
  <groupId>com.tencentcloudapi</groupId>
  <!-- Note: Please see the latest version number in the GitHub or Maven repository -->
  <artifactId>tencentcloud-sdk-java-intl</artifactId>
  <version>3.0.0</version>
</dependency>
```

3. For reference methods, see the example.

Example

Take the API for querying available zones as an example:



```
import com.tencentcloudapi.common.Credential;
import com.tencentcloudapi.common.exception.TencentCloudSDKException;
// Import the client of the corresponding product module.
import com.tencentcloudapi.cvm.v20170312.CvmClient;
// Import the request response class corresponding to the request API.
import com.tencentcloudapi.cvm.v20170312.models.DescribeZonesRequest;
import com.tencentcloudapi.cvm.v20170312.models.DescribeZonesResponse;
```



```
public class DescribeZones
{
    public static void main(String [] args) {
        try{
            // Instantiate an authentication object. The Tencent Cloud account secret
            Credential cred = new Credential("secretId", "secretKey");

            // Instantiate the client object to request the product (with CVM as an
            CvmClient client = new CvmClient(cred, "ap-guangzhou");

            // Instantiate a request object.
            DescribeZonesRequest req = new DescribeZonesRequest();

            // Call the API you want to access through the client object; you need
            DescribeZonesResponse resp = client.DescribeZones(req);

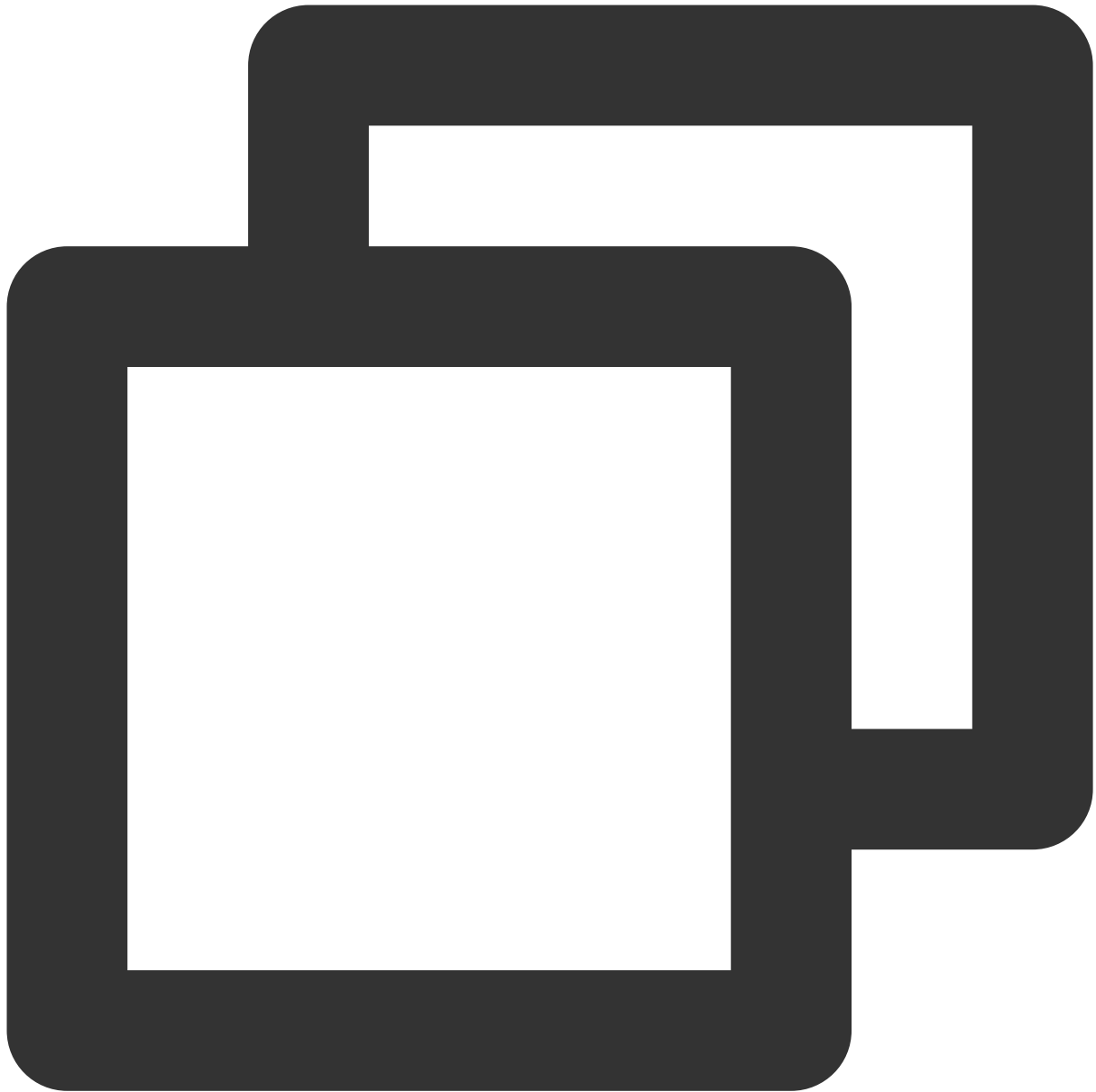
            // A string return packet in json format is output.
            System.out.println(DescribeZonesRequest.toJsonString(resp));
        } catch (TencentCloudSDKException e) {
            System.out.println(e.toString());
        }
    }
}
```

More Examples

You can find more detailed examples in the examples directory of the GitHub repository.

Older SDK

We recommend that you use the new version of SDK. If you need an older version, add the following dependencies to your Maven's pom.xml:



```
<dependency>  
<groupId>com.qcloud</groupId>  
<artifactId>qcloud-java-sdk</artifactId>  
<version>2.0.6</version>  
</dependency>
```

Go

Last updated : 2022-08-09 16:20:04

Overview

Welcome to Tencent Cloud Software Development Kit (SDK) 3.0, a companion tool for the Cloud API 3.0 platform. All cloud services and products will be integrated here for access in the future. The new version of SDK is unified and features the same SDK usage, API call methods, error codes and return packet formats for different languages.

To make it easier for Go developers to debug and access the APIs of Tencent Cloud products, this document describes the Tencent Cloud SDK for Go and provides a simple example of using the SDK for the first time, helping you quickly get the SDK and start calling.

Dependent Environment

1. Go version 1.9 or higher. Plus, the necessary environment variables such as GOPATH have to be set properly.
2. Before using, make sure to activate the corresponding product in the Tencent Cloud [Console](#).
3. Get the SecretID and SecretKey on the [Access Management](#) page in the Tencent Cloud Console.

Installation

Obtain the security credentials before installing the SDK for Go. Before using the Cloud API for the first time, you need to first apply for security credentials in the Tencent Cloud Console, including SecretID and SecretKey. SecretID is used to identify the API caller, while SecretKey is used to encrypt the signature string and verify it on the server. You must keep the SecretKey private and avoid disclosure.

Installing via go get (Recommended)

It is recommended to install the SDK using the tool that comes with the language:



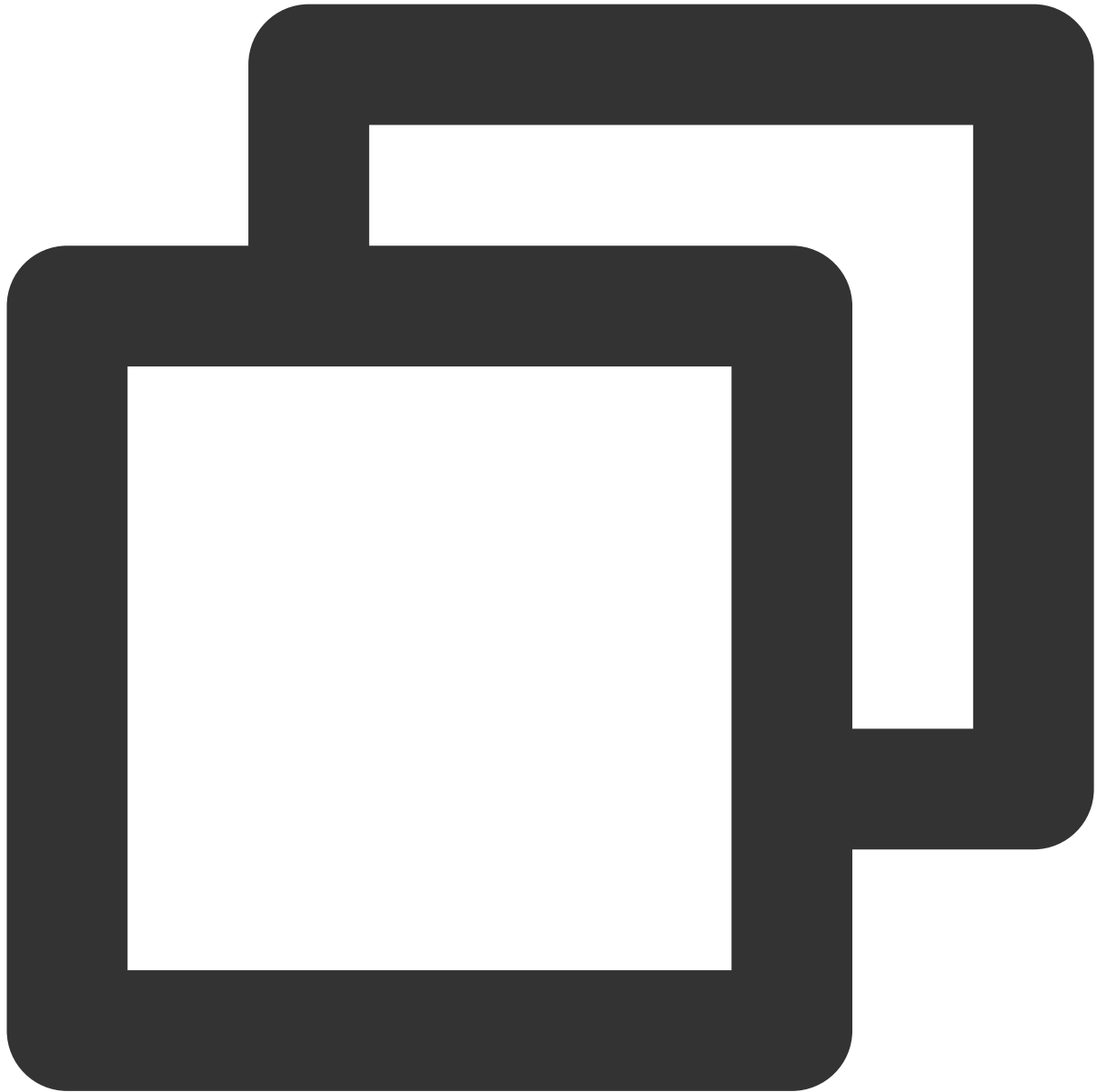
```
go get -u github.com/tencentcloud/tencentcloud-sdk-go-intl-en
```

Example

Each API has a corresponding request structure and a response structure. For example, the DescribeZones API for querying availability zones has a corresponding request structure DescribeZonesRequest and a response structure

DescribeZonesResponse.

The following uses the API for querying availability zones as an example to introduce the basic usage of the SDK.



```
package main

import (
    "fmt"

    "github.com/tencentcloud/tencentcloud-sdk-go-intl-en/tencentcloud/common"
    "github.com/tencentcloud/tencentcloud-sdk-go-intl-en/tencentcloud/common/er
    "github.com/tencentcloud/tencentcloud-sdk-go-intl-en/tencentcloud/common/pr
```

```
cvm "github.com/tencentcloud/tencentcloud-sdk-go-intl-en/tencentcloud/cvm/v
)

func main() {
    // Instantiate an authentication object. The Tencent Cloud account secretId
    credential := common.NewCredential(
        "your-secret-id",
        "your-secret-key",
    )

    // Instantiate a client configuration object; you can specify the timeout a
    cpf := profile.NewClientProfile()
    cpf.HttpProfile.ReqMethod = "GET"
    cpf.HttpProfile.ReqTimeout = 5
    cpf.SignMethod = "HmacSHA1"

    // Instantiate the client object to request the product (with CVM as an exa
    client, _ := cvm.NewClient(credential, "ap-beijing", cpf)
    // Instantiate a request object; you can further set the request parameters
    request := cvm.NewDescribeZonesRequest()
    // Call the API you want to access through the client object; you need to p
    response, err := client.DescribeZones(request)
    // Handle the exception
    if _, ok := err.(*errors.TencentCloudSDKError); ok {
        fmt.Printf("An API error has returned: %s", err)
        return
    }
    // unexpected errors
    if err != nil {
        panic(err)
    }
    // Print the returned json string
    fmt.Printf("%s", response.ToJsonString())
}
```

More Examples

For more examples, see the [examples](#) directory. For an example of request initialization for complex APIs, see [examples/cvm/v20170312/run_instances.go](#). For an example of initializing a request using a json string, see [examples/cvm/v20170312/describe_instances.go](#).

.NET

Last updated : 2019-11-01 10:47:43

Overview

Welcome to Tencent Cloud Software Development Kit (SDK) 3.0, a companion tool for the TencentCloud API 3.0 platform. All cloud services and products will be integrated here for access in the future. The new version of SDK is unified and features the same SDK usage, API call methods, error codes and return packet formats for different languages.

To make it easier for .NET developers to debug and access the APIs of Tencent Cloud products, this document describes the Tencent Cloud SDK for .NET and provides a simple example of using the SDK for the first time, helping you quickly get the SDK and start calling.

Dependent Environment

1. Dependent environment: .NET Framework 4.5+ and .NET Core 2.1.
2. Activate the corresponding product in the Tencent Cloud [Console](#).
3. Get the SecretID, SecretKey and call address (endpoint). The general format of endpoint is *.tencentcloudapi.com. For example, the call address of CVM is cvm.tencentcloudapi.com. For details, see the documentation of the specific product.
4. Download the relevant materials and configure the relevant files.

Installation

Obtain the security credentials before installing the SDK for .NET. Before using the TencentCloud API for the first time, you need to first apply for security credentials in the Tencent Cloud Console, including SecretID and SecretKey. SecretID is used to identify the API caller, while SecretKey is used to encrypt the signature string and verify it on the server. You must keep the SecretKey private and avoid disclosure.

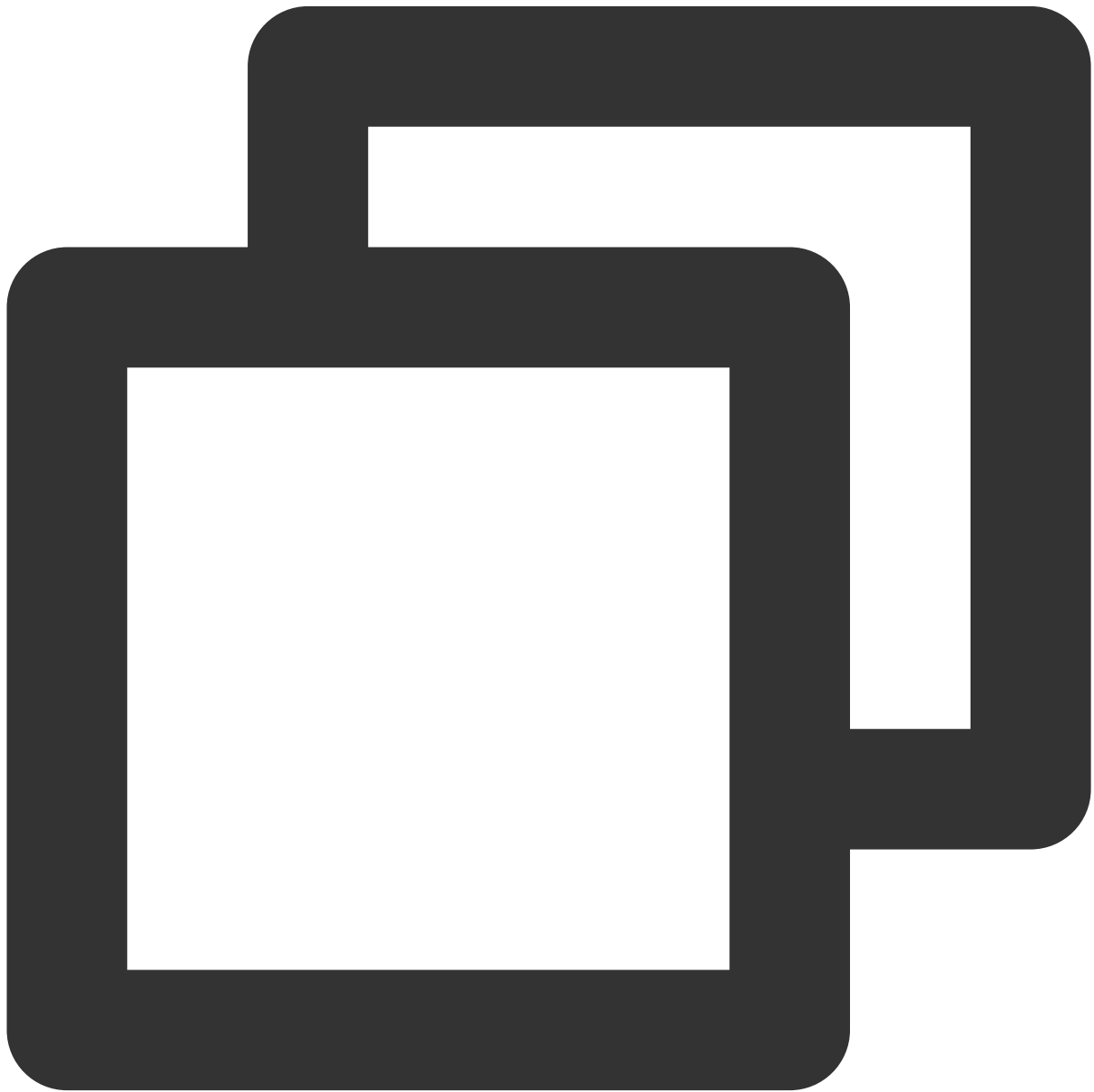
Installation via NuGet (Recommended)

1. Install using the command line `dotnet add package TencentCloudSDKIntl --version 3.0.0`. Please go to the [NuGet](#) page to get additional information.
2. Add the package via Visual Studio.

Example

Each API has a corresponding request structure and a response structure. For example, the DescribeInstances API for querying CVM instance list has a corresponding request structure DescribeInstancesRequest and a response structure DescribeInstancesResponse.

The following uses the API for querying CVM instance list as an example to introduce the basic usage of the SDK. For the purpose of demonstration, some nonessential items have been added to show the common functions of the SDK, which makes the example look bloated. When using the SDK to write code, it is recommended to keep it simple.




```
using System;
using System.Threading.Tasks;
using TencentCloud.Common;
using TencentCloud.Common.Profile;
using TencentCloud.Cvm.V20170312;
using TencentCloud.Cvm.V20170312.Models;

namespace TencentCloudExamples
{
    class DescribeInstances
    {
        static void Main(string[] args)
        {
            try
            {
                // Essential steps:
                // Instantiate an authentication object. The Tencent Cloud account
                // The example here uses the way to read from environment variable,
                // You can also write the key pair directly into the code, but be c
                // Otherwise, the key pair may be leaked, causing damage to your pr
                Credential cred = new Credential {
                    SecretId = Environment.GetEnvironmentVariable("TENCENTCLOUD_SEC
                    SecretKey = Environment.GetEnvironmentVariable("TENCENTCLOUD_SE
                };

                // Instantiate a client option (optional; skip if no special requir
                ClientProfile clientProfile = new ClientProfile();
                // Specify the signature algorithm (HmacSHA256 by default).
                clientProfile.SignMethod = ClientProfile.SIGN_SHA1;
                // Nonessential steps
                // Instantiate a client configuration object; you can specify the t
                HttpProfile httpProfile = new HttpProfile();
                // The SDK uses the POST method by default.
                // If you have to use the GET method, you can set it here, but the
                httpProfile.ReqMethod = "POST";
                // The SDK has a default timeout; do not adjust it unless absolutel
                // If needed, check in the code to get the latest default value.
                httpProfile.Timeout = 10; // Request connection timeout in seconds
                // The SDK automatically specifies the domain name. Generally, you
                // you have to manually specify the domain name, such as cvm.ap-sha
                httpProfile.Endpoint = ("cvm.tencentcloudapi.com");
                // Proxy server; set when there is a proxy server in your environme
                httpProfile.WebProxy = Environment.GetEnvironmentVariable("HTTPS_PR

                clientProfile.HttpProfile = httpProfile;
            }
        }
    }
}
```

```
// Instantiate the client object to request the product (with CVM a
// The second parameter is the region information. You can directly
CvmClient client = new CvmClient(cred, "ap-guangzhou", clientProfil

// Instantiate a request object; you can further set the request pa
// You can directly query the SDK source code to determine which at
// The attribute may be of a basic type or refer to another data st
// It is recommended to use the IDE for development where you can e
DescribeInstancesRequest req = new DescribeInstancesRequest();

// Settings of a basic parameter.
// This API allows you to set the number of instances returned. Her

req.Limit = 1;
// Settings of an array.
// This API allows for filtering based on the specified instance ID
// req.InstanceIds = new string[] { "ins-r8hr2upy" };

// Settings of a complex object.
// In this API, "Filters" is an array whose elements are complex ob
// Populated request parameters. Here, the member variables of the
// You can view the definition of the request parameter in the API
Filter respFilter = new Filter(); // Create a Filter object and que
respFilter.Name = "zone";
respFilter.Values = new string[] { "ap-guangzhou-1", "ap-guangzhou-
req.Filters = new Filter[] { respFilter }; // Filters is a list of

///// Here, you can assign values to the request parameters using a
//string strParams = "{\\"Filters\\":[{\\"Name\\":\\"zone\\",\\"Val
//req = DescribeInstancesRequest.FromJsonString<DescribeInstancesRe

// Initialize the request by calling the DescribeInstances method o
// The returned resp is an instance of the DescribeInstancesRespons
DescribeInstancesResponse resp = client.DescribeInstances(req).
    ConfigureAwait(false).GetAwaiter().GetResult();

// A string return packet in json format is output.
Console.WriteLine(AbstractModel.ToJsonString(resp));

// You can also take a single value.
// You can view the definition of the return field in the API docum
Console.WriteLine(resp.TotalCount);

}
catch (Exception e)
{
    Console.WriteLine(e.ToString());
}
```

```
        }
        Console.Read();
    }
}
```

More Examples

For more examples, see the TencentCloudExamples directory at [GitHub](#).

Node.js

Last updated : 2023-03-22 11:14:46

Overview

Welcome to Tencent Cloud Software Development Kit (SDK) 3.0, a companion tool for the TencentCloud API 3.0 platform. Currently, it supports products such as CVM, VPC and CBS. All cloud services and products will be integrated here for access in the future. The new version of SDK is unified and features the same SDK usage, API call methods, error codes and return packet formats for different languages.

To make it easier for NODEJS developers to debug and access the APIs of Tencent Cloud products, this document describes the Tencent Cloud SDK for NODEJS and provides a simple example of using the SDK for the first time, helping you quickly get the SDK and start calling.

API Explore

API Explore provides various capabilities such as online call, signature verification, SDK code generation and quick API retrieval that significantly reduce the difficulty of using cloud APIs.

Dependent Environment

1. NODEJS version 7.10.1 or higher.
2. Activate the corresponding product in the Tencent Cloud [Console](#).
3. Get the SecretID, SecretKey and call address (endpoint). The general format of endpoint is *.tencentcloudapi.com. For example, the call address of CVM is cvm.tencentcloudapi.com. For details, see the documentation of the specific product.

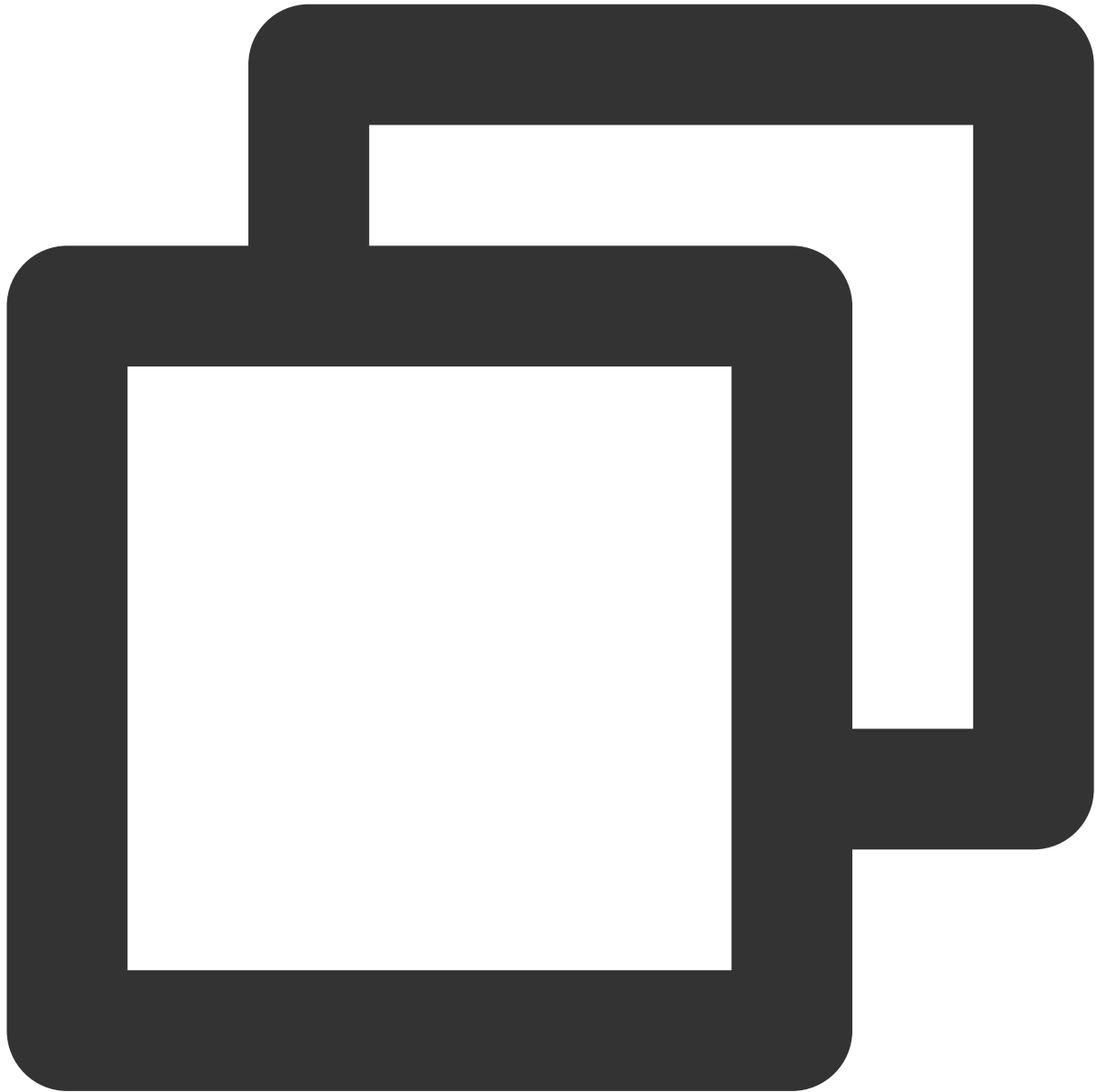
Installation

Obtain the security credentials before installing the SDK for NODEJS. Before using the TencentCloud API for the first time, you need to first apply for security credentials in the Tencent Cloud Console, including SecretID and SecretKey. SecretID is used to identify the API caller, while SecretKey is used to encrypt the signature string and verify it on the server. You must keep the SecretKey private and avoid disclosure.

Installing via npm

Installing via npm, a NODEJS package management tool, is the recommended way to use the SDK for NODEJS. For more information on npm, see [npm's official website](#).

1. Execute the following installation command:



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

2. Refer to the corresponding module code in your code. For details, see the example.

Example



```
const tencentcloud = require("../../../../../tencentcloud-sdk-nodejs-intl-en");

// Import the client models of the corresponding product module.
const CvmClient = tencentcloud.cvm.v20170312.Client;
const models = tencentcloud.cvm.v20170312.Models;

const Credential = tencentcloud.common.Credential;

// Instantiate an authentication object. The Tencent Cloud account secretId and secretKey
let cred = new Credential("secretId", "secretKey");
```

```
// Instantiate the client object to request the product (with CVM as an example).
let client = new CvmClient(cred, "ap-shanghai");

// Instantiate a request object.
let req = new models.DescribeZonesRequest();

// Call the API you want to access through the client object; you need to pass in t
client.DescribeZones(req, function(errMsg, response) {
  // The request is returned exceptionally and the exception information is print
  if (errMsg) {
    console.log(errMsg);
    return;
  }
  // The request is returned normally and the response object is printed.
  console.log(response.toJson_string());
});
```

More Examples

You can find more detailed examples in the examples directory of the [GitHub](#) repository.

C++

Last updated : 2021-02-26 17:07:02

Overview

Welcome to Tencent Cloud Software Development Kit (SDK) 3.0, a companion tool for the TencentCloud API 3.0 platform. SDK 3.0 is unified and features the same SDK usage, API call methods, error codes, and return packet formats for different programming languages.

This document describes how to use, debug, and connect to TencentCloud APIs with SDK 3.0 for C++ as an example.

This version currently supports various [Tencent Cloud services](#) such as CVM, VPC, and CBS and will support more services in the future.

Dependent Environment

The compiler for C++ v11 or above is required, i.e., GCC v4.8 or above. Currently, only the Linux installation environment is supported.

Get the security credential, which consists of `SecretId` and `SecretKey`. `SecretId` is used to identify the API requester, and `SecretKey` is a key used for signature string encryption and authentication by the server. You can get them on the [API Key Management](#) page as shown below:

Manage API Key

Safety Warning

- 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 keep the key safe and change it regularly.
- Please do not upload or share your key information by any means (such as GitHub). Once leaked to external channels, it may cause significant loss of your cloud assets
- There are security risks if you call the TencentCloud API using a low-version TLS protocol. TLS 1.2 or later is recommended.

Usage Notes

- The API Keys is used to generate a signature when you call the Tencent Cloud API. Check the algorithm for generating a signature.
- Your API key represents your account identity and permissions, and acts as your login password. Do not disclose it to others.
- Last Access Time is when this API key last accessed the TencentCloud API to use the Last Accessed Service in last 30 days. If no access records exist or the most recent records are from more than 30 days ago, no records will be displayed. The records are from CloudAudit. See the list of services and APIs supported by CloudAudit.

[Create Key](#)

APPID	Key	Creation Date	Last Access Time	Last Accessed Service	Status	Action
[redacted]	SecretId: [redacted] SecretKey: ***** Show	2019-06-27 17:25:38	-	-	Enabled	Disable

Note:

Your security credential represents your account identity and granted permissions, which is equivalent to your login password. Do not disclose it to others.

Get the calling address (endpoint), which is generally in the format of `*.tencentcloudapi.com` and varies by service. For example, the endpoint of CVM is `cvm.tencentcloudapi.com`. For specific endpoints, please see the [API documentation](#) of the corresponding service.

Installation

Building SDK from source code

1. Go to the [GitHub code hosting page](#) to download the latest source code package.
2. Enter the SDK and create necessary build files. Here, `path/to/` represents the actual path of the `tencentcloud-sdk-cpp` package.



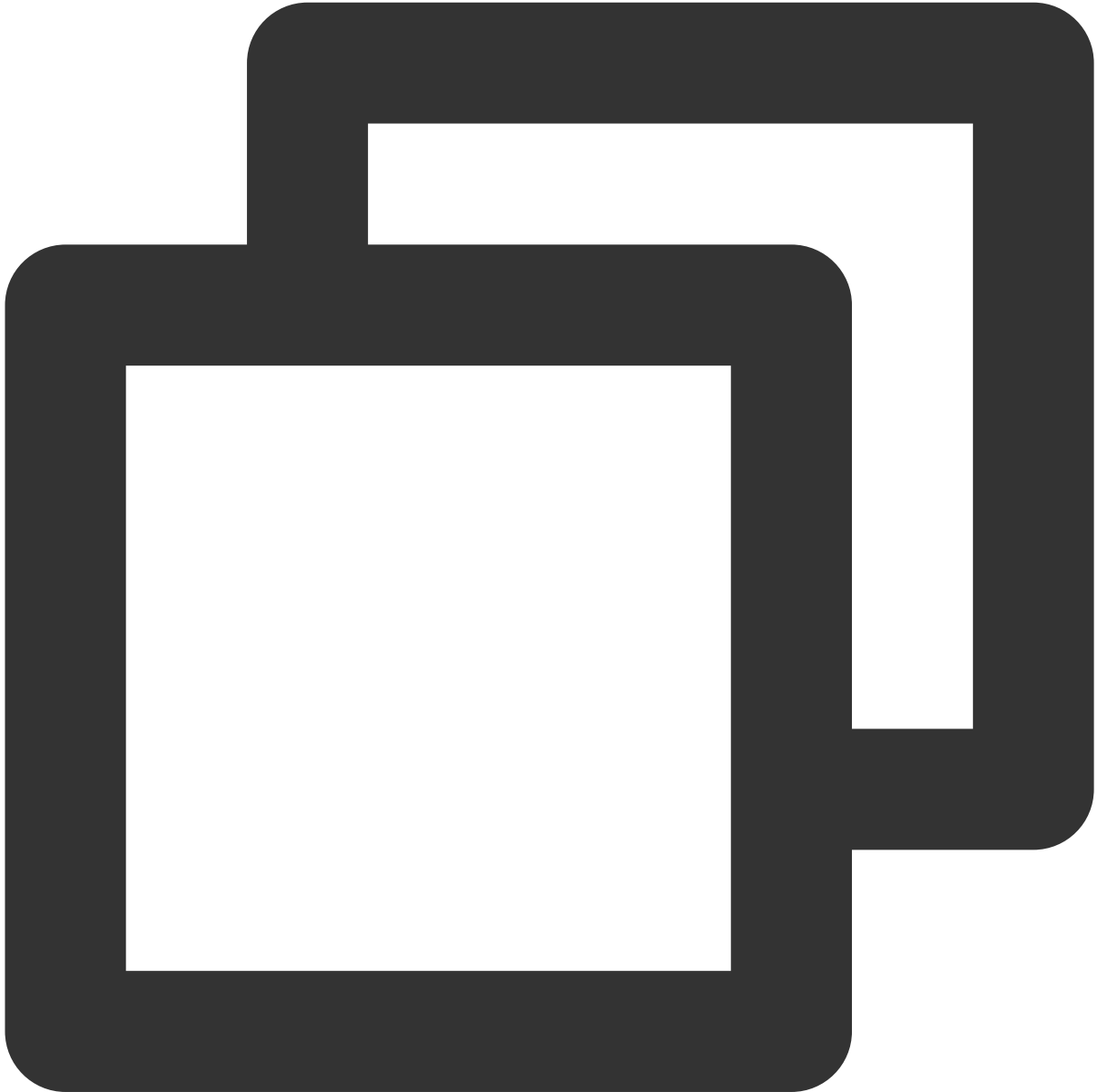
```
cd <path/to/tencentcloud-sdk-cpp>  
mkdir sdk_build  
cd sdk_build  
cmake ..  
make  
sudo make install
```

Example

Note:

The demo cannot be run directly. You need to replace the key and other information with your real information. It is better to configure them in environment variables to avoid exposing them in the code.

The following describes how to call the `DescribeInstances` API of CVM.



```
#include <tencentcloud/core/TencentCloud.h>
#include <tencentcloud/core/profile/HttpProfile.h>
#include <tencentcloud/core/profile/ClientProfile.h>
#include <tencentcloud/core/Credential.h>
#include <tencentcloud/core/NetworkProxy.h>
#include <tencentcloud/core/AsyncCallerContext.h>
```

```
#include <tencentcloud/cvm/v20170312/CvmClient.h>
#include <tencentcloud/cvm/v20170312/model/DescribeInstancesRequest.h>
#include <tencentcloud/cvm/v20170312/model/DescribeInstancesResponse.h>
#include <tencentcloud/cvm/v20170312/model/Instance.h>

#include <iostream>
#include <string>

using namespace TencentCloud;
using namespace TencentCloud::Cvm::V20170312;
using namespace TencentCloud::Cvm::V20170312::Model;
using namespace std;

int main()
{
    TencentCloud::InitAPI();

    // Use the SDK
    // Instantiate an authentication object. Pass in `secretId` and `secretKey` of
    string secretId = "<your secret id>";
    string secretKey = "<your secret key>";
    Credential cred = Credential(secretId, secretKey);

    // (Optional) Instantiate an HTTP option
    HttpProfile httpProfile = HttpProfile();
    httpProfile.SetKeepAlive(true); // Specify whether to enable the keepalive fea
    httpProfile.SetEndpoint("cvm.ap-guangzhou.tencentcloudapi.com"); // Specify th
    httpProfile.SetReqTimeout(30); // Specify the request timeout value in seconds
    httpProfile.SetConnectTimeout(30); // Specify the response timeout value in sec

    ClientProfile clientProfile = ClientProfile(httpProfile);

    DescribeInstancesRequest req = DescribeInstancesRequest();
    req.SetOffset(0);
    req.SetLimit(5);

    CvmClient cvm_client = CvmClient(cred, "ap-guangzhou", clientProfile);

    // Set proxy
    // NetworkProxy proxy = NetworkProxy(NetworkProxy::Type::HTTP, "localhost.proxy
    // cvm_client.SetNetworkProxy(proxy);

    auto outcome = cvm_client.DescribeInstances(req);
    if (!outcome.IsSuccess())
    {
        cout << outcome.GetError().PrintAll() << endl;
        TencentCloud::ShutdownAPI();
    }
}
```

```
        return -1;
    }
    DescribeInstancesResponse rsp = outcome.GetResult();
    cout<<"RequestId="<<rsp.GetRequestId()<<endl;
    cout<<"TotalCount="<<rsp.GetTotalCount()<<endl;
    if (rsp.InstanceSetHasBeenSet())
    {
        vector<Instance> instanceSet = rsp.GetInstanceSet();
        for (auto itr=instanceSet.begin(); itr!=instanceSet.end(); ++itr)
        {
            cout<<(*itr).GetPlacement().GetZone()<<endl;
        }
    }

    TencentCloud::ShutdownAPI();

    return 0;
}
```

Compile and run the demo code:



```
cd example/cvm/v20170312
mkdir build
cd build
cmake ..
make
./DescribeInstances
```

If the system returns an error indicating that a dynamic library is not found, you can specify the path to the dynamic library. For example, if the `libtencentcloud-sdk-cpp-core.so` library is installed in `/usr/local/lib`, then add the following line:



```
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH
./DescribeInstances
```

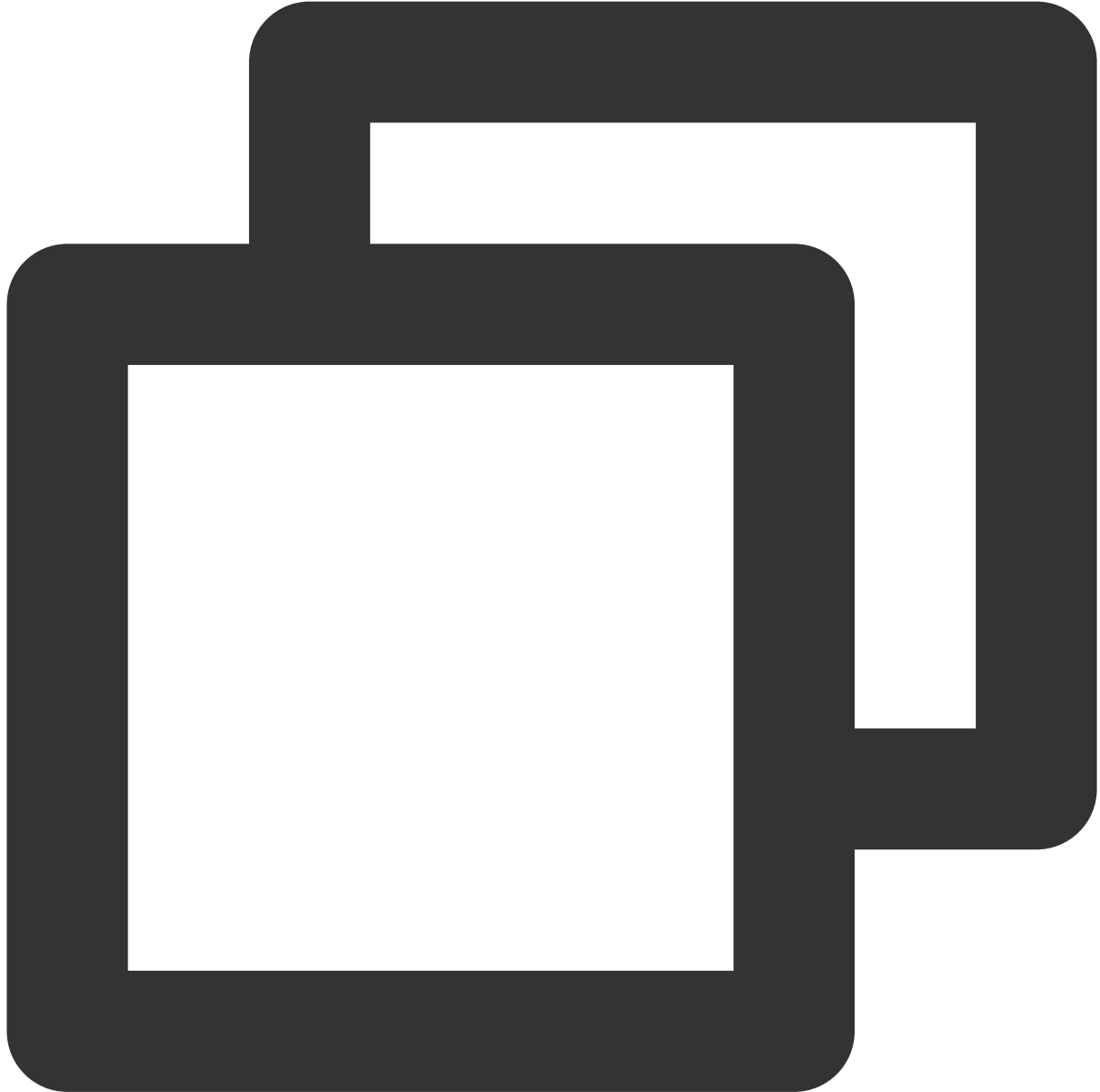
Note:

For more examples, please see the `example` directory.

Unit Testing

Creating dependent library for gtest

The sample installation code is as follows:



```
git clone https://github.com/google/googletest
cd googletest
cmake CMakeLists.txt
make
```

Copy the generated `libgtest.a` and `libgtest_main.a` static libraries and the header file of gtest to the system directory.

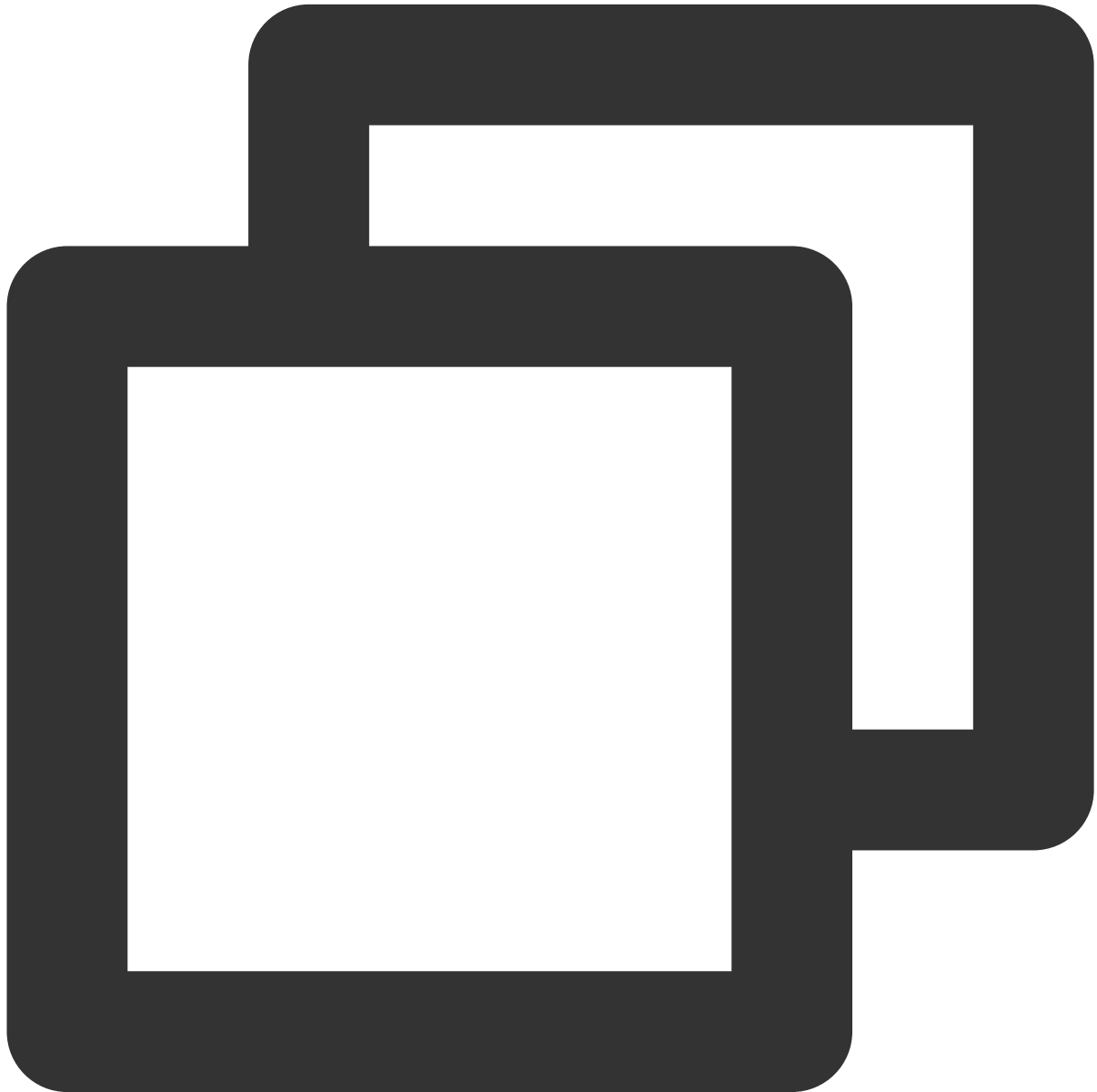
Configuring environment variables

TENCENTCLOUD_SECRET_ID: key ID

TENCENTCLOUD_SECRET_KEY: key

Testing

Run the following script:



```
sh function_test.sh
```

Relevant Configuration

Proxy

If you use the SDK to call an API in a proxy environment, you need to set the system environment variable

`https_proxy` (as shown in the sample code); otherwise, it may not be called normally, and a connection timeout exception will be thrown.