审批管理

产品文档

目录

审批管理 3
- 操作指南 3
  - 审批业务管理 3
  - 审批流程管理 4
- API文档 9
  - 租户端审批流服务（tapproval） 9
    - 版本（2022-05-18） 9
      - API 概览 9
      - 调用方式 10
        - 接口签名v1 10
        - 接口签名v3 17
        - 请求结构 26
        - 返回结果 27
        - 公共参数 30
      - 审批服务接口 32
        - 批量进行审批 32
        - 批量提交审批申请 34
        - 删除审批流 36
        - 修改审批流 37
        - 操作审批流状态 39
        - 查询审批流状态 41
        - 查询审批单详情 43
        - 获取租户端审批单详情 45
        - 查询待申请列表 47
        - 发送审批提醒 49
        - 发送审批验证码 50
        - 校验审批短信验证码 52
        - 撤销审批申请 54
      - 数据结构 56
      - 错误码 69

操作指南
审批业务管理

审批业务管理

审批业务里，会列出所有接入审批的业务名称

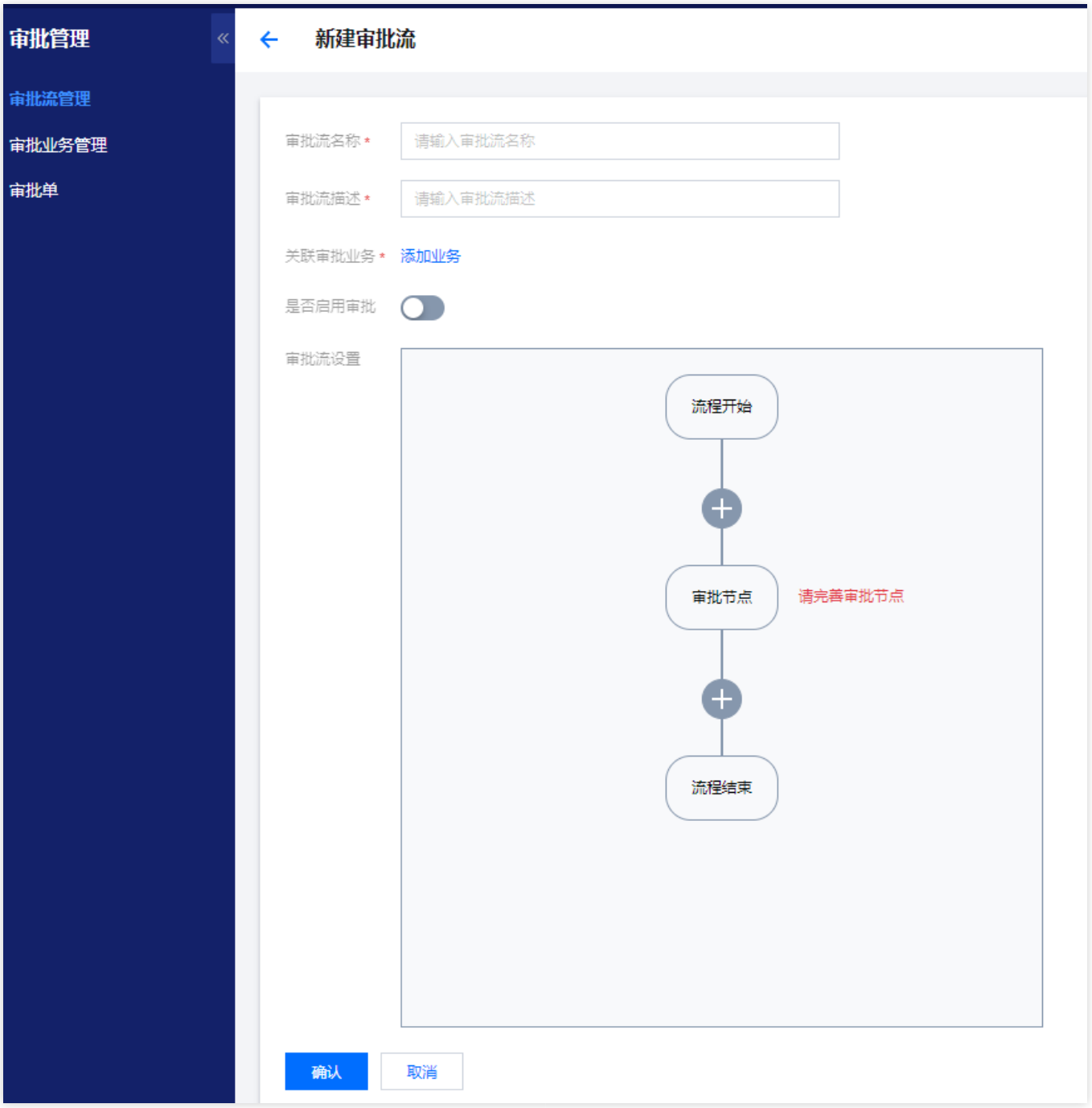
审批流程管理

审批流管理

审批流用于基于审批业务来定义一个审批流程。审批流程里定义了用户在　页面触发指定的业务时，需要由哪些审批人来审批。

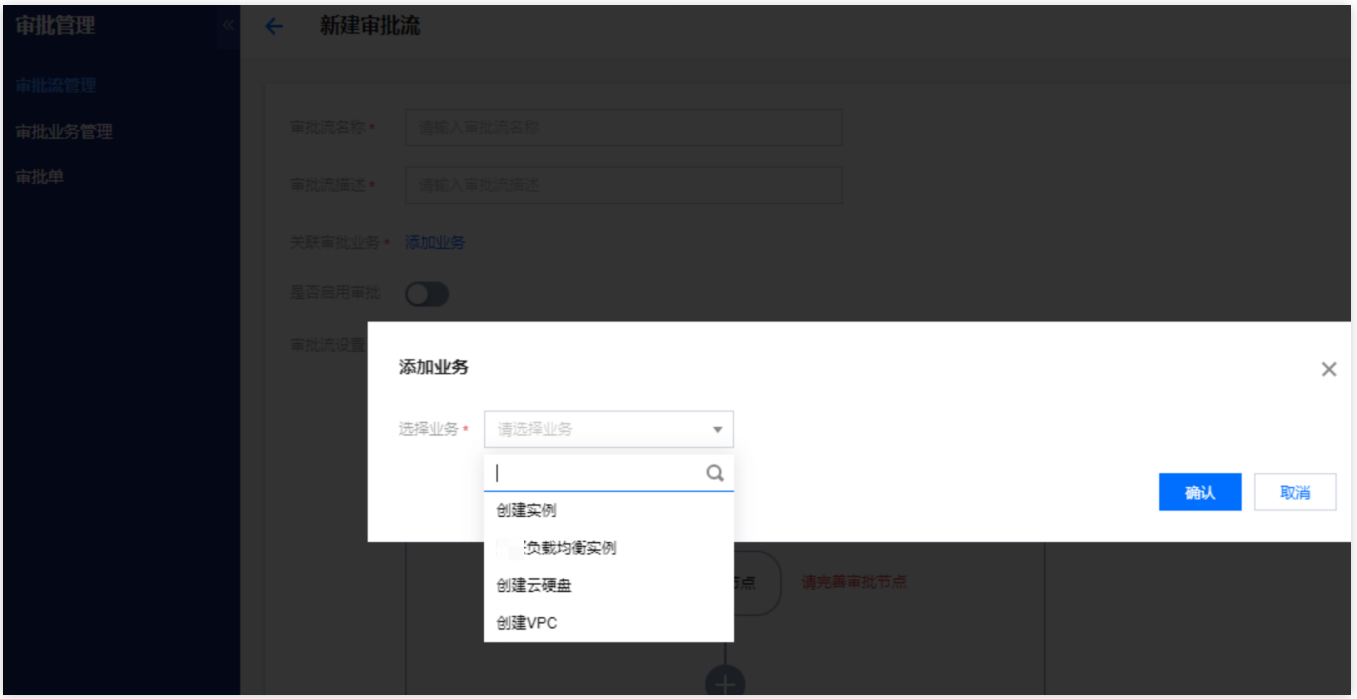
新建审批流

- 输入审批流名称和审批流描述

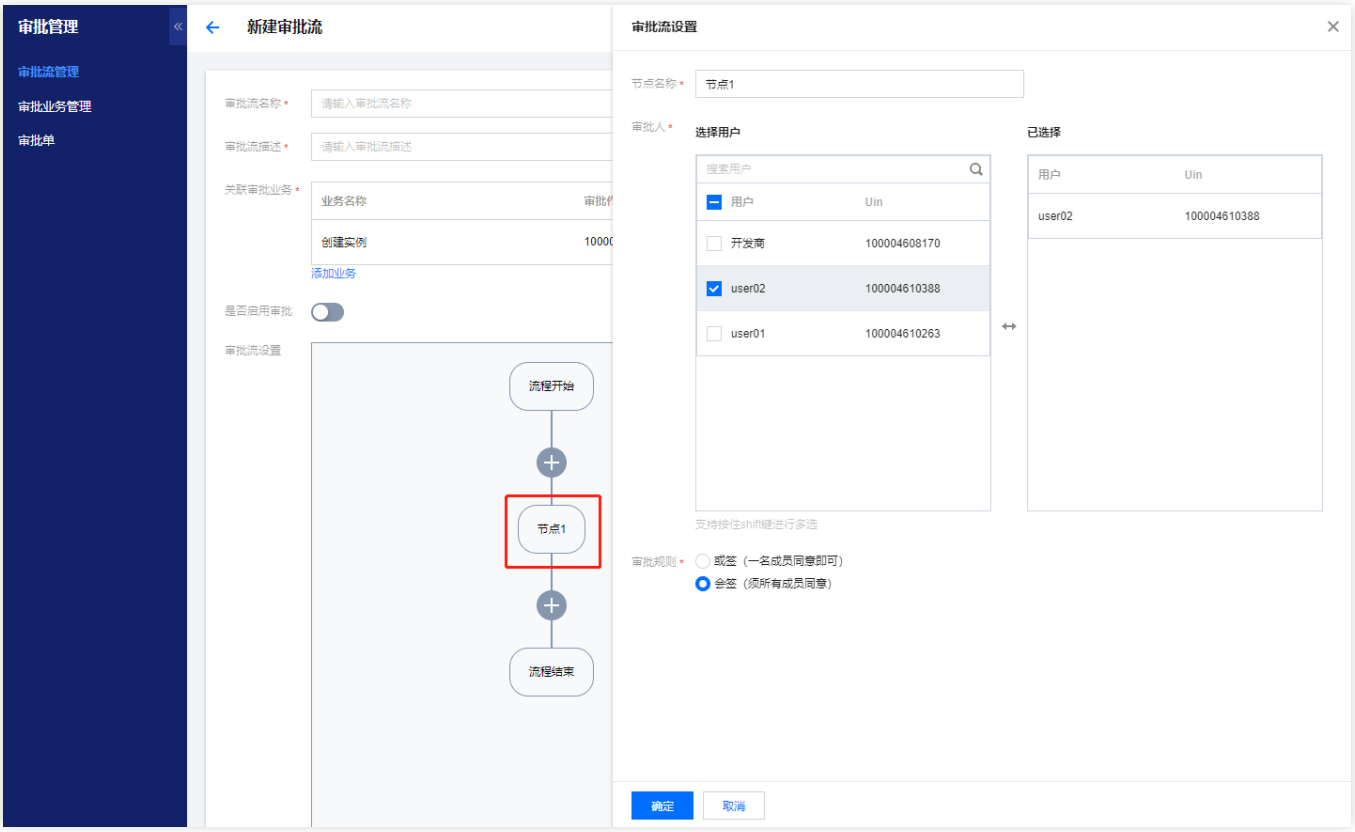


- 添加业务

  选择要审批的业务，当前入口创建的审批流，只对当前租户生效，对其他租户不生效



- 设置审批流
  可设置多个审批节点，目前最多支持设置5个审批节点。每个节点可设置多个审批人，审批人列表拉取的是当前租户下的用户列表。
  若运营端管理员基于某个业务定义了一个审批流，且审批流的作用范围包括租户A，租户A在租户端也基于该业务定义并开启了审批流，则租户A下的用户在页面触发该业务时，实际的审批流程是将租户A下该业务的审批流和运营端该业务的审批流拼接起来形成一个新的审批流，当租户A在页面触发该业务时，需先由租户A下该业务的审批人审批通过之后，再流转到运营端，由运营端该业务的审批人审批。
  举个例子：运营端基于“创建云服务器实例”定义并开启了一个审批流，审批作用范围包含租户A，审批人是运营端的子用户user01;租户A在租户端也基于“创建云服务器实例”定义并开启了一个审批流，审批人是租户A的子用户user02;此时，租户A下的某个子用户在创建云服务器实例时，会生成一个审批单，审批单先流转到租户A的子用户user02名下，若user02驳回该审批单，则整个审批单状态为驳回，云服务器无法被创建出来，若租户A的子用户user02同意该审批单，则单子会被流转到运营端的子用户user01下，若user01同意该单，则整个审批单的状态为通过，此时云服务器实例会被创建出来。



启用/禁用审批流

可在审批流列表里启用/禁用审批流。启用之后，该审批流作用范围内的用户在页面触发相应业务时，会生成审批单走审批流程；禁用后，该审批流会失效

查看所有审批单

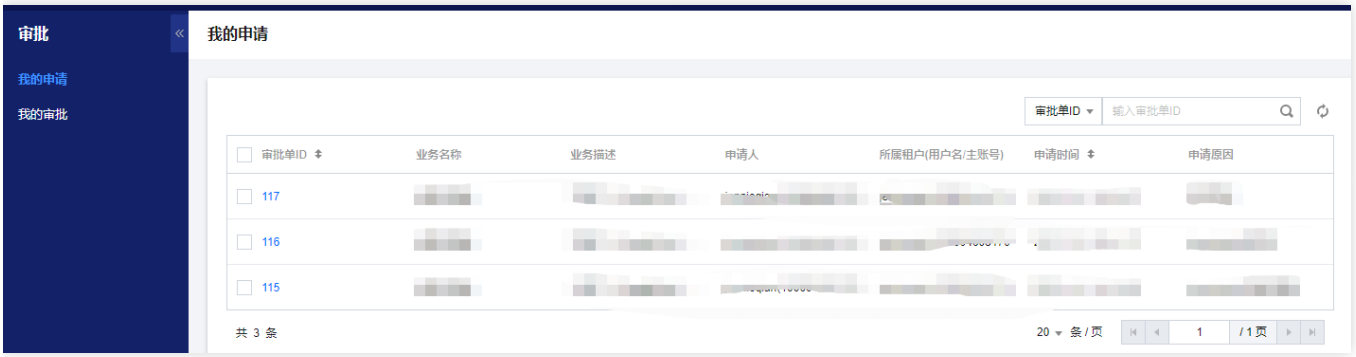
若给子用户关联审批管理全读写权限ApprovalFullAccess，则在审批管理下的审批单菜单下，可查看当前租户的所有审批单。

处理审批单

每个用户都都权限查看【我的审批】入口，该入口会列出当前待我处理的审批单和我已处理过的审批单，可选择同意或者驳回。

查看我的审批列表

每个用户都权限查看【我的申请】入口，该入口会列出待申请和已申请的所有审批单，可查看审批单详情和状态。



API文档
租户端审批流服务（tapproval）
版本（2022-05-18）
API 概览

API版本

V3

审批服务接口

接口名称	接口功能
BatchPerformApproval	批量进行审批
BatchSubmitApproval	批量提交审批申请
DeleteApprovalFlow	删除审批流
ModifyApprovalFlow	修改审批流
OperateFlowStatus	操作审批流状态
QueryApprovalFlowStatus	查询审批流状态
QueryCurrApprovalDetail	查询审批单详情
QueryCustomerApprovalDetail	获取租户端审批单详情
QueryPendingApprovalDoc	查询待申请列表
SendApprovalReminder	发送审批提醒
SendApprovalVerifyCode	发送审批验证码
VerifyApprovalSmsCode	校验审批短信验证码
WithdrawApplication	撤销审批申请

调用方式
接口签名v1

TCloudFinanceZone API 会对每个访问请求进行身份验证，即每个请求都需要在公共请求参数中包含签名信息（ Signature ）以验证请求者身份。
签名信息由安全凭证生成，安全凭证包括 SecretId 和 SecretKey；若用户还没有安全凭证，请前往云API密钥页面申请，否则无法调用云API接口。

1. 申请安全凭证

在第一次使用云API之前，请前往云API密钥页面申请安全凭证。
安全凭证包括 SecretId 和 SecretKey：

- SecretId 用于标识 API 调用者身份
- SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。
- 用户必须严格保管安全凭证，避免泄露。

申请安全凭证的具体步骤如下：

1. 登录TCloudFinanceZone管理中心控制台。
2. 前往云API密钥的控制台页面
3. 在云API密钥页面，点击【新建】即可以创建一对SecretId/SecretKey

注意：开发商帐号最多可以拥有两对 SecretId / SecretKey。

2. 生成签名串

有了安全凭证SecretId 和 SecretKey后，就可以生成签名串了。以下是生成签名串的详细过程：

假设用户的 SecretId 和 SecretKey 分别是：

- SecretId: AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE
- SecretKey: Gu5t9xGARNpq86cd98joQYCN3EXAMPLE

注意：这里只是示例，请根据用户实际申请的 SecretId 和 SecretKey 进行后续操作！

以云服务器查看实例列表(DescribeInstances)请求为例，当用户调用这一接口时，其请求参数可能如下:

参数名称	中文	参数值

参数名称	中文	参数值
Action	方法名	DescribeInstances
SecretId	密钥Id	AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE
Timestamp	当前时间戳	1465185768
Nonce	随机正整数	11886
Region	实例所在区域	shjr
InstanceIds.0	待查询的实例ID	ins-09dx96dg
Offset	偏移量	0
Limit	最大允许输出	20
Version	接口版本号	2017-03-12

2.1. 对参数排序

首先对所有请求参数按参数名的字典序（ ASCII 码）升序排序。注意：1）只按参数名进行排序，参数值保持对应即可，不参与比大小；2）按 ASCII 码比大小，如 InstanceIds.2 要排在 InstanceIds.12 后面，不是按字母表，也不是按数值。用户可以借助编程语言中的相关排序函数来实现这一功能，如 php 中的 ksort 函数。上述示例参数的排序结果如下:

```
{
    'Action' : 'DescribeInstances',
    'InstanceIds.0' : 'ins-09dx96dg',
    'Limit' : 20,
    'Nonce' : 11886,
    'Offset' : 0,
    'Region' : 'shjr',
    'SecretId' : 'AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE',
    'Timestamp' : 1465185768,
    'Version': '2017-03-12',
}
```

使用其它程序设计语言开发时，可对上面示例中的参数进行排序，得到的结果一致即可。

2.2. 拼接请求字符串

此步骤生成请求字符串。
将把上一步排序好的请求参数格式化成“参数名称”=“参数值”的形式，如对 Action 参数，其参数名称为 "Action" ，参数值为 "DescribeInstances" ，因此格式化后就为 Action=DescribeInstances 。
注意：“参数值”为原始值而非url编码后的值。

然后将格式化后的各个参数用"&"拼接在一起，最终生成的请求字符串为:

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=shjr&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12
```

2.3. 拼接签名原文字符串

此步骤生成签名原文字符串。
签名原文字符串由以下几个参数构成:

1. 请求方法: 支持 POST 和 GET 方式，这里使用 GET 请求，注意方法为全大写。
2. 请求主机:查看实例列表(DescribeInstances)的请求域名为：cvm.finance.cloud.tencent.com。实际的请求域名根据接口所属模块的不同而不同，详见各接口说明。
3. 请求路径: 当前版本云API的请求路径固定为 / 。
4. 请求字符串: 即上一步生成的请求字符串。

签名原文串的拼接规则为: 请求方法 + 请求主机 +请求路径 + ? + 请求字符串

示例的拼接结果为：

```
GETcvm.finance.cloud.tencent.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=shjr&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12
```

2.4. 生成签名串

此步骤生成签名串。
首先使用 HMAC-SHA1 算法对上一步中获得的签名原文字符串进行签名，然后将生成的签名串使用 Base64 进行编码，即可获得最终的签名串。

具体代码如下，以 PHP 语言为例:

```
$secretKey = 'Gu5t9xGARNpq86cd98joQYCN3EXAMPLE';
$srcStr = 'GETcvm.finance.cloud.tencent.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=shjr&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

最终得到的签名串为:

```
EliP9YW3pW28FpsEdkXt/+WcGeI=
```

使用其它程序设计语言开发时，可用上面示例中的原文进行签名验证，得到的签名串与例子中的一致即可。

3. 签名串编码

生成的签名串并不能直接作为请求参数，需要对其进行 URL 编码。

如上一步生成的签名串为 EliP9YW3pW28FpsEdkXt/+WcGeI= ，最终得到的签名串请求参数（Signature）为：EliP9YW3pW28FpsEdkXt%2f%2bWcGeI%3d，它将用于生成最终的请求 URL。

注意：如果用户的请求方法是 GET，或者请求方法为 POST 同时 Content-Type 为 application/x-www-form-urlencoded，则发送请求时所有请求参数的值均需要做 URL 编码，参数键和=符号不需要编码。非 ASCII 字符在 URL 编码前需要先以 UTF-8 进行编码。

注意：有些编程语言的 http 库会自动为所有参数进行 urlencode，在这种情况下，就不需要对签名串进行 URL 编码了，否则两次 URL 编码会导致签名失败。

注意：其他参数值也需要进行编码，编码采用 RFC 3986。使用 %XY 对特殊字符例如汉字进行百分比编码，其中“X”和“Y”为十六进制字符（0-9 和大写字母 A-F），使用小写将引发错误。

4. 签名失败

根据实际情况，存在以下签名失败的错误码，请根据实际情况处理

错误代码	错误描述
AuthFailure.SignatureExpire	签名过期
AuthFailure.SecretIdNotFound	密钥不存在
AuthFailure.SignatureFailure	签名错误
AuthFailure.TokenFailure	token 错误
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）

5. 签名演示

在实际调用 API 3.0 时，推荐使用配套的TCloudFinanceZone SDK 3.0 ，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 SDK 中心。当前支持的编程语言有：

- Python
- Java

- PHP
- Go
- Node

为了更清楚的解释签名过程，下面以实际编程语言为例，将上述的签名过程具体实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK 。

最终输出的 url 可能为： https://cvm.finance.cloud.tencent.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=shjr&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Signature=EliP9YW3pW28FpsEdkXt%2F%2BWcGeI%3D&Timestamp=1465185768&Version=2017-03-12

注意：由于示例中的密钥是虚构的，时间戳也不是系统当前时间，因此如果将此 url 在浏览器中打开或者用 curl 等命令调用时会返回鉴权错误：签名过期。为了得到一个可以正常返回的 url ，需要修改示例中的 SecretId 和 SecretKey 为真实的密钥，并使用系统当前时间戳作为 Timestamp 。

注意：在下面的示例中，不同编程语言，甚至同一语言每次执行得到的 url 可能都有所不同，表现为参数的顺序不同，但这并不影响正确性。只要所有参数都在，且签名计算正确即可。

注意：以下代码仅适用于 API 3.0，不能直接用于其他的签名流程，即使是旧版的 API ，由于存在细节差异也会导致签名计算错误，请以对应的实际文档为准。

Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class CloudAPIDemo {
    private final static String CHARSET = "UTF-8";

    public static String sign(String s, String key, String method) throws Exception {
        Mac mac = Mac.getInstance(method);
        SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
        mac.init(secretKeySpec);
        byte[] hash = mac.doFinal(s.getBytes(CHARSET));
        return DatatypeConverter.printBase64Binary(hash);
    }

    public static String getStringToSign(TreeMap<String, Object> params) {
        StringBuilder s2s = new StringBuilder("GETcvm.finance.cloud.tencent.com/?");
```

```
        // 签名时要求对参数进行字典排序，此处用TreeMap保证顺序
        for (String k : params.keySet()) {
            s2s.append(k).append("=").append(params.get(k).toString()).append("&");
        }
        return s2s.toString().substring(0, s2s.length() - 1);
    }

    public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException
{
        StringBuilder url = new StringBuilder("https://cvm.finance.cloud.tencent.com/?");
        // 实际请求的url中对参数顺序没有要求
        for (String k : params.keySet()) {
            // 需要对请求串进行urlencode，由于key都是英文字母，故此处仅对其value进行urlencode
            url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).app
end("&");
        }
        return url.toString().substring(0, url.length() - 1);
    }

    public static void main(String[] args) throws Exception {
        TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap可以自动排序
        // 实际调用时应当使用随机数，例如：params.put("Nonce", new Random().nextInt(java.lang.Intege
r.MAX_VALUE));
        params.put("Nonce", 11886); // 公共参数
        // 实际调用时应当使用系统当前时间，例如：  params.put("Timestamp", System.currentTimeMillis() /
1000);
        params.put("Timestamp", 1465185768); // 公共参数
        params.put("SecretId", "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"); // 公共参数
        params.put("Action", "DescribeInstances"); // 公共参数
        params.put("Version", "2017-03-12"); // 公共参数
        params.put("Region", "shjr"); // 公共参数
        params.put("Limit", 20); // 业务参数
        params.put("Offset", 0); // 业务参数
        params.put("InstanceIds.0", "ins-09dx96dg"); // 业务参数
        params.put("Signature", sign(getStringToSign(params), "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE
", "HmacSHA1")); // 公共参数
        System.out.println(getUrl(params));
    }
}
```

Python

注意：如果是在 Python 2 环境中运行，需要先安装 requests 依赖包： pip install requests 。

```
# -*- coding: utf8 -*-
import base64
```

```
import hashlib
import hmac
import time

import requests

secret_id = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"
secret_key = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"

def get_string_to_sign(method, endpoint, params):
    s = method + endpoint + "/?"
    query_str = "&".join("%s=%s" % (k, params[k]) for k in sorted(params))
    return s + query_str

def sign_str(key, s, method):
    hmac_str = hmac.new(key.encode("utf8"), s.encode("utf8"), method).digest()
    return base64.b64encode(hmac_str)

if __name__ == '__main__':
    endpoint = "cvm.finance.cloud.tencent.com"
    data = {
        'Action' : 'DescribeInstances',
        'InstanceIds.0' : 'ins-09dx96dg',
        'Limit' : 20,
        'Nonce' : 11886,
        'Offset' : 0,
        'Region' : 'shjr',
        'SecretId' : secret_id,
        'Timestamp' : 1465185768, # int(time.time())
        'Version': '2017-03-12'
    }
    s = get_string_to_sign("GET", endpoint, data)
    data["Signature"] = sign_str(secret_key, s, hashlib.sha1)
    print(data["Signature"])
    # 此处会实际调用，成功后可能产生计费
    # resp = requests.get("https://" + endpoint, params=data)
    # print(resp.url)
```

接口签名v3

TCloudFinanceZone API 会对每个访问请求进行身份验证，即每个请求都需要在公共请求参数中包含签名信息（Signature）以验证请求者身份。
签名信息由安全凭证生成，安全凭证包括 SecretId 和 SecretKey；若用户还没有安全凭证，请前往云API密钥页面申请，否则无法调用云API接口。

1. 申请安全凭证

在第一次使用云API之前，请前往云API密钥页面申请安全凭证。
安全凭证包括 SecretId 和 SecretKey：

- SecretId 用于标识 API 调用者身份
- SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。
- 用户必须严格保管安全凭证，避免泄露。

申请安全凭证的具体步骤如下：

1. 登录TCloudFinanceZone管理中心控制台。
2. 前往云API密钥的控制台页面
3. 在云API密钥页面，点击【新建】即可以创建一对SecretId/SecretKey

注意：开发商帐号最多可以拥有两对 SecretId / SecretKey。

2. TC3-HMAC-SHA256 签名方法

注意：对于GET方法，只支持 Content-Type: application/x-www-form-urlencoded 协议格式。对于POST方法，目前支持 Content-Type: application/json 以及 Content-Type: multipart/form-data 两种协议格式，json 格式默认所有业务接口均支持，multipart 格式只有特定业务接口支持，此时该接口不能使用 json 格式调用，参考具体业务接口文档说明。

下面以云服务器查询广州区实例列表作为例子，分步骤介绍签名的计算过程。我们仅用到了查询实例列表的两个参数：Limit 和 Offset，使用 GET 方法调用。

假设用户的 SecretId 和 SecretKey 分别是：AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE 和 Gu5t9xGARNpq86cd98joQYCN3EXAMPLE

2.1. 拼接规范请求串

按如下格式拼接规范请求串（CanonicalRequest）：

```
CanonicalRequest =
    HTTPRequestMethod + '\n' +
    CanonicalURI + '\n' +
    CanonicalQueryString + '\n' +
    CanonicalHeaders + '\n' +
    SignedHeaders + '\n' +
    HashedRequestPayload
```

- HTTPRequestMethod：HTTP 请求方法（GET、POST ），本示例中为 GET；
- CanonicalURI：URI 参数，API 3.0 固定为正斜杠（/）；
- CanonicalQueryString：发起 HTTP 请求 URL 中的查询字符串，对于 POST 请求，固定为空字符串，对于 GET 请求，则为 URL 中问号（?）后面的字符串内容，本示例取值为：Limit=10&Offset=0。注意：CanonicalQueryString 需要经过 URL 编码。
- CanonicalHeaders：参与签名的头部信息，至少包含 host 和 content-type 两个头部，也可加入自定义的头部参与签名以提高自身请求的唯一性和安全性。拼接规则：1）头部 key 和 value 统一转成小写，并去掉首尾空格，按照 key:value\n 格式拼接；2）多个头部，按照头部 key（小写）的字典排序进行拼接。此例中为： content-type:application/x-www-form-urlencoded\nhost:cvm.finance.cloud.tencent.com\n
- SignedHeaders：参与签名的头部信息，说明此次请求有哪些头部参与了签名，和 CanonicalHeaders 包含的头部内容是一一对应的。content-type 和 host 为必选头部。拼接规则：1）头部 key 统一转成小写；2）多个头部 key（小写）按照字典排序进行拼接，并且以分号（;）分隔。此例中为： content-type;host
- HashedRequestPayload：请求正文的哈希值，计算方法为 Lowercase(HexEncode(Hash.SHA256(RequestPayload)))，对 HTTP 请求整个正文 payload 做 SHA256 哈希，然后十六进制编码，最后编码串转换成小写字母。注意：对于 GET 请求，RequestPayload 固定为空字符串，对于 POST 请求，RequestPayload 即为 HTTP 请求正文 payload。

根据以上规则，示例中得到的规范请求串如下（为了展示清晰，\n 换行符通过另起打印新的一行替代）：

```
GET
/
Limit=10&Offset=0
content-type:application/x-www-form-urlencoded
host:cvm.finance.cloud.tencent.com

content-type;host
e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```

2.2. 拼接待签名字符串

按如下格式拼接待签名字符串：

```
StringToSign =
    Algorithm + \n +
```

```
RequestTimestamp + \n +
CredentialScope + \n +
HashedCanonicalRequest
```

- Algorithm：签名算法，目前固定为 TC3-HMAC-SHA256；
- RequestTimestamp：请求时间戳，即请求头部的 X-TC-Timestamp 取值，如上示例请求为 1539084154；
- CredentialScope：凭证范围，格式为 Date/service/tc3_request，包含日期、所请求的服务和终止字符串（tc3_request）。Date 为 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service 为产品名，必须与调用的产品域名一致，例如 cvm。如上示例请求，取值为 2018-10-09/cvm/tc3_request；
- HashedCanonicalRequest：前述步骤拼接所得规范请求串的哈希值，计算方法为 Lowercase(HexEncode(Hash.SHA256(CanonicalRequest)))。

注意：

> 1. Date 必须从时间戳 X-TC-Timestamp 计算得到，且时区为 UTC+0。如果加入系统本地时区信息，例如东八区，将导致白天和晚上调用成功，但是凌晨时调用必定失败。假设时间戳为 1551113065，在东八区的时间是 2019-02-26 00:44:25，但是计算得到的 Date 取 UTC+0 的日期应为 2019-02-25，而不是 2019-02-26。
> 2. Timestamp 必须是当前系统时间，且需确保系统时间和标准时间是同步的，如果相差超过五分钟则必定失败。如果长时间不和标准时间同步，可能导致运行一段时间后，请求必定失败（返回签名过期错误）。

根据以上规则，示例中得到的待签名字符串如下（为了展示清晰，\n 换行符通过另起打印新的一行替代）：

```
TC3-HMAC-SHA256
1539084154
2018-10-09/cvm/tc3_request
91c9c192c14460df6c1ffc69e34e6c5e90708de2a6d282cccf957dbf1aa7f3a7
```

2.3. 计算签名

1）计算派生签名密钥，伪代码如下

```
SecretKey = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"
SecretDate = HMAC_SHA256("TC3" + SecretKey, Date)
SecretService = HMAC_SHA256(SecretDate, Service)
SecretSigning = HMAC_SHA256(SecretService, "tc3_request")
```

- SecretKey：原始的 SecretKey；
- Date：即 Credential 中的 Date 字段信息，如上示例，为2018-10-09；
- Service：即 Credential 中的 Service 字段信息，如上示例，为 cvm；

2）计算签名，伪代码如下

```
Signature = HexEncode(HMAC_SHA256(SecretSigning, StringToSign))
```

- SecretSigning：即以上计算得到的派生签名密钥；
- StringToSign：即步骤2计算得到的待签名字符串；

2.4. 拼接 Authorization

按如下格式拼接 Authorization：

```
Authorization =
    Algorithm + ' ' +
    'Credential=' + SecretId + '/' + CredentialScope + ', ' +
    'SignedHeaders=' + SignedHeaders + ', '
    'Signature=' + Signature
```

- Algorithm：签名方法，固定为 TC3-HMAC-SHA256；
- SecretId：密钥对中的 SecretId；
- CredentialScope：见上文，凭证范围；
- SignedHeaders：见上文，参与签名的头部信息；
- Signature：签名值

根据以上规则，示例中得到的值为：

```
TC3-HMAC-SHA256 Credential=AKIDEXAMPLE/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
```

最终完整的调用信息如下：

```
https://cvm.finance.cloud.tencent.com/?Limit=10&Offset=0

Authorization: TC3-HMAC-SHA256 Credential=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE/2018-10-09/cvm/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
Content-Type: application/x-www-form-urlencoded
Host: cvm.finance.cloud.tencent.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1539084154
X-TC-Region: shjr
```

3. 签名失败

根据实际情况，存在以下签名失败的错误码，请根据实际情况处理

错误代码	错误描述
AuthFailure.SignatureExpire	签名过期
AuthFailure.SecretIdNotFound	密钥不存在
AuthFailure.SignatureFailure	签名错误
AuthFailure.TokenFailure	token 错误
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）

4. 签名演示

Java

```
import java.io.BufferedReader;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.URL;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Map;
import java.util.TimeZone;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.net.ssl.HttpsURLConnection;
import javax.xml.bind.DatatypeConverter;

import org.apache.commons.codec.digest.DigestUtils;

public class CloudAPITC3Demo {
    private final static String CHARSET = "UTF-8";
    private final static String ENDPOINT = "cvm.finance.cloud.tencent.com";
    private final static String PATH = "/";
    private final static String SECRET_ID = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE";
    private final static String SECRET_KEY = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE";
    private final static String CT_X_WWW_FORM_URLENCODED = "application/x-www-form-urlencode
d";
    private final static String CT_JSON = "application/json";
```

```
    private final static String CT_FORM_DATA = "multipart/form-data";

    public static byte[] sign256(byte[] key, String msg) throws Exception {
        Mac mac = Mac.getInstance("HmacSHA256");
        SecretKeySpec secretKeySpec = new SecretKeySpec(key, mac.getAlgorithm());
        mac.init(secretKeySpec);
        return mac.doFinal(msg.getBytes(CHARSET));
    }

    public static void main(String[] args) throws Exception {
        String service = "cvm";
        String host = "cvm.finance.cloud.tencent.com";
        String region = "shjr";
        String action = "DescribeInstances";
        String version = "2017-03-12";
        String algorithm = "TC3-HMAC-SHA256";
        String timestamp = "1539084154";
        //String timestamp = String.valueOf(System.currentTimeMillis() / 1000);
        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
        // 注意时区，否则容易出错
        sdf.setTimeZone(TimeZone.getTimeZone("UTC"));
        String date = sdf.format(new Date(Long.valueOf(timestamp + "000")));

        // ************* 步骤 1：拼接规范请求串 *************
        String httpRequestMethod = "GET";
        String canonicalUri = "/";
        String canonicalQueryString = "Limit=10&Offset=0";
        String canonicalHeaders = "content-type:application/x-www-form-urlencoded\n" + "host:" + host
+ "\n";
        String signedHeaders = "content-type;host";
        String hashedRequestPayload = DigestUtils.sha256Hex("");
        String canonicalRequest = httpRequestMethod + "\n" + canonicalUri + "\n" + canonicalQueryStri
ng + "\n"
                + canonicalHeaders + "\n" + signedHeaders + "\n" + hashedRequestPayload;
        System.out.println(canonicalRequest);

        // ************* 步骤 2：拼接待签名字符串 *************
        String credentialScope = date + "/" + service + "/" + "tc3_request";
        String hashedCanonicalRequest = DigestUtils.sha256Hex(canonicalRequest.getBytes(CHARSET));
        String stringToSign = algorithm + "\n" + timestamp + "\n" + credentialScope + "\n" + hashedCan
onicalRequest;
        System.out.println(stringToSign);

        // ************* 步骤 3：计算签名 *************
        byte[] secretDate = sign256(("TC3" + SECRET_KEY).getBytes(CHARSET), date);
        byte[] secretService = sign256(secretDate, service);
        byte[] secretSigning = sign256(secretService, "tc3_request");
```

```
        String signature = DatatypeConverter.printHexBinary(sign256(secretSigning, stringToSign)).toLow
erCase();
        System.out.println(signature);

        // ************* 步骤 4：拼接 Authorization *************
        String authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
                + "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
        System.out.println(authorization);

        TreeMap<String, String> headers = new TreeMap<String, String>();
        headers.put("Authorization", authorization);
        headers.put("Host", host);
        headers.put("Content-Type", CT_X_WWW_FORM_URLENCODED);
        headers.put("X-TC-Action", action);
        headers.put("X-TC-Timestamp", timestamp);
        headers.put("X-TC-Version", version);
        headers.put("X-TC-Region", region);
    }
}
```

Python

```
# -*- coding: utf-8 -*-
import hashlib, hmac, json, os, sys, time
from datetime import datetime

# 密钥参数
secret_id = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"
secret_key = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"

service = "cvm"
host = "cvm.finance.cloud.tencent.com"
endpoint = "https://" + host
region = "shjr"
action = "DescribeInstances"
version = "2017-03-12"
algorithm = "TC3-HMAC-SHA256"
timestamp = 1539084154
date = datetime.utcfromtimestamp(timestamp).strftime("%Y-%m-%d")
params = {"Limit": 10, "Offset": 0}

# ************* 步骤 1：拼接规范请求串 *************
http_request_method = "GET"
canonical_uri = "/"
canonical_querystring = "Limit=10&Offset=0"
ct = "x-www-form-urlencoded"
```

```
payload = ""
if http_request_method == "POST":
    canonical_querystring = ""
    ct = "json"
    payload = json.dumps(params)
canonical_headers = "content-type:application/%s\nhost:%s\n" % (ct, host)
signed_headers = "content-type;host"
hashed_request_payload = hashlib.sha256(payload.encode("utf-8")).hexdigest()
canonical_request = (http_request_method + "\n" +
                     canonical_uri + "\n" +
                     canonical_querystring + "\n" +
                     canonical_headers + "\n" +
                     signed_headers + "\n" +
                     hashed_request_payload)
print(canonical_request)

# ************* 步骤 2：拼接待签名字符串 *************
credential_scope = date + "/" + service + "/" + "tc3_request"
hashed_canonical_request = hashlib.sha256(canonical_request.encode("utf-8")).hexdigest()
string_to_sign = (algorithm + "\n" +
                  str(timestamp) + "\n" +
                  credential_scope + "\n" +
                  hashed_canonical_request)
print(string_to_sign)

# ************* 步骤 3：计算签名 *************
# 计算签名摘要函数
def sign(key, msg):
    return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()
secret_date = sign(("TC3" + secret_key).encode("utf-8"), date)
secret_service = sign(secret_date, service)
secret_signing = sign(secret_service, "tc3_request")
signature = hmac.new(secret_signing, string_to_sign.encode("utf-8"), hashlib.sha256).hexdigest()
print(signature)

# ************* 步骤 4：拼接 Authorization *************
authorization = (algorithm + " " +
                 "Credential=" + secret_id + "/" + credential_scope + ", " +
                 "SignedHeaders=" + signed_headers + ", " +
                 "Signature=" + signature)
print(authorization)

# 公共参数添加到请求头部
headers = {
    "Authorization": authorization,
    "Host": host,
    "Content-Type": "application/%s" % ct,
```

```
    "X-TC-Action": action,
    "X-TC-Timestamp": str(timestamp),
    "X-TC-Version": version,
    "X-TC-Region": region,
}
```

请求结构

1. 服务地址

地域（Region）是指物理的数据中心的地理区域。TCloudFinanceZone交付验证不同地域之间完全隔离，保证不同地域间最大程度的稳定性和容错性。为了降低访问时延、提高下载速度，建议您选择最靠近您客户的地域。

您可以通过 API接口 查询地域列表 查看完成的地域列表。

2. 通信协议

TCloudFinanceZone API 的所有接口均通过 HTTPS 进行通信，提供高安全性的通信通道。

3. 请求方法

支持的 HTTP 请求方法:

- POST（推荐）
- GET

POST 请求支持的 Content-Type 类型：

- application/json（推荐），必须使用 TC3-HMAC-SHA256 签名方法。
- application/x-www-form-urlencoded，必须使用 HmacSHA1 或 HmacSHA256 签名方法。
- multipart/form-data（仅部分接口支持），必须使用 TC3-HMAC-SHA256 签名方法。

GET 请求的请求包大小不得超过 32 KB。POST 请求使用签名方法为 HmacSHA1、HmacSHA256 时不得超过 1 MB。POST 请求使用签名方法为 TC3-HMAC-SHA256 时支持 10 MB。

4. 字符编码

均使用UTF-8编码。

返回结果

正确返回结果

以云服务器的接口查看实例状态列表 (DescribeInstancesStatus) 2017-03-12 版本为例，若调用成功，其可能的返回如下为：

```
{
    "Response": {
        "TotalCount": 0,
        "InstanceStatusSet": [],
        "RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
    }
}
```

- Response 及其内部的 RequestId 是固定的字段，无论请求成功与否，只要 API 处理了，则必定会返回。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系我们，并提供该 ID 来解决问题。
- 除了固定的字段外，其余均为具体接口定义的字段，不同的接口所返回的字段参见接口文档中的定义。此例中的 TotalCount 和 InstanceStatusSet 均为 DescribeInstancesStatus 接口定义的字段，由于调用请求的用户暂时还没有云服务器实例，因此 TotalCount 在此情况下的返回值为 0， InstanceStatusSet 列表为空。

错误返回结果

若调用失败，其返回值示例如下为：

```
{
    "Response": {
        "Error": {
            "Code": "AuthFailure.SignatureFailure",
            "Message": "The provided credentials could not be validated. Please check your signature is cor
rect."
        },
        "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
    }
}
```

- Error 的出现代表着该请求调用失败。Error 字段连同其内部的 Code 和 Message 字段在调用失败时是必定返回的。
- Code 表示具体出错的错误码，当请求出错时可以先根据该错误码在公共错误码和当前接口对应的错误码列表里面查找对应原因和解决方案。

- Message 显示出了这个错误发生的具体原因，随着业务发展或体验优化，此文本可能会经常保持变更或更新，用户不应依赖这个返回值。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系我们，并提供该 ID 来解决问题。

公共错误码

返回结果中如果存在 Error 字段，则表示调用 API 接口失败。 Error 中的 Code 字段表示错误码，所有业务都可能出现的错误码为公共错误码，下表列出了公共错误码。

错误码	错误描述
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。
AuthFailure.MFAFailure	MFA 错误。
AuthFailure.SecretIdNotFound	密钥不存在。
AuthFailure.SignatureExpire	签名过期。
AuthFailure.SignatureFailure	签名错误。
AuthFailure.TokenFailure	token 错误。
AuthFailure.UnauthorizedOperation	请求未 CAM 授权。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。
FailedOperation	操作失败。
InternalError	内部错误。
InvalidAction	接口不存在。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
NoSuchVersion	接口版本不存在。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。

错误码	错误描述
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。
UnsupportedProtocol	http(s)请求协议错误，只支持 GET 和 POST 请求。
UnsupportedRegion	接口不支持所传地域。

公共参数

公共参数是用于标识用户和接口鉴权目的的参数，如非必要，在每个接口单独的接口文档中不再对这些参数进行说明，但每次请求均需要携带这些参数，才能正常发起请求。

签名方法 v3

使用 TC3-HMAC-SHA256 签名方法时，公共参数需要统一放到 HTTP Header 请求头部中，如下：

参数名称	类型	必选	描述
X-TC-Action	String	是	操作的接口名称。取值参考接口文档中输入参数公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。
X-TC-Region	String	是	地域参数，用来标识希望操作哪个地域的数据。接口接受的地域取值参考接口文档中输入参数公共参数 Region 的说明。注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。
X-TC-Timestamp	Integer	是	当前 UNIX 时间戳，可记录发起 API 请求的时间。例如 1529223702。注意：如果与服务器时间相差超过5分钟，会引起签名过期错误。
X-TC-Version	String	是	操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
Authorization	String	是	HTTP 标准身份认证头部字段，例如： TC3-HMAC-SHA256 Credential=AKIDEXAMPLE/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024 其中， - TC3-HMAC-SHA256：签名方法，目前固定取该值； - Credential：签名凭证，AKIDEXAMPLE 是 SecretId；Date 是 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service为产品名，必须与调用的产品域名一致，例如cvm； - SignedHeaders：参与签名计算的头部信息，content-type 和 host 为必选头部； - Signature：签名摘要。
X-TC-Token	String	否	临时证书所用的 Token ，需要结合临时密钥一起使用。临时密钥和 Token 需要到访问管理服务调用接口获取。长期密钥不需要 Token。

签名方法 v1

使用 HmacSHA1 和 HmacSHA256 签名方法时，公共参数需要统一放到请求串中，如下

参数名称	类型	必选	描述
Action	String	是	操作的接口名称。取值参考接口文档中输入参数公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。
Region	String	是	地域参数，用来标识希望操作哪个地域的数据。接口接受的地域取值参考接口文档中输入参数公共参数 Region 的说明。注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。

参数名称	类型	必选	描述
Timestamp	Integer	是	当前 UNIX 时间戳，可记录发起 API 请求的时间。例如1529223702，如果与当前时间相差过大，会引起签名过期错误。
Nonce	Integer	是	随机正整数，与 Timestamp 联合起来，用于防止重放攻击。
SecretId	String	是	在云API密钥上申请的标识身份的 SecretId，一个 SecretId 对应唯一的 SecretKey ，而 SecretKey 会用来生成请求签名 Signature。
Signature	String	是	请求签名，用来验证此次请求的合法性，需要用户根据实际的输入参数计算得出。具体计算方法参见接口鉴权文档。
Version	String	是	操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。
SignatureMethod	String	否	签名方式，目前支持 HmacSHA256 和 HmacSHA1。只有指定此参数为 HmacSHA256 时，才使用 HmacSHA256 算法验证签名，其他情况均使用 HmacSHA1 验证签名。
Token	String	否	临时证书所用的 Token ，需要结合临时密钥一起使用。临时密钥和 Token 需要到访问管理服务调用接口获取。长期密钥不需要 Token 。

地域列表

地域（Region）是指物理的数据中心的地理区域。TCloudFinanceZone交付验证不同地域之间完全隔离，保证不同地域间最大程度的稳定性和容错性。为了降低访问时延、提高下载速度，建议您选择最靠近您客户的地域。

您可以通过 API接口 查询地域列表 查看完成的地域列表。

审批服务接口
批量进行审批

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

批量进行审批

默认接口请求频率限制：20次/秒。

接口更新时间：2023-12-15 04:27:58。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：BatchPerformApproval
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
PaperIDs	是	否	Array of Uint64	待审批ID列表 示例值： 1
Operate	是	否	Uint64	枚举值，12: 驳回，14: 同意 示例值： 14
Opinion	否	否	String	审批意见 示例值： Approved

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

批量提交审批申请

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

批量提交审批申请

默认接口请求频率限制：20次/秒。

接口更新时间：2023-12-15 04:25:16。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：BatchSubmitApproval
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
Reason	是	否	String	申请理由 示例值： Reason
PaperIDs	是	否	Array of Uint64	审批单ID列表 示例值： 1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

删除审批流

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

删除审批流

默认接口请求频率限制：20次/秒。

接口更新时间：2023-12-15 04:21:07。

接口既验签名又鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：DeleteApprovalFlow
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
FlowID	是	否	Uint64	审批流id 示例值： 1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

修改审批流

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

修改审批流

默认接口请求频率限制：20次/秒。

接口更新时间：2025-04-26 21:40:23。

接口既验签名又鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：ModifyApprovalFlow
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
Name	是	否	String	审批流名称 示例值： Name
Description	是	否	String	审批流描述 示例值： Description
SchemaProps	是	否	String	可修改的action schema 示例值： SchemaProps
Activated	是	否	Bool	是否激活 示例值： true
ActionIDs	是	否	Array of Uint64	关联业务id列表 示例值： [1,2,3]
Stages	是	否	Array of StageParam	节点列表 示例值：查看

参数名称	必选	允许NULL	类型	描述
Scopes	是	否	Array of Scope	授权范围 示例值：查看
FlowID	是	否	Uint64	审批流id 示例值： 1
AllowSms	否	否	Bool	允许审批单使用手机验证码进行审批 示例值： false
AutoReject	否	否	Bool	是否自动拒绝审批 示例值： false
Remark	否	否	String	备注 示例值： remark

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见公共错误码。

错误码	描述
InternalError.Approval	内部错误

操作审批流状态

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

操作审批流状态

默认接口请求频率限制：20次/秒。

接口更新时间：2023-12-15 04:21:07。

接口既验签名又鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：OperateFlowStatus
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
FlowID	是	否	Uint64	审批流id 示例值： 1
Activated	是	否	Bool	true启用, false审批 示例值： true

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

查询审批流状态

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

查询租户关联的审批流状态

默认接口请求频率限制：20次/秒。

接口更新时间：2024-11-06 21:52:54。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：QueryApprovalFlowStatus
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
ApprovalApiInfo	是	否	ApiInfo	v3接口参数 示例值：查看

3. 输出参数

参数名称	类型	描述
Data	IsOpenAttr	data 示例值：查看
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

查询审批单详情

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

我的审批-"我的审批/我的申请"：查看审批详情接口

默认接口请求频率限制：20次/秒。

接口更新时间：2025-04-26 22:46:29。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：QueryCurrApprovalDetail
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
ID	是	否	Uint64	审批单 id 示例值： 1

3. 输出参数

参数名称	类型	描述
Data	Array of ApprovalPaperAttr	Data 示例值：查看
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

获取租户端审批单详情

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

获取租户端审批单详情

默认接口请求频率限制：20次/秒。

接口更新时间：2025-04-26 22:30:20。

接口既验签名又鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：QueryCustomerApprovalDetail
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
ID	是	否	Uint64	审批单id 示例值： 1

3. 输出参数

参数名称	类型	描述
Data	Array of ApprovalPaperAttr	Data 示例值：查看
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

查询待申请列表

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

查询待申请列表

默认接口请求频率限制：20次/秒。

接口更新时间：2025-01-24 22:43:58。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：QueryPendingApprovalDoc
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
ActionName	否	否	String	业务名称 示例值： Name
ApplicantUin	否	否	String	申请人uin 示例值： 130000000281
Applicant	否	否	String	申请人 示例值： Lucy
OwnerAccount	否	否	String	所属主账号 示例值： Tom
Reason	否	否	String	申请理由 示例值： Reason
Limit	是	否	Uint64	页数 示例值： 1

参数名称	必选	允许NULL	类型	描述
Offset	是	否	Uint64	偏移量 示例值： 10
ID	否	否	Uint64	审批单id 示例值： 1
Status	否	否	Uint64	审批单状态 示例值： 1
Sort	否	否	Order	排序字段信息 示例值：查看

3. 输出参数

参数名称	类型	描述
Data	ApprovalPaperSet	返回数据 示例值：查看
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

发送审批提醒

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

发送审批提醒

默认接口请求频率限制：20次/秒。

接口更新时间：2023-12-15 04:31:06。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：SendApprovalReminder
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
PaperID	是	否	Uint64	审批单ID 示例值： 1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

发送审批验证码

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

发送审批验证码

默认接口请求频率限制：20次/秒。

接口更新时间：2023-12-15 04:59:22。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：SendApprovalVerifyCode
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
ApproverUin	否	否	String	审批人UIN 示例值： 130000000281
PaperID	否	否	Uint64	审批单ID 示例值： 1
StageSerialNum	否	否	Uint64	当前审批单节点序号 示例值： 1

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

校验审批短信验证码

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

校验审批短信验证码

默认接口请求频率限制：20次/秒。

接口更新时间：2023-12-15 05:01:08。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：VerifyApprovalSmsCode
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
ApproverUin	否	否	String	审批人UIN 示例值： 130000000281
PaperID	否	否	Uint64	审批单ID 示例值： 1
StageSerialNum	否	否	Uint64	审批单当前节点序号 示例值： 1
Code	否	否	String	审批短信验证码 示例值： 1234

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

撤销审批申请

1. 接口描述

接口请求域名： tapproval.api3.finance.cloud.tencent.com。

租户撤销已提交的审批单， 会通知租户撤销结果， 当前审批人无需处理此审批单

默认接口请求频率限制：20次/秒。

接口更新时间：2024-04-14 21:18:23。

接口只验签名不鉴权。

2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见公共请求参数。

参数名称	必选	允许NULL	类型	描述
Action	是	否	String	公共参数，本接口取值：WithdrawApplication
Version	是	否	String	公共参数，本接口取值：2022-05-18
Region	是	否	String	公共参数，地域信息本接口不需要传递此参数。
PaperID	否	否	Uint64	审批单id 示例值： 1
Reason	否	否	String	撤回审批单理由 示例值： Reason

3. 输出参数

参数名称	类型	描述
RequestId	String	唯一请求 ID，每次请求都会返回。定位问题时需要提供该次请求的 RequestId。

4. 错误码

该接口暂无业务逻辑相关的错误码，其他错误码详见公共错误码。

数据结构

ActionAttr

审批业务

被如下接口引用：QueryActionSet

名称	必选	允许NULL	类型	描述
Action	否	否	String	审批业务接口名称 示例值： AddMachine
Module	否	否	String	审批业务模块 示例值： cvm
Version	否	否	String	审批业务版本 示例值： 2022-02-02
ActionID	否	否	Int64	审批业务 ID 示例值： 1
Description	否	否	String	审批业务描述 示例值： AddMachine
Schema	否	否	String	Schema 示例值： Schema
ActionName	否	否	String	审批业务接口名 示例值： AddMachine
ModuleName	否	否	String	审批业务模块名 示例值： cvm
YunProductName	否	否	String	审批业务云产品名称 示例值： cvm
Category	否	否	String	类型 示例值： 6
Creator	否	否	String	创建人 示例值： admin
UpdateUser	否	否	String	修改人 示例值： Lucy

名称	必选	允许NULL	类型	描述
CreateUserUin	否	否	String	创建人 Uin 示例值： 130000000281
UpdateUserUin	否	否	String	修改人 Uin 示例值： 130000000281
CreateTime	否	否	Datetime_iso	创建日期 示例值： 2024-08-03T12:00:00+08:00
UpdateTime	否	否	Datetime_iso	修改日期 示例值： 2024-08-03T12:00:00+08:00
Origin	否	否	Int64	来源 示例值： 1
Type	否	否	Int64	类型 示例值： 6
JsonPath	否	否	String	JsonPath 示例值： JsonPath

ApiInfo

yunapi v3接口 三元组参数

被如下接口引用：QueryApprovalFlowStatus、QueryCustomerApprovalDoc

名称	必选	允许NULL	类型	描述
Action	是	否	String	接口名 示例值： CreateFlow
Module	是	否	String	模块名 示例值： Module
Version	是	否	String	版本 示例值： 2024-01-01

QueryActionParma

查询action

被如下接口引用：QueryActionSet

名称	必选	允许NULL	类型	描述
ActionIDs	否	否	Array of Uint64	id 列表 示例值： [1,2,3]
Action	否	否	String	action名称 示例值： CreateFlow
Module	否	否	String	所属模块 示例值： Module
Limit	是	否	Uint64	页数 示例值： 1
Offset	是	否	Uint64	偏移量 示例值： 10
ActionName	否	否	String	业务名称 示例值： Name
ModuleName	否	否	String	接口对应模块(中文名) 示例值： approval
YunProductName	否	否	String	接口对应产品(中文名) 示例值： BMPPro
Sort	否	否	Order	排序字段 示例值：查看
Filters	否	否	Array of FilterAttr	过滤条件列表 示例值：查看

StageParam

stage参数

被如下接口引用：CreateCustomerFlow、GetApprovedSet、GetFlowDetail、ModifyApprovalFlow、QueryApplicationForm、QueryCurrApprovalDetail、QueryCurrApprovalDoc、QueryCustomerApprovalDetail、QueryCustomerApprovalDoc、QueryPendingApprovalDoc

名称	必选	允许NULL	类型	描述
StageID	否	否	Uint64	stage id 示例值： 1

名称	必选	允许NULL	类型	描述
Name	是	否	String	节点名称 示例值： Name
SingleSeal	是	否	Bool	false 会签， true或签 示例值： true
Approvers	是	是	Array of String	审批租户uin列表 示例值： [130000000281]
SerialNumber	是	否	Uint64	节点序号 示例值： 1
Seals	否	是	Array of SealParam	Seals 示例值：查看
StageStatus	否	否	Int64	StageStatus 示例值： 14
ApproverInfo	否	是	Array of ApproverInfoParam	ApproverInfo 示例值：查看

ActionRelatedInfo

审批业务关联审批流相关信息

被如下接口引用：QueryCustomerActionScope

名称	必选	允许NULL	类型	描述
Users	否	否	Array of UserScope	关联的用户信息，-1代表全体租户 示例值：查看
Flow	否	否	Array of FlowBaseAttr	关联的审批流信息 示例值：查看
ActionID	否	否	Int64	审批业务ID 示例值： 1

ApprovalPaperSet

审批单列表信息

被如下接口引用：GetApprovedSet、QueryApplicationForm、QueryCurrApprovalDoc、

QueryCustomerApprovalDoc、QueryPendingApprovalDoc

名称	必选	允许NULL	类型	描述
PaperSet	是	是	Array of ApprovalPaperAttr	审批单列表 示例值：查看
Total	是	否	Int64	总数 示例值： 100

FlowDetail

审批流详情

被如下接口引用：GetFlowDetail

名称	必选	允许NULL	类型	描述
FlowID	否	否	Int64	审批流 ID 示例值： 18
Platform	否	否	String	平台 示例值： tcloud
Name	否	否	String	审批流名称 示例值： Name
Description	否	否	String	审批流描述 示例值： Description
SchemaProps	否	否	String	SchemaProps 示例值： SchemaProps
Activated	否	否	Bool	是否启用 示例值： true
OwnerUin	否	否	String	主账号 Uin 示例值： 130000000013
CreateUin	否	否	String	创建人 Uin 示例值： 130000000013
CTime	否	否	Datetime_iso	创建日期 示例值： 2025-04-25T17:12:03+08:00
UpTime	否	否	Datetime_iso	修改日期 示例值： 2025-04-25T17:12:03+08:00

名称	必选	允许NULL	类型	描述
Version	否	否	Int64	版本 示例值： 12
CreateUser	否	否	String	创建用户 示例值： Lucy
ActionIDs	否	是	Array of Int64	审批业务 ID 示例值： [1,2,3]
AllowSms	否	是	Bool	是否允许短信审批 示例值： false
AutoReject	否	是	Bool	是否自动拒绝 示例值： false
Remark	否	是	String	备注 示例值： Remark
Stages	否	是	Array of StageParam	审批节点列表 示例值：查看
Scopes	否	是	Array of Scope	审批作用范围列表 示例值：查看

ApproverInfoParam

ApproverInfoParam

被如下接口引用：CreateCustomerFlow、GetApprovedSet、GetFlowDetail、ModifyApprovalFlow、QueryApplicationForm、QueryCurrApprovalDetail、QueryCurrApprovalDoc、QueryCustomerApprovalDetail、QueryCustomerApprovalDoc、QueryPendingApprovalDoc

名称	必选	允许NULL	类型	描述
ApproverUin	否	否	String	ApproverUin 示例值： 130000000281
ApproverUserName	否	否	String	ApproverUserName 示例值： Lucy
ApproverStatus	否	否	String	ApproverStatus 示例值： 14

FilterAttr

过滤参数

被如下接口引用：DescribeFlowSet、QueryActionSet

名称	必选	允许NULL	类型	描述
Name	是	否	String	参数名称 示例值： Name
Operator	是	否	String	操作符，支持equal, like 示例值： in
Value	是	否	String	参数值 示例值： dcdb

UserScope

作用用户范围

被如下接口引用：CreateCustomerFlow、GetFlowDetail、ModifyApprovalFlow、QueryCustomerActionScope

名称	必选	允许NULL	类型	描述
Uin	否	否	String	-1，全体租户或具体租户uin 示例值： 130000000281
IsSubAccount	否	否	Bool	是否为子账号 示例值： true
UserName	否	否	String	用户名 示例值： Lucy

FlowsData

审批流列表

被如下接口引用：DescribeFlowSet

名称	必选	允许NULL	类型	描述
Flows	否	否	Array of FlowBaseAttr	审批流列表 示例值：查看

名称	必选	允许NULL	类型	描述
Total	否	否	Int64	总数 示例值： 100

SealParam

SealParam for user approve record

被如下接口引用：CreateCustomerFlow、GetApprovedSet、GetFlowDetail、ModifyApprovalFlow、QueryApplicationForm、QueryCurrApprovalDetail、QueryCurrApprovalDoc、QueryCustomerApprovalDetail、QueryCustomerApprovalDoc、QueryPendingApprovalDoc

名称	必选	允许NULL	类型	描述
ID	是	否	Int64	ID 示例值： 1
PaperID	是	否	Int64	审批单 id 示例值： 1
StageSerialNum	是	否	Int64	审批阶段顺序 示例值： 1
OpUin	是	否	String	OpUin 示例值： 130000000281
ApproveTime	是	否	Datetime_iso	ApproveTime 示例值： 2024-08-03T12:00:00+08:00
Operate	是	否	Int64	Operate 示例值： 14
Opinion	是	否	String	Opinion 示例值： Approved
Request	否	是	String	Request 示例值： json RawRequest

Scope

作用用户范围

被如下接口引用：CreateCustomerFlow、GetFlowDetail、ModifyApprovalFlow

名称	必选	允许NULL	类型	描述
ActionID	是	否	Uint64	业务接口id 示例值： 1
Users	是	否	Array of UserScope	作用用户列表 示例值：查看

ApprovalPaperAttr

审批单属性

被如下接口引用：GetApprovedSet、QueryApplicationForm、QueryCurrApprovalDetail、QueryCurrApprovalDoc、QueryCustomerApprovalDetail、QueryCustomerApprovalDoc、QueryPendingApprovalDoc

名称	必选	允许NULL	类型	描述
Action	是	否	String	对应云api接口 示例值： CreateFlow
ActionDescription	是	否	String	审批业务描述信息 示例值： Description
ActionID	是	否	Int64	审批业务ID 示例值： 1
ActionName	是	否	String	审批业务中文名 示例值： Name
Applicant	是	否	String	申请人 示例值： Lucy
ApplicantUin	是	否	String	申请人uin 示例值： 130000000281
CTime	是	否	String	申请时间 示例值： 2024-08-03T12:00:00+08:00
CallbackStatus	是	否	Int64	审批通过后，回调状态. 0：未开始, 100：回调成功，101：回调失败 示例值： 100
CurrStageNum	是	否	Int64	当前审批节点 示例值： 1

名称	必选	允许NULL	类型	描述
FlowDescription	否	否	String	关联审批流描述信息 示例值： Description
FlowID	是	否	Int64	审批流ID 示例值： 1
ModuleName	是	否	String	审批业务对应云API模块中文名 示例值： approval
OwnerAccount	否	否	String	主账号名称 示例值： Tom
OwnerUin	是	否	String	主账号uIn 示例值： 130000000281
PaperID	是	否	Int64	审批单ID 示例值： 1
ParentID	否	否	Int64	上层审批单ID 示例值： 1
ProductName	是	否	String	云产品名称 示例值： Name
Reason	否	否	String	审批理由 示例值： Reason
Schema	否	否	String	schema 示例值： Schema
RequestBody	否	是	String	请求参数，此处无法按云API3规范 示例值： RequestBody
Seals	否	是	SealParam	已审批信息 示例值：查看
Status	否	否	Int64	审批单状态，0：审批单初始化，1：审批进行中，11:申请撤销，12:审批拒绝, 14: 审批通过，15：自动同意 示例值： 1
Stages	否	是	Array of StageParam	对应审批节点信息 示例值：查看
CallBackResult	否	否	String	CallBackResult 示例值： CallBackResult

名称	必选	允许NULL	类型	描述
FinalStatus	否	否	Uint64	最终审批单状态 示例值： 14
RelatedPapers	否	否	Array of ApprovalPaperAttr	关联的审批单信息 示例值：查看
Category	否	否	Uint64	审批单类型，1: 由页面触发，2: 合并申请单 示例值： Category
LastRemindTime	否	是	String	最后一次提醒审批人的时间 示例值： 2024-08-03T12:00:00+08:00
AllowSms	否	否	Bool	是否允许短信审批 示例值： false

FlowBaseAttr

审批流基础属性

被如下接口引用：DescribeFlowSet、QueryCustomerActionScope

名称	必选	允许NULL	类型	描述
FlowID	否	否	Int64	审批流 ID 示例值： 18
Platform	否	否	String	平台 示例值： tcloud
Name	否	否	String	审批流名称 示例值： Name
Description	否	否	String	审批流描述 示例值： Description
SchemaProps	否	否	String	SchemaProps 示例值： SchemaProps
Activated	否	否	Bool	是否启用 示例值： true
OwnerUin	否	否	String	主账号 Uin 示例值： 130000000013

名称	必选	允许NULL	类型	描述
CreateUin	否	否	String	创建人 Uin 示例值： 130000000013
CTime	否	否	Datetime_iso	创建日期 示例值： 2025-04-25T17:12:03+08:00
UpTime	否	否	Datetime_iso	修改日期 示例值： 2025-04-25T17:12:03+08:00
Version	否	否	Int64	版本 示例值： 12
CreateUser	否	否	String	创建用户 示例值： Lucy
ActionIDs	否	否	Array of Int64	审批业务 ID 示例值： [1,2,3]
AllowSms	否	否	Bool	是否允许短信审批 示例值： false
AutoReject	否	否	Bool	是否自动拒绝 示例值： false
Remark	否	否	String	备注 示例值： Remark

Order

排序字段信息

被如下接口引用：DescribeFlowSet、GetApprovedSet、QueryActionSet、QueryApplicationForm、QueryCurrApprovalDoc、QueryCustomerApprovalDoc、QueryPendingApprovalDoc

名称	必选	允许NULL	类型	描述
Field	是	否	String	排序字段英文名 示例值： ActionName
IsDesc	是	否	Bool	是否降序，true-升序，false-降序 示例值： true

IsOpenAttr

是否开启审批流

被如下接口引用：QueryApprovalFlowStatus

名称	必选	允许NULL	类型	描述
IsOpen	是	否	Bool	是否存在已开启的审批流， true：存在， false： 不存在 示例值： false

QueryActionsOutput

QueryActionsOutput

被如下接口引用：QueryActionSet

名称	必选	允许NULL	类型	描述
Total	否	否	Int64	总数 示例值： 100
Actions	否	是	Array of ActionAttr	审批业务列表 示例值：查看

错误码

功能说明

如果返回结果中存在 Error 字段，则表示调用 API 接口失败。例如：

```
{
  "Response": {
    "Error": {
      "Code": "AuthFailure.SignatureFailure",
      "Message": "The provided credentials could not be validated. Please check your signature is cor
rect."
    },
    "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
  }
}
```

Error 中的 Code 表示错误码，Message 表示该错误的具体信息。

错误码列表

公共错误码

错误码	说明
AuthFailure.InvalidSecretId	密钥非法（不是云 API 密钥类型）。
AuthFailure.MFAFailure	MFA 错误。
AuthFailure.SecretIdNotFound	密钥不存在。 请在控制台检查密钥是否已被删除或者禁用，如状态正常，请检查密钥是否填写正确，注意前后不得有空格。
AuthFailure.SignatureExpire	签名过期。Timestamp 和服务器时间相差不得超过五分钟，请检查本地时间是否和标准时间同步。
AuthFailure.SignatureFailure	签名错误。 签名计算错误，请对照调用方式中的接口鉴权文档检查签名计算过程。
AuthFailure.TokenFailure	token 错误。
AuthFailure.UnauthorizedOperation	请求未 CAM 授权。
DryRunOperation	DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。

错误码	说明
FailedOperation	操作失败。
InternalError	内部错误。
InvalidAction	接口不存在。
InvalidParameter	参数错误。
InvalidParameterValue	参数取值错误。
LimitExceeded	超过配额限制。
MissingParameter	缺少参数错误。
NoSuchVersion	接口版本不存在。
RequestLimitExceeded	请求的次数超过了频率限制。
ResourceInUse	资源被占用。
ResourceInsufficient	资源不足。
ResourceNotFound	资源不存在。
ResourceUnavailable	资源不可用。
UnauthorizedOperation	未授权操作。
UnknownParameter	未知参数错误。
UnsupportedOperation	操作不支持。
UnsupportedProtocol	http(s)请求协议错误，只支持 GET 和 POST 请求。
UnsupportedRegion	接口不支持所传地域。

业务错误码

错误码	说明
InternalError.Approval	内部错误
InternalError.DatabaseError	数据库错误