

APIGUIDE 文档



目录

目录	2
API调用视频指南	3
初见百度智能云API	3
如何阅读API文档	3
从零开始用Python调用API接口	3
用SDK对云产品进行管控	3
了解API	3
API文字版入门指南	4

API调用视频指南

初见百度智能云API

本视频面向初次接触百度智能云API的开发者，向您简单介绍API是什么、API的优势在哪里，以及借助API可以做到哪些事情。

如果您对本视频的内容有所疑问，或者希望能够推出更多其他方面的视频，欢迎您通过右侧的“[文档反馈](#)”提出您的建议。

如何阅读API文档

本视频借助Postman工具，向您展示了调用一次API需要哪些必要的信息，以及这些信息该如何从文档中取得。

tips：

1. Pre-request Script 是直接把[add_authorization.js](#)里边的内容复制进去就行，不需要对内容进行改动
2. 需要将sample.json文件导入环境，并配置AK/SK
3. header里需要手动配置一下域名
4. 如果确定自己的配置完全正确但仍然无法正常调用API接口，可以尝试升级Postman版本

如果您对本视频的内容有所疑问，或者希望能够推出更多其他方面的视频，欢迎您通过右侧的“[文档反馈](#)”提出您的建议。

从零开始用Python调用API接口

本视频将为您详细介绍列百度智能云鉴权认证机制，并从第一行代码开始，为您展示利用Python调用API修改云服务器BCC实例名称的全过程。

需要说明的是，本视频仅完成了调用API所需要的必要工作，如果您对鉴权认证有更多的要求，请参阅[鉴权认证机制](#)。

本视频的示例代码供您参考。[下载地址](#)

如果您对本视频的内容有所疑问，或者希望能够推出更多其他方面的视频，欢迎您通过右侧的“[文档反馈](#)”提出您的建议。

用SDK对云产品进行管控

本视频为您展示了如何利用百度智能云官方提供的Python SDK对云服务器BCC做各种管控操作。

如果您对本视频的内容有所疑问，或者希望能够推出更多其他方面的视频，欢迎您通过右侧的“[文档反馈](#)”提出您的建议。

了解API

🔗 API新手入门操作

视频简介：非常适合掌握百度智能云API的入门操作、以及初学者理解签名算法。



🔗 用postman调用百度智能云API

视频简介：借助Postman完成一次对BCC的API接口调用，过程中讲解产品API文档的阅读方法和API相关知识



如果您对本视频的内容有所疑问，或者希望能够推出更多其他方面的视频，欢迎您通过右侧的“文档反馈”提出您的建议。

API文字版入门指南

前言

云API是百度智能云开放能力的基础，借助API可以极大的节省人力成本，提高效率，用更高效的方式完成对云上资源的管控。对于开发者来说，使用云 API 完成一些重复性强的工作可以极大节约时间和精力；除此之外，API还有便于组合能力、便于自动化、扩展性强、对系统要求低等优点。

也许你已经了解过API和HTTP相关的知识概念，但刚刚使用百度智能云产品，不知道该从哪里开始API的调用工作。本文将带您理解百度智能云API签名算法，并以**创建BCC云服务器实例**为例，让您掌握调用百度智能云API的方法。

本文可以看做是鉴权认证机制文档的精简提炼版，更关注流程，忽略了一些具体的要求和细节。如果您在本文中遇到需要进一步了解的内容，请到[鉴权认证机制](#)中进行查阅。

要调用API，我们需要先[获取API请求的结构和参数](#)，了解API请求应该如何组织、以及含有哪些信息；再构造[认证字符串](#)；将认证字符串以及其他信息按要求填到API请求的各个位置后，即可发送API请求，并接收服务端发回的响应。

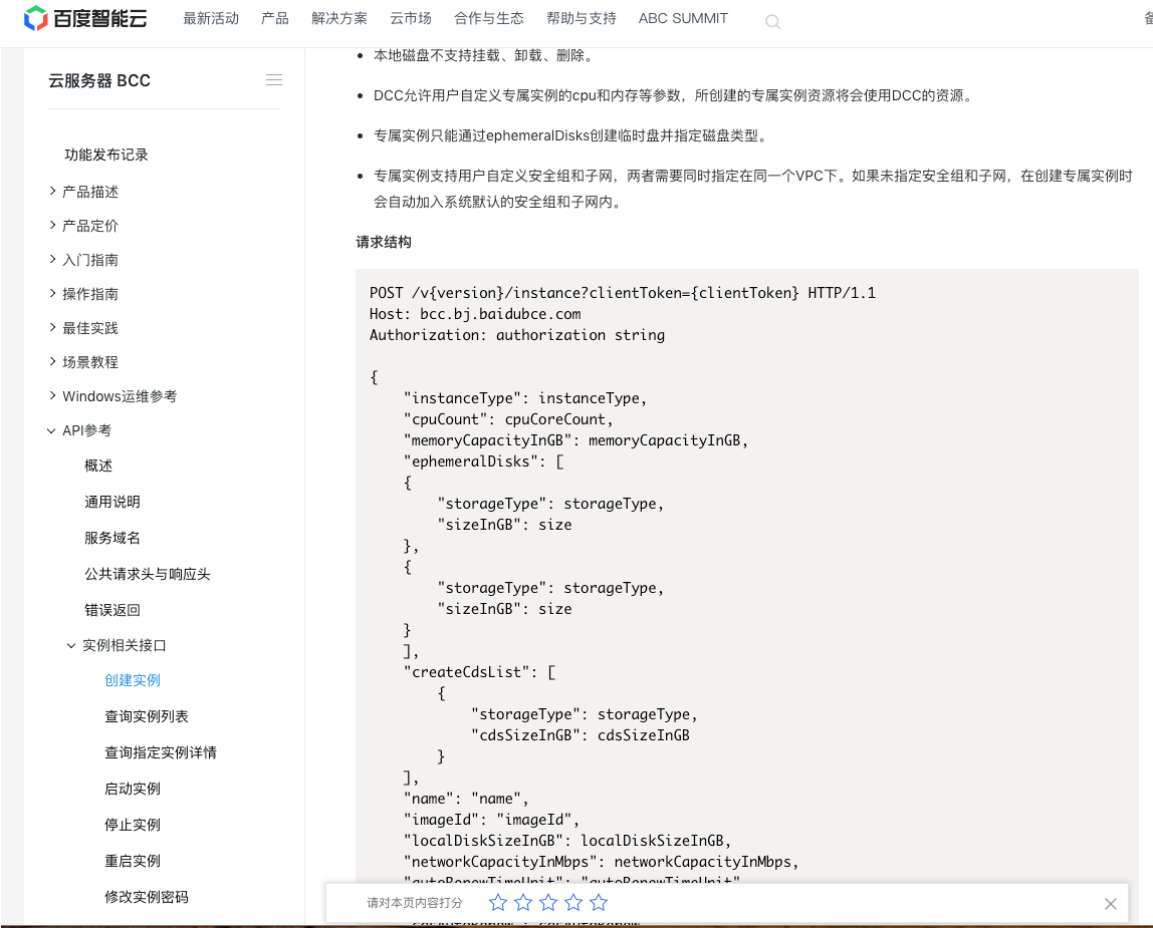
获取API请求的结构和参数

全篇以 **云服务器BCC创建实例**为例，完成一遍完整的API调用过程。

BCC 是什么？

云服务器BCC（Baidu Cloud Compute）是百度智能云可弹性伸缩的计算服务。管理方式比物理服务器更简单高效，API可以实现批量创建实例、释放任意多台云服务器实例，提升运维效率。更多详情请参考 [BCC API文档](#)。

要获取调用该接口所需要的信息，我们要先查阅该接口的API文档。登录百度智能云官网，在顶部导航栏中选择“帮助与支持>文档”，选择云服务器BCC，在文档左侧导航栏里选择“API参考>实例相关接口>[创建实例](#)”。



接口文档中的请求结构部分向我们展示了待发送的API请求的请求方法，URL、请求头和请求体该分别如何进行组织，而请求参数向我们介绍了API请求里的各个变量该如何进行取值。

参数位置	说明
方法	请求的方法，包括PUT、 POST、 GET、 DELETE等。具体取哪个方法需要以接口文档的请求结构和请求示例为准。
URL绝对路径 (URL参数)	指整段URL里用/分隔的部分，如本例中的 /v2/instance,v2是API版本号
查询字符串 (Query参数)	指整段URL里 ? 后边的部分。如果查询字符串中含有多组参数，用&连接。
请求头 (header参数)	大多数API请求都需要提交一些公共的内容（必要的有服务域名host和认证字符串Authorization，认证字符串的生成步骤见下方构造认证字符串），将这些内容统一放在请求头header里，就被称为公共请求头。
请求体 (RequestBody参数)	填写待创建的BCC实例的各项参数，请求体使用JSON格式

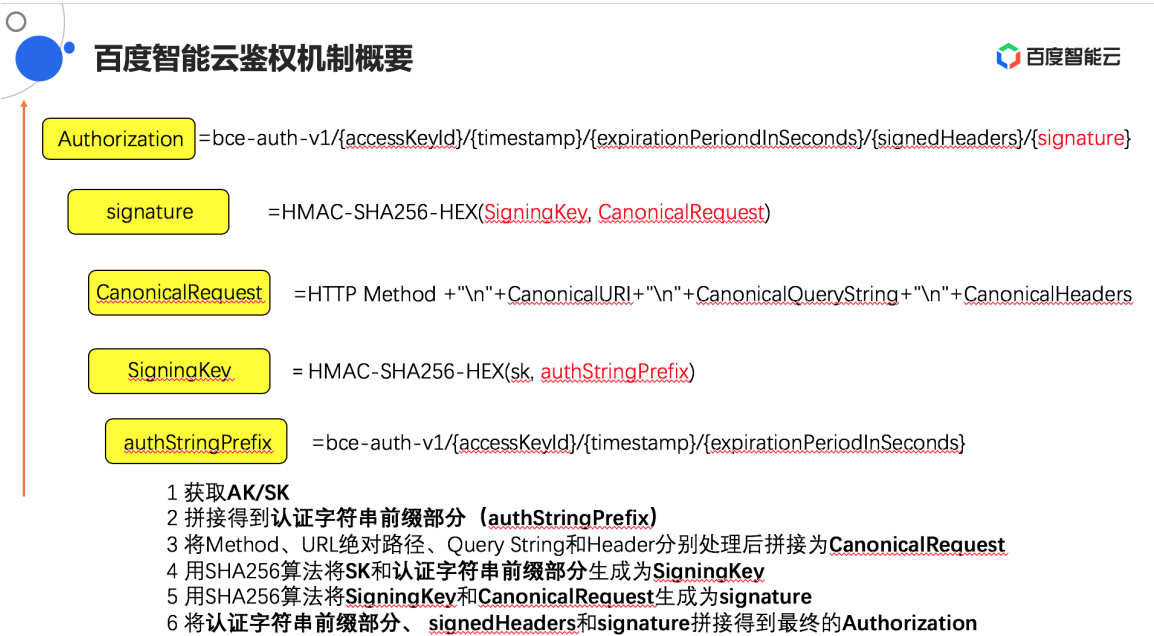
构造认证字符串

API请求中的大部分信息只需要您根据请求参数中的要求填好即可，比较容易获取。但请求头中的认证字符串Authorization需要利用众多参数，经过多次运算和拼接得到。

认证字符串的目的主要是为了对API调用发起者进行身份验证，同时也有防止非法篡改，防止重放攻击的作用。为了生成认证字符串，我们需要如下信息：

参数名称	参数解释
accessKeyId	简称AK，相当于您调用API时的用户名，获取方式见 获取AK/SK
SecretAccessKey	简称SK，相当于您调用API时的密码，获取方式见 获取AK/SK
authStringPrefix	认证字符串前缀，由"bce-auth-v1"、{accessKeyId}、{timestamp}、{expirationPeriodInSeconds}用/拼接而成。三个参数的取值说明详见 生成认证字符串
SigningKey	由SK和authStringPrefix经过HMAC-SHA256-HEX运算得到
CanonicalRequest	由请求方法、处理过的URL绝对路径、处理过的QueryString、处理过的header用换行符"\n"拼接而成。处理方法详见 生成canonicalrequest
signature	由SigningKey和CanonicalRequest经过HMAC-SHA256-HEX运算得到
Authorization	认证字符串，是鉴权认证工作的最终成果。由authStringPrefix、signedHeaders和signature用/拼接而成
signedHeaders	签名算法中涉及到的HTTP头域列表，也就是说，让哪些头域参与签名运算是可以由调用者自行决定的（host必须参与）。具体参见 CanonicalHeaders

综上，生成Authorization的过程可以用下面这张图来表示：



认证字符串构造完毕后，将其放入请求头，就完成了鉴权认证工作。

🔗 客户端发送API请求并接收响应

在编程环境中将所有信息都准备好之后，即可发送API请求。发送方法可以参考[用Python完成API接口调用](#)。

将API发送出去后，分两种情况：

- 如果API请求的内容正确无误，对于有返回值的API接口来说，会收到API请求所要求的返回值信息；对于没有返回值的API接口来说，只会收到状态码和响应头，响应体为空。
- 如果API请求内容存在问题，则会收到含有错误码和错误说明的响应。您可以根据错误码对API请求的各个部分做检查，排除问题。

🔗 Python示例讲解

官网帮助中心提供了Python编程的签名函数方法，以下讲解是在官网demo基础上增加请求函数，完成调用BCC API创建实例。

代码下载

- 1、demo代码示例请到官网下载 [Python demo签名示例](#)。

2、本文使用的实操代码 [点击下载](#)。

前置条件

- 1、已注册百度账号
- 2、实名认证
- 3、购买实例前需保证账户无欠款，且保证账户余额和可用代金券总和大于或等于100元

数据准备

- 1、获取AK/SK，参见 [获取AK/SK](#)
- 2、创建BCC实例所需数据，参见API参考文档 [创建BCC实例](#)

说明：下载官网demo的Php示例、Java示例、Javascript示例请访问 [帮助中心-》相关参考-》鉴权认证-》示例下载](#)

配置说明

本次举例中全部使用指定header_to_sign方法进行配置，实现创建BCC实例。

操作步骤

- 1、在官网demo示例中，要导入requests等模块，以便支持请求函数和幂等性函数。

```
import hashlib
import hmac
import string
import datetime
import requests
import json
import random
```

demo示例中使用的函数有：保存AK/SK、正规化字符串、生成规范时间戳、生成规范URI、生成规范查询字符串。

- 2、生成规范header。

需注意：当不指定header_to_sign时，默认使用如下5项参与签名算法。

```
# 生成规范header
def get_canonical_headers(headers, headers_to_sign=None):
    headers = headers or {}

    # 没有指定header_to_sign的情况下，默认使用：
    # 1.host
    # 2.content-md5
    # 3.content-length
    # 4.content-type
    # 5.所有以x-bce-开头的header项
    # 生成规范header
```

- 3、以下开始均使用指定header_to_sign方法进行配置。

- 4、签名主算法模块。

其中timestamp=0, expiration_in_seconds=18000, headers_to_sign=None已经给出默认值。

```
# 签名主算法
def sign(credentials, http_method, path, headers, params,
          timestamp=0, expiration_in_seconds=18000, headers_to_sign=None):
    headers = headers or {}
    params = params or {}
```

本次举例使用非默认配置，要指定header_to_sign，在步骤7里，完成最终签名字符串的拼接。

```
# 7. 拼接最终签名结果串
if headers_to_sign:
    # 指定header to sign
    result = '%s/%s/%s' % (sign_key_info, ';'.join(headers_to_sign), sign_result)
else:
    result = '%s/host/%s' % (sign_key_info, sign_result)
return result
```

5、使用幂等性函数。

可以保证创建BCC时，避免同一个请求被重复调用。

```
def clientToken():# 幂等性函数
    """
    The alternative method to generate the random string for client_token
    if the optional parameter client_token is not specified by the user.
    :return:
    :rtype string
    """
    client_token = ''.join(random.sample(string.ascii_letters + string.digits, 36))
    return client_token
```

6、主工程中填写配置数据。

在【数据准备】获得的AK、Sk、请求方法、请求路径，请求头域，幂等性，填入主工程响应参数中。

```
if __name__ == "__main__":
    # 填写AK SK
    credentials = BceCredentials("AK", "SK")
    # API接口的请求方法
    http_method = "POST"
    # 接口请求路径
    path = "/v2/instance"
    # 接口请求的header头
    headers = {
        "host": "bcc.bj.baidubce.com",
        "content-type": "application/json; charset=utf-8",
        "x-bce-date": "2019-04-22T06:06:49Z"
    }
    # 接口请求参数
    params = {
        "clientToken": clientToken()
    }
```

7、构造body数据。

在请求体body数据构造中，为了是举例简单易懂，仅使用创建BCC的最小配置，包括 实例类型、CPU核数、实例名、内存容量、镜像ID，付费方式，可用区。

```
# 接口请求的body数据
body = {
    "instanceType": "N1", # 实例类型
    "cpuCount": 1, # 待创建虚拟机实例的CPU核数
    "memoryCapacityInGB": 1, # 待创建虚拟机实例的内存容量，单位GB
    "name": "test001",
    "imageId": "m-oVrsHBHm", # 待创建虚拟机实例的镜像ID，可通过调用查询镜像列表接口选择获取所需镜像ID。
    "billing": {
        "paymentTiming": "Postpaid", # 付费方式，付费方式，包括预支付 (Prepaid) 和后支付 (Postpaid)
    },
    "zoneName": "cn-bj-b" # 需要通过/v2/zone接口获取可用区字段
}
```

8、本步骤是重点，指定header_to_sign参与鉴权编码。

通过查看文档中给出的请求参数，从中挑选必选和部分可选公共头，填进指定header_to_sign中，参与鉴权编码。这里需要理解的是，选择哪些Header进行编码并不会影响API的功能，但是如果选择太少则可能遭到中间人攻击，但一定必须包含host，如本例中使用host和x-bce-date。

```
# 设置参与鉴权编码的header，即headers_to_sign，至少包含host，百度智能云API的唯一要求是Host域必须被编码
headers_to_sign = {"host", "x-bce-date"}
```

9、继续配置到期时间以及时间戳。

时间戳是生成鉴权生效的开始时间，可以取当前的时间点，或需要的时间点。

```
# 设置到期时间，默认1800s
expiration_in_seconds = 1800000
# 设置参与鉴权的时间戳
timestamp = 1555913209
# 生成鉴权字符串
```

10、在构造和生成鉴权字符串后，加到头域中的认证字段。

```
# headers字典中需要加上鉴权字符串authorization的请求头
headers['authorization'] = result
```

11、接下来构造request请求和拼接URL，以及构造发起请求。

```
request = {
    'method': http_method,
    'uri': path,
    'headers': headers,
    'params': params
}
# 拼接接口的url地址
url = 'http://%s%s?clientToken=%s' % (headers['host'], request['uri'], params['clientToken'])
```

执行结果

现在执行一次请求，在执行结果中，http状态为200，表明已经成功创建BCC实例。同样可以在百度智能云控制台查看创建结果。

```
/Users/baidu/PycharmProjects/codedemo/venv/bin/python /Users/baidu/Downloads/sample_code/官网demo/CreateBcc.py
bce-auth-v1/020693e06ad848da8fe5aacfdb680217/2019-04-22T06:06:49Z/18000000/x-bce-date;host/bcfc2db8492e3ebf0f31b69101078322cb7d76f47e0d3485f984cb7ccc0ff47
url: http://bcc.bj.baidubce.com/v2/instance?clientToken=ig7nsabuyj3494AcZUfNmbjPF36ogRd13ChI
status: 200
headers: {'Content-Length': '30', 'X-Application-Context': 'application:8688', 'Server': 'nginx/1.14.0', 'Connection': 'keep-alive', 'Date': 'Thu, 16 May 2019 05:43:54 GMT', 'X-Bce-Request-
response: {"instanceIds":["i-wcxK5Hkh"]}
```

登录百度智能云控制台，选择区域北京，在实例列表页可以查看到刚刚创建的BCC云服务器实例，正在创建中。创建成功后，便可以使用实例资源。



后续操作

- 查询BCC实例
- 停止BCC实例
- 释放BCC实例

在创建BCC代码基础上，已经完成了签名和认证的关键操作，只需要根据API文档提供的数据，就可以自学调用BCC实例的查询、停止、释放。

🔗 参考与反馈

参考资料	说明
API平台	百度智能云所有产品API文档入口、开发者工具、新手入门
如何查找自己的AK/SK	在控制台申请和管理自己的访问密钥
鉴权认证机制	鉴权介绍、使用方法、各种语言签名函数下载、常见错误故障处理
区域选择说明	有关每个服务所支持的区域信息
签名认证错误FAQ	介绍常见签名认证错误排查方法

更多帮助与支持，请将意见填写到产品API参考文档页（如下图），或提交[工单](#)咨询。



🔗 API调用常见错误码

错误码	错误消息	HTTP状态码	描述
AccessDenied	Access denied.	403Forbidden	无权限访问对应的资源。
InappropriateJSON	The JSON you provided was well-formed and valid, but not appropriate for this operation.	400Bad Request	请求中的JSON格式正确，但语义上不符合要求。如缺少某个必需项，或值类型不匹配等。出于兼容性考虑，对于所有无法识别的项应直接忽略，不应该返回这个错误。
InternalServerError	We encountered an internal error Please try again.	500Internal Server Error	所有未定义的其他错误。在有明确对应的其他类型的错误时（包括通用的和服务自定义的）不应该使用。
InvalidAccessKeyId	The Access Key ID you provided does not exist in our records.	403Forbidden	Access Key ID不存在。
InvalidHTTPAuthHeader	The Access Key ID you provided does not exist in our records.	400Bad Request	Authorization头域格式错误。
InvalidHTTPRequest	There was an error in the body of your HTTP request.	400Bad Request	HTTP body格式错误。例如不符合指定的Encoding等。
InvalidURI	Could not parse the specified URI.	400Bad Request	URI形式不正确。例如一些服务定义的关键词不匹配等。对于ID不匹配的问题，应定义更加具体的错误码，如NoSuchKey。
MalformedJSON	The JSON you provided was not well-formed.	400Bad Request	JSON格式不合法。
InvalidVersion	The API version specified was invalid.	404NotFound	URI的版本号不合法。
OntInRe		403Forbidden	

Subscription required	A subscription for the service is required.	403 Forbidden	没有开通对应的服务。
PreconditionFailed	The specified If-Match header doesn't match the ETag header.	412 PreconditionFailed	详见Etag。
Request Expired	Request has expired. Timestamp date is <Data>.	400 Bad Request	请求超时。要改成x-bce-date。若请求中只有Date，需将Date转成datetime。
IdempotentParameterMismatch	The request uses the same client token as a previous, but non-identical request.	403 Forbidden	clientToken对应的API参数不一样。
SignatureDoesNotMatch	The request signature we calculated does not match the signature you provided. Check your Secret Access Key and signing method. Consult the service documentation for details.	400 Bad Request	Authorization头域中附带的签名和服务端验证不一致。