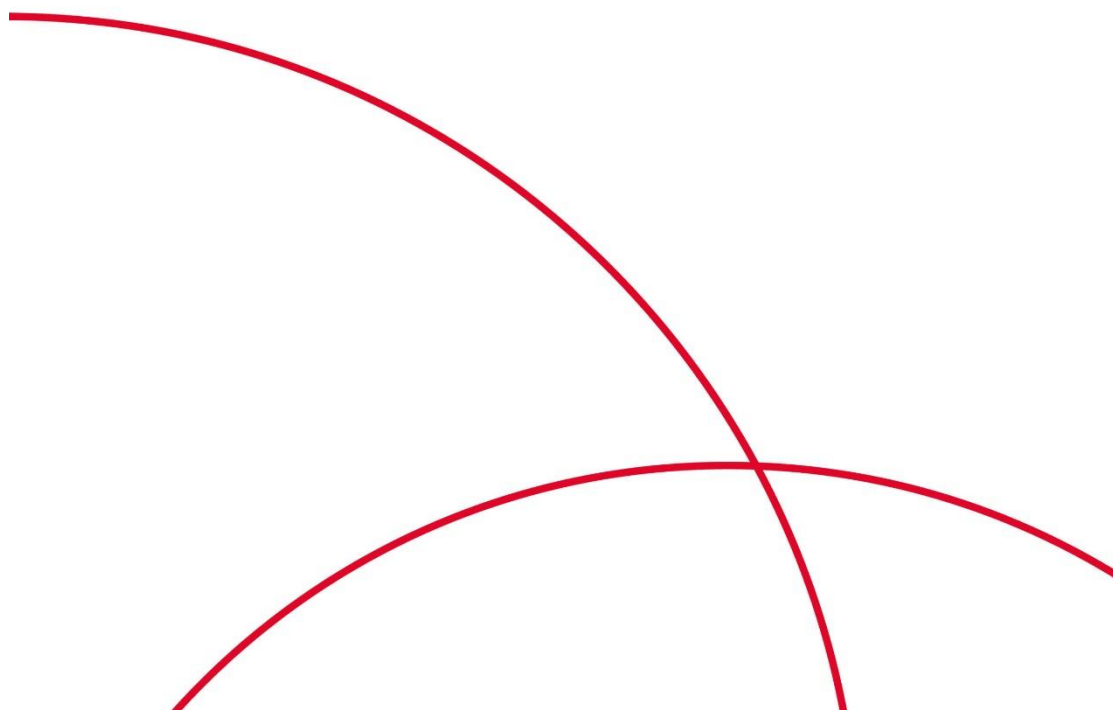




云通信短信服务

用户使用指南

天翼云科技有限公司



1.API 概览	8
1.1 短信发送接口	8
1.2 短信查询接口	8
1.3 签名接口	8
1.4 回执消息	9
1.5 模板接口	9
2.公共参数	9
2.1 总体说明	9
2.1.1 短信发送流程	10
2.1.2 服务地址	10
2.1.3 公共请求头和公共响应头	10
2.2 公共请求参数	10
2.2.1 相关代码	11
2.2.2 请求参数	11
2.3 公共响应参数	12
2.3.1 返回示例	12
2.3.2 返回参数	12
2.4 请求签名	12
步骤一：获取 AccessKey 和 SecurityKey	12

步骤二：构造时间戳.....	13
步骤三：构造请求流水号.....	13
步骤四：构造待签名字符串.....	13
步骤五：构造签名.....	14
步骤六：构造请求头.....	14
JAVA 示例.....	14
Golang 示例	19
nodejs 示例.....	21
3.回调	23
3.1 回调消息简介与配置流程	23
3.1.1 配置回调消息接收 URL	23
3.1.2 回调消息类型.....	23
3.2 回调.....	24
3.2.1 协议说明.....	24
3.2.2 请求说明.....	25
3.2.3 请求样例.....	25
3.2.4 字段说明	25
3.2.5 响应说明.....	26
3.3 事件回调	26
3.3.1 协议说明.....	26

3.3.2 请求说明.....	27
3.3.3 响应说明.....	28
3.4 状态回调	28
3.4.1 协议说明.....	29
3.4.2 请求说明.....	29
3.4.3 请求样例.....	29
3.4.4 字段说明.....	29
3.5 响应说明.....	30
4.模板	31
4.1 查询模板列表	31
4.1.1 请求参数.....	31
4.1.2 返回参数.....	32
4.1.3 示例.....	33
4.1.4 错误码.....	34
4.2 查询模板	34
4.2.1 请求参数.....	35
4.2.2 返回参数.....	35
4.2.3 示例.....	37
4.2.4 错误码.....	38
4.3 添加模板	38

4.3.1 请求参数.....	38
4.3.2 返回数据.....	39
4.3.3 示例.....	40
4.3.4 错误码.....	41
4.4 删除模板	41
4.4.1 请求参数.....	41
4.4.2 返回参数.....	42
4.4.3 示例.....	42
4.4.4 错误码.....	43
4.5 修改模板	43
4.5.1 请求参数.....	43
4.5.2 返回参数.....	45
4.5.3 示例.....	46
4.5.4 错误码.....	46
5.签名	46
5.1 查询签名列表	46
5.1.1 请求参数.....	47
5.1.2 返回参数.....	47
5.1.3 示例.....	49
5.1.4 错误码.....	50

5.2 添加签名	50
5.2.1 请求参数.....	51
5.2.2 返回数据.....	55
5.2.3 示例.....	55
5.2.4 错误码.....	56
5.3 删除签名	56
5.3.1 请求参数.....	56
5.3.2 返回数据.....	57
5.3.3 示例.....	57
5.3.4 错误码.....	58
5.4 修改签名	58
5.4.1 请求参数.....	59
5.4.2 返回数据.....	65
5.4.3 示例.....	66
5.4.4 错误码.....	66
5.5 查询签名	66
5.5.1 请求参数.....	67
5.5.2 返回参数.....	67
5.5.3 示例.....	68
5.5.4 错误码.....	69

6.发送短信	69
6.1 请求参数.....	69
6.2 返回参数.....	71
6.3 示例.....	72
7.查询发送记录	72
7.1 请求参数.....	73
7.2 返回数据.....	74
7.3 示例.....	75
7.4 错误码.....	76

1.API 概览

1.1 短信发送接口

API	描述
sendSms	发送短信。

1.2 短信查询接口

API	描述
querySendDetails	查询短信发送的状态。

1.3 签名接口

API	描述
addSmsSign	调用短信 AddSmsSign 申请短信签名。
deleteSmsSign	调用接口 DeleteSmsSign 删除短信签名。
querySmsSign	调用接口 QuerySmsSign 查询短信签名申请状态。
modifySmsSign	调用接口 ModifySmsSign 修改未审核通过的短信签名，并重新提交审核。

1.4 回执消息

API	描述
<u>smsReport</u>	订阅 SmsReport 短信状态报告，获取短信发送状态。
<u>smsMo</u>	订阅 SmsMo 上行短信消息，获取终端用户回复短信的内容。
<u>eventReport</u>	订阅签名和模板审核状态消息。

1.5 模板接口

API	描述
<u>modifySmsTemplate</u>	调用接口 ModifySmsTemplate 修改未通过审核的短信模板。
<u>querySmsTemplate</u>	调用接口 QuerySmsTemplate 查询短信模板的审核状态。
<u>addSmsTemplate</u>	调用接口 AddSmsTemplate 申请短信模板。
<u>deleteSmsTemplate</u>	调用接口 DeleteSmsTemplate 删除短信模板。

2.公共参数

2.1 总体说明

短信服务支持基于 Restful 的 POST 方式发送 HTTP/HTTPS 请求。请求参数需要包含在 Body 中，请求及返回结果都使用 UTF8 编码。

2.1.1 短信发送流程

1. 在控制台添加签名、模板并经审核通过。
2. 调用短信服务的短信发送接口。
3. 短信服务成功收到请求后转发请求给运营商，运营商发送短信。
4. 用户收到短信后，短信服务会有最终的状态消息确认，即消息回执。

对应的协议是：

1. 支持 HTTP 或 HTTPS 协议请求通信。为了获得更高的安全性，推荐您使用 HTTPS 协议发送请求。
2. 发送 API 采用 Rest 协议，其中签名算法使用了天翼云的 EOP 协议。
3. 发送后的消息回执采用 Restful 的方式异步通知给客户。

2.1.2 服务地址

调用 API 的服务地址为 <https://sms-global.ctapi.ctyun.cn/sms/api/v1>

2.1.3 公共请求头和公共响应头

API 接口中使用了公共请求头（Common Request Headers）和公共响应头（Common Response Headers），这些内容可以被所有的短信服务请求使用。

详细说明请参考[公共请求参数](#)和[公共响应参数](#)。

2.2 公共请求参数

API 接口中使用了公共请求头（Http Request Headers），这些公共请求头可以被所有的短

信服务请求使用。

2.2.1 相关代码

```
{
    'content-Type': 'application/json',
    'eop-date': '20220330T112800Z',
    'eop-Authorization': 'cb8ea12351a24d1d8cec37f9d9c568f6 Headers=ctyun-
eop-request-id;eop-
date Signature=iu4+bv/gqckuBIgrIecC7aMQXrc+ldx1wbL0Ml3U3ps=,
    'ctyun-eop-request-id': '87cee26a-a8b6-4e26-816c-8530182e2312',
}
```

复制

2.2.2 请求参数

名称	类型	是否必填	说明
content-Type	String	是	请求数据类型
eop-date	String	是	时间戳，格式为‘YYYYMMDDTHH:mm:ssZ’
eop-Authorization	String	是	加密参数，包括三部分：1、AK;2、参与加密的请求头 Headers，默认为 ctyun-eop-request-id，eop-date，且为必须添加的参数;3、加密结果 Signature 以空格分开
ctyun-eop-request-id	String	是	请求流水号，建议采用 UUID 生成

2.3 公共响应参数

API 接口中使用了公共响应头 (Http Response Headers)，这些公共响应头可以被所有的短信服务请求使用。

2.3.1 返回示例

除业务参数之外，短信服务还会返回以下公共参数：

```
{
  'ctyun-eop-request-id': '87cee26a-a8b6-4e26-816c-8530182e2312',
}
```

复制

2.3.2 返回参数

名称	类型	描述
ctyun-eop-request-id	String	请求流水号。无论调用接口成功与否，都会返回，与请求流水号对应。

2.4 请求签名

对于每一次 HTTP 或 HTTPS 协议请求，我们会根据访问中的签名信息验证访问请求者身份。具体由使用 AccessKey 和 SecurityKey 加密验证实现。其中 AccessKey 是访问者身份，SecurityKey 是加密签名字符串和服务器端验证签名字符串的密钥，必须严格保密谨防泄露。

步骤一：获取 AccessKey 和 SecurityKey

1. 登录[天翼云官网](#)。
2. 前往[个人中心](#)。
3. 在安全设置中查看**用户 AccessKey** 和 **SecurityKey**。

步骤二：构造时间戳

1. 构造一个 eop-date 的时间戳，格式为 `yyyymmddTHHMMSSZ`，简单来说就是“年月日 T 时分秒 Z”。

步骤三：构造请求流水号

1. 构造一个 `ctyun-eop-request-id` 的流水号，最好为每次请求不同，可以简单的使用 UUID。

步骤四：构造待签名字符串

1. 构造进行签名的 Header：以 `header_name:header_value` 来一个一个通过 `"\n"` 拼接起来，强制要求 `ctyun-eop-request-id` 和 `eop-date` 这个头作为 Header 中的一部分。将待签名算法的 Header 需要进行排序（`header_name` 以英文字母的顺序来排序），将排序后得到的列表进行遍历组装成待签名的 header。
2. 构造待签名的 Query: query 以 `&` 作为拼接，key 和 value 以 `"="` 连接，排序规则使用 26 个英文字母的顺序来排序，Query 参数全部都需要进行签名。
3. 构造待签名的 Body: 待发送的 body 参数进行 sha256 摘要，对摘要出来的结果转十六进制。
4. 将待签名的 Header、Query、Body 通过 `"\n"` 进行连接。

步骤五：构造签名

1. 先将 SecurityKey 作为密钥，eop-date 作为数据，根据 hmacsha256 加密算法算出 ktime。
2. 将 ktime 作为密钥，AccessKey 作为数据，根据 hmacsha256 加密算法算出 kAk。
3. 将 kAk 作为密钥，eop-date 的年月日值（前 8 位）作为数据，根据 hmacsha256 加密算法算出 kdate。
4. 将 kdate 作为密钥，步骤二的待签名字符串作为数据，根据 hmacsha256 加密算法签名并转化为 BASE64 编码算出 signature。

步骤六：构造请求头

1. 将 eop-date 作为 Key，步骤二的结果作为 Value 加入 http 请求头中。
2. 将 ctyun-eop-request-id 作为 Key，步骤三的结果作为 Value 加入 http 请求头中。
3. 将 Eop-Authorization 作为 Key，通过字符串拼接的方式将 AK、Header、Signature 通过空格进行拼接，并将结果作为 Value 加入 http 请求头中。

JAVA 示例

完整的 Java 签名 Demo 代码：

```
package com.example;  
  
/**
```

```
* Hello world!
*
*/
import java.net.URL;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.text.SimpleDateFormat;
import java.util.Arrays;
import java.util.Base64;
import java.util.Date;
import java.util.HashMap;
import java.util.Locale;
import java.util.Map;
import java.util.UUID;

import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.ContentType;
import org.apache.http.entity.StringEntity;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.util.EntityUtils;

import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import com.alibaba.fastjson.*;

public class App
{
    public static void main( String[] args ) throws Exception
    {
        Map<String, Object> params = new HashMap<>();

        params.put("signName", ""); //签名
        params.put("phoneNumber", ""); //手机号
        params.put("action", "SendSms");
    }
}
```

```

    //--模板 1

    params.put("templateCode", ""); //模板编码

    params.put("templateParam", "{\"ctyunName\":\"ABC\"}"); //根据模板中的变量填写,内容是 JSON 字符串


    params.put("extendCode", ""); //选填
    String body = JSONObject.toJSONString(params);


    // SETUP1:获取 AccessKey 和 SecurityKey

    String accessKey = ""; //填写控制台->个人中心->用户 AccessKey->查看->AccessKey

    String securityKey = ""; //填写控制台->个人中心->用户 AccessKey->查看->SecurityKey


    // SETUP2:构造时间戳

    SimpleDateFormat TIME_FORMATTER = new
SimpleDateFormat("yyyyMMdd'T'HHmmss'Z'");

    SimpleDateFormat DATE_FORMATTER = new SimpleDateFormat("yyyyMMdd");

    Date nowdate = new Date();

    String singerDate = TIME_FORMATTER.format(nowdate);

    String singerDd = DATE_FORMATTER.format(nowdate);

    System.out.println("singerDate:"+singerDate);

    System.out.println("singerDd:"+singerDd);


    // SETUP3:构造请求流水号

    String uuId = UUID.randomUUID().toString();

    System.out.println("uuId:"+uuId);


    // SETUP4:构造待签名字符串

    String CampmocalHeader = String.format("ctyun-eop-request-id:%s\nneop-
date:%s\n", uuId, singerDate); //uuid 去掉 this

    //header 的 key 按照 26 字母进行排序, 以&作为连接符连起来


    URL url = new URL("https://sms-global.ctapi.ctyun.cn/sms/api/v1");

    String query = url.getQuery();

    String afterQuery = "";

    if (query != null) {

        String param[] = query.split("&");

```

```

        Arrays.sort(param);
        for (String str : param) {
            if (afterQuery.length() < 1)
                afterQuery = afterQuery + str;
            else
                afterQuery = afterQuery + "&" + str;
        }
    }

    // String body = "";
    String calculateContentHash = getSHA256(body); //报文原封不动进行 sha256 摘要
    String sigtureStr = CampmocalHeader + "\n" + afterQuery + "\n" +
        calculateContentHash;
    System.out.println("calculateContentHash: " + calculateContentHash);
    System.out.println("sigtureStr: " + sigtureStr);

    // SETUP5:构造签名

    byte[] ktime = HmacSHA256(singerDate.getBytes(),
        securityKey.getBytes());
    byte[] kAk = HmacSHA256(accessKey.getBytes(), ktime);
    byte[] kdate = HmacSHA256(singerDd.getBytes(), kAk);
    String Signature =
        Base64.getEncoder().encodeToString(HmacSHA256(sigtureStr.getBytes("UTF-8"),
            kdate));

    // SETUP6:构造请求头
    HttpPost httpPost = new HttpPost(String.valueOf(url));
    httpPost.setHeader("Content-Type", "application/json;charset=UTF-8");
    httpPost.setHeader("ctyun-eop-request-id", uuId);
    httpPost.setHeader("Eop-date", singerDate);
    String signHeader = String.format("%s Headers=ctyun-eop-request-id;eop-
        date Signature=%s", accessKey, Signature);
    httpPost.setHeader("Eop-Authorization", signHeader);
    System.out.println("Signature" + Signature);
    System.out.println("signHeader" + signHeader);

    httpPost.setEntity(new StringEntity(body,

```

```

ContentType.create("application/json", "utf-8")));

    try (CloseableHttpClient httpClient = HttpClients.createDefault();

        CloseableHttpResponse response = httpClient.execute(httpPost) ) {

        String string = EntityUtils.toString(response.getEntity(), "utf-
8");

        System.out.println("返回结果: " + string);
    } catch (Exception e) {
        System.out.println(e.getMessage());
    }
}

private static String toHex(byte[] data) {
    StringBuilder sb = new StringBuilder(data.length * 2);
    byte[] var2 = data;
    int var3 = data.length;
    for (int var4 = 0; var4 < var3; ++var4) {
        byte b = var2[var4];
        String hex = Integer.toHexString(b);
        if (hex.length() == 1) {
            sb.append("0");
        } else if (hex.length() == 8) {
            hex = hex.substring(6);
        }
        sb.append(hex);
    }
    return sb.toString().toLowerCase(Locale.getDefault());
}

private static String getSHA256(String text) {
    try {
        MessageDigest md = MessageDigest.getInstance("SHA-256");
        // byte[] a = text.getBytes(StandardCharsets.UTF_8);

        md.update(text.getBytes(StandardCharsets.UTF_8));
        return toHex(md.digest());
    } catch (NoSuchAlgorithmException var3) {

```

```

        return null;
    }
}

private static byte[] HmacSHA256(byte[] data, byte[] key) throws
Exception {
    try {
        Mac mac = Mac.getInstance("HmacSHA256");
        mac.init(new SecretKeySpec(key, "HmacSHA256"));
        return mac.doFinal(data);
    } catch (Exception e) {
        return null;
    }
}
}

```

复制

Golang 示例

完整的 golang 签名 Demo 代码:

```

func EopPost(url string, body []byte) ([]byte, error) {
    client := fasthttp.Client{}
    req := fasthttp.AcquireRequest()
    resp := fasthttp.AcquireResponse()
    defer func() {
        fasthttp.ReleaseResponse(resp)
        fasthttp.ReleaseRequest(req)
    }()
    req.Header.SetContentType("application/json")
    req.Header.SetMethod("POST")

    // SETUP1:获取 AccessKey 和 SecurityKey
    accessKey := "CTYUN-AK"
    securityKey := "CTYUN-SK"

```

```

// SETUP2:构造时间戳
eop_date := time.Now().Format("20060102T150405Z")


// SETUP3:构造请求流水号
uuid := uuid.New()


// SETUP4:构造待签名字符串
req.SetRequestURI(url)
args := req.URI().QueryArgs()
args.Sort(bytes.Compare)
headerStr := "ctyun-eop-request-id:" + uuid + "\n" + "eop-
date:" + eop_date + "\n"
queryStr := args.String()
calculateContentHash := getSha256(body)
signStr := headerStr + "\n" + queryStr + "\n" + calculateContentHash


// SETUP5:构造签名
Ktime := hmacSha256(eop_date, securityKey)
Kak := hmacSha256(accessKey, string(Ktime))
Kdata := hmacSha256(eop_date[:8], string(Kak))
signatureDate := hmacSha256(signStr, string(Kdata))
signature := base64.StdEncoding.EncodeToString(signatureDate)


// SETUP6:构造请求头
req.Header.Add("Eop-date", eop_date)
req.Header.Add("ctyun-eop-request-id", uuid)
signatureHeader := AK + " Headers=ctyun-eop-request-id;eop-
date Signature=" + signature
req.Header.Add("Eop-Authorization", signatureHeader)


req.SetBody(body)


if err := client.Do(req, resp); err != nil {
    return nil, err
}

b := resp.Body()

```

```

        return b, nil
    }

func hmacSha256(data, secret string) []byte {
    h := hmac.New(sha256.New, String2Byte(secret))
    h.Write([]byte(data))

    return h.Sum(nil)
}

func getSha256(data []byte) string {
    hash := sha256.New()
    hash.Write(data)
    bytes := hash.Sum(nil)
    return hex.EncodeToString(bytes)
}

```

复制

nodejs 示例

完整的 nodejs 签名 Demo 代码:

```

import axios from 'axios'
import dayjs from 'dayjs'
import * as crypto from 'crypto'
import * as querystring from 'querystring'

// sha256 算法
const sha256 = data =>
    crypto.createHash('SHA256').update(data).digest('hex')

// hmacsha256 算法
const hmacsha256 = (data, key) => crypto.createHmac('sha256',
    key).update(data).digest('hex')

// SETUP1: 获取 AccessKey 和 SecurityKey
const accessKey = '65038a8f227242ab987efef3acea40dd'
const securityKey = 'f3f220c94cb242f2b51065a92485ca3f'

```

```
const body = { foo: 'bar' }

// SETUP2:构造时间戳
const timestamp = dayjs().format('YYYYMMDDTHH:mm:ss') + 'Z'

// SETUP3:构造请求流水号
const requestId = crypto.randomUUID()

// SETUP4:构造待签名字符串
const headerStr = `ctyun-eop-request-id:${requestId}\neop-
date:${timestamp}\n\n`
const calculateContentHash = sha256(JSON.stringify(body))
const rawString = `${headerStr}\n${calculateContentHash}`

// SETUP5:构造签名
const signTime = hmacsha256(timestamp, securityKey)
const signAK = hmacsha256(accessKey, Buffer.from(signTime, 'hex'))
const signDate = hmacsha256(timestamp.slice(0, 8), Buffer.from(signAK,
'hex'))
const sign = hmacsha256(rawString, Buffer.from(signDate, 'hex'))
const signature = Buffer.from(sign, 'hex').toString('base64')

// SETUP:6 构造请求头
const signatureHeader = `${accessKey} Headers=ctyun-eop-request-id;eop-date
Signature=${signature}`
const headers = {
  'Content-Type': 'application/json',
  'eop-date': timestamp,
  'Eop-Authorization': signatureHeader,
  'ctyun-eop-request-id': requestId,
}

// SETUP:7 构造请求
axios
  .post('https://sms-global.ctapi.ctyun.cn/sms/api/v1', body, {
    headers,
  })
  .then(res => {
```

```
console.log(res.data)
}))
.catch(err => {
  console.log(err)
}))
```

3.回调

回调消息是在短信发送之后，HTTP 批量推送模式，接收短信发送状态的结果和用户回复的短信内容，帮助您掌握短信的发送成功率和用户的反馈信息。

3.1 回调消息简介与配置流程

3.1.1 配置回调消息接收 URL

如果需要接收回执消息，必须先是在控制台上开启消息接收。

1. 登录[云通信控制台](#)。
2. 在左侧导航栏中选择 **消息设置**。
3. 在事件回调配置中分别配置 **状态报告通知 URL**、**事件回调 URL**、**用户回复 URL**。

3.1.2 回调消息类型

名称	描述	HTTP 批量推送模式
smsMo (上行短信消息)	上行短信指用户发送给通信服务提供商的短信，用于定制某种服务、完成某种查询、或是办理某种业务等。通过订阅 SmsMo 上行短信消息，可以获知	smsMo

名称	描述	HTTP 批量推送模式
	终端用户回复短信的内容。	
smsReport (短信状态报告)	与上行短信相对应的是下行短信。下行是指用户收到的短信，例如运营商发送的消息通知、业务提醒等短信。通过订阅 SmsReport 短信下行状态报告，可以获知每条短信的发送情况，了解短信是否达到终端用户的状态与相关信息。	smsReport
eventReport (事件回调消息)	签名审核状态消息是指通过 API 接口 AddSmsSign 申请短信签名后，可以通过 SignSmsReport 接口获取签名的审核状态消息。	eventReport

通过 HTTP 批量推送方式可以拉取上行短信消息 (SmsMo) 。

3.2 回调

3.2.1 协议说明

参数	说明
协议	HTTP+JSON
编码	UTF-8

3.2.2 请求说明

请求内容为 JSON Array 格式，单次请求可能会包含多个上行短信内容。

3.2.3 请求样例

```
{
  "phoneNumber" : "18612345678",
  "sendTime" : "2017-09-01 00:00:00",
  "content" : "内容",
  "signName" : "签名",
  "code" : "1234",
  "transId" : 1234567890
}
```

复制

3.2.4 字段说明

名称	类型	描述	示例值	是否必须
phone	String	手机号码	18900000001	必须
sendTime	String	发送时间	2022-05-01 00:00:00	必须
content	String	发送内容	这是一条上行短信	必须
signName	String	签名信息	签名	必须
destCode	String	上行短信扩展号码	1234	必须
sequenceId	Number	序列号	1234567890	必须

3.2.5 响应说明

响应样例

```
{
  "code" : 0,
  "message" : "接收成功"
}
```

复制

字段说明

名称	类型	是否必须	说明	示例值
code	Number	必须	应答编码	0
msg	String	可选	描述信息	接收成功

说明： httpStatus 必须是 200。

3.3 事件回调

通过 HTTP 批量推送方式可以拉取签名审核状态消息（EventReport）。

3.3.1 协议说明

参数	说明
协议	HTTP+JSON
编码	UTF-8

3.3.2 请求说明

请求内容为 JSON Array 格式，单次请求仅包含一条审核状态报告。

请求样例

```
{
  "type": "sign",
  "status": 1,
  "source": "天翼云",
  "reason": "通过",
  "createTime": "2019-05-30 19:58:25"
}
```

复制

字段说明

名称	类型	是否必选	示例	描述
type	String	必选	sign	sign：签名。 template：模板。
reason	String	必选	文件不能证明信息真实性，请重新上传	审核未通过原因。
status	String	必选	审核状态	1：审核通过。 2：审核不通过。
source	String	必选	天翼云	签名名称或者模板编号

名称	类型	是否必选	示例	描述
createTime	String	可选	2022-03-30 19:58:25	创建事件

3.3.3 响应说明

响应样例

```
{
  "code" : 0,
  "msg" : "接收成功"
}
```

复制

字段说明

名称	类型	是否必选	说明	示例值
code	Number	必选	应答编码	0
msg	String	可选	描述信息	接收成功

说明： httpStatus 必须是 200。

3.4 状态回调

您可以指定 HTTP URL 来接收由 SMS 推送的短信发送状态报告。然后 SmsReport 使用 HTTP 服务接收消息传递报告。

3.4.1 协议说明

参数	说明
协议	HTTP+JSON
编码	UTF-8

3.4.2 请求说明

请求内容为 JSON Array 格式，单次请求可能会包含多个状态报告结果。

3.4.3 请求样例

```
{
  "data": [
    {
      "phone": "13900000001",
      "sendTime": "2022-08-01 00:00:00",
      "reportTime": "2022-08-01 00:00:00",
      "reportCode": "DELIVERED",
      "requestId": "123456"
    }
  ]
}
```

复制

3.4.4 字段说明

名称	类型	描述	示例值	是否必须
phone	String	手机号码	18900000001	必须
sendTime	String	发送时间	2022-05-01 00:00:00	必须

名称	类型	描述	示例值	是否必须
reportTime	String	状态报告时间	2022-05-01 00:00:00	必须
reportCode	String	状态码	DELIVERED	必须
requestId	String	发送序列号	67890	必须

3.5 响应说明

响应样例

```
{
  "code" : 0,
  "msg" : "接收成功"
}
```

复制

字段说明

名称	类型	描述	示例值	是否必须
code	Number	应答编码	0	必须
msg	String	描述信息	接收成功	可选

说明： httpStatus 必须是 200。

申请短信模板后，可以通过接口 QuerySmsTemplate 查询短信模板的审核状态。如果审核未通过，会返回审核失败的原因，请针对具体原因重新修改短信模板。

4.模板

4.1 查询模板列表

申请短信模板后，可以通过接口 QuerySmsTemplateList 查询短信模板的审核状态。如果审核未通过，会返回审核失败的原因，请针对具体原因重新修改短信模板。请求参数名称类型是否必选示例值描述。

4.1.1 请求参数

名称	类型	是否必选	示例值	描述
templateCode	String	是	SMS73419576145	短信模板 CODE。您可以在控制台 模板管理 页面或 API 接口 AddSmsTemplate 的返回参数中获取短信模板 CODE。
action	String	是	QuerySmsTemplate	系统规定参数。取值： QuerySmsTemplate 。

4.1.2 返回参数

名称	类型	示例值	描述
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
createTime	String	2019-06-04 11:42:17	短信模板的创建日期和时间。
message	String	OK	状态码的描述。
reason	String	非验证码类型短信, 请选择短信通知类型为推广短信。	审核备注。 如果审核状态为 审核通过 或 审核中 , 参数 Reason 显示为“无审核备注”。 如果审核状态为 审核未通过 , 参数 Reason 显示审核的具体原因。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。

名称	类型	示例值	描述
templateCode	String	SMS73419576145	短信模板 CODE。
templateContent	String	亲爱的会员! 天翼云短信服务 祝您新年快乐!	模板内容。
templateName	String	天翼云短信测试模板	模板名称。
templateStatus	Integer	1	模板审核状态。其中： 0：审核中。 1：审核通过。 2：审核失败，请在返回参数 Reason 中查看审核失败原因。
templateType	Integer	1	短信类型。其中： 1：验证码。 2：短信通知。 3：推广短信。

4.1.3 示例

请求示例

```
{
  "action": "QuerySmsTemplate",
  "templateCode": "SMS13889962005"
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "createTime": "2022-02-11T02:03:32.711Z",
  "message": "success",
  "reason": "",
  "requestId": "ZuM53ZgNrhGDgz0W",
  "templateCode": "SMS13889962005",
  "templateContent": "您的验证码为${code},有效期为${time}分钟!",
  "templateName": "登陆验证码",
  "templateStatus": 0,
  "templateType": 0
}
```

复制

4.1.4 错误码

[查看本产品错误码](#)

4.2 查询模板

申请短信模板后，可以通过接口 QuerySmsTemplateList 查询短信模板的审核状态。如果审核未通过，会返回审核失败的原因，请针对具体原因重新修改短信模板。

4.2.1 请求参数

名称	类型	是否必选	示例值	描述
action	String	是	QuerySmsTemplateList	系统规定参数。取值： QuerySmsTemplateList 。
pageIndex	Int	是	1	页码，即获取第几页的模板
pageSize	Int	是	10	个数，即每页包含多少条模板

4.2.2 返回参数

名称	类型	示例值	描述
code	String	OK	请求状态码。返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
message	String	OK	状态码的描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。

名称	类型	示例值	描述
data	Array		短信模板 CODE。
data.templateContent	String	亲爱的会员! 天翼云短信服务祝您新年快乐!	模板内容。
data.templateName	String	天翼云短信测试模板	模板名称。
data.templateStatus	Int	1	模板审核状态。 0: 审核中。 1: 审核通过。 2: 审核失败, 请在返回参数 Reason 中查看审核失败原因。
data.templateType	Int	1	短信类型。 1: 验证码; 2: 短信通知; 3: 推广短信
data.status	Int	0	审核状态 。 0: 未审核; 1: 审核通过; 2: 审核未通过

名称	类型	示例值	描述
data.remark	String	测试	模板备注
data.reason	String	审核意见	通过!

4.2.3 示例

请求示例

```
{
  "action": "QuerySmsTemplate",
  "pageIndex": 1,
  "pageSize": 10,
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success",
  "requestId": "cgudhib511imu8sci3b0",
  "totalCount": 1,
  "data": [
    {
      "reason": "通过",
      "createTime": "2022-12-01T14:26:06.466Z",
      "remark": "测试",
      "status": 1,
      "templateCode": "SMS71288409782",
      "templateType": 2,
      "templateName": "天翼云登录",
      "templateContent": "您的验证码${code}，该验证码 5 分钟内有效，请勿泄漏于他人!"
    }
  ]
}
```

```
}
```

复制

4.2.4 错误码

[查看本产品错误码](#)

4.3 添加模板

您可以通过短信服务 API 接口或云通信控制台申请短信模板，模板内容需要符合[模板规范](#)。

短信模板审核流程请参考[模板审核](#)流程。

说明： 一个自然日最多可以申请 100 个模板，间隔建议您控制在 30S 以上。

4.3.1 请求参数

名称	类型	是否必选	示例值	描述
action	String	是	AddSmsTemplate	系统规定参数。取值： AddSmsTemplate 。
templateType	Integer	是	1	短信类型。其中：1：验证码。 2：短信通知。 3：推广短信。

名称	类型	是否必选	示例值	描述
templateName	String	是	天翼云注册	模板名称，长度为 1~30 个字符。
templateContent	String	是	您正在进行天翼云门户注册，验证码为： \${code}，5 分钟内有效！	模板内容，长度为 1~500 个字符。模板内容需要符合 模板规范 。
remark	String	是	天翼云注册	短信模板申请说明。请在申请说明中描述您的业务使用场景，长度为 1~100 个字符。

4.3.2 返回数据

名称	类型	示例值	描述
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。

名称	类型	示例值	描述
message	String	success	状态码的描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
templateCode	String	SMS9090121230500	短信模板 CODE。您可以使用模板 CODE 通过 API 接口 QuerySmsTemplate 或在控制台查看模板申请状态和结果。

4.3.3 示例

请求示例

```
{
  "action": "AddSmsTemplate",
  "templateType": 0,
  "templateContent": "您正在进行天翼云门户注册，验证码为：${code}，5 分钟内有效！",
  "templateName": "天翼云注册",
  "remark": "天翼云注册"
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success",
  "requestId": "lcIEDgU9RXydyk3Q",
  "templateCode": "SMS67753530327"
}
```

复制

4.3.4 错误码

[查看本产品错误码](#)

4.4 删除模板

注意：

- 不支持删除正在审核中的模板。
- 删除短信模板后不可恢复，请谨慎操作。

4.4.1 请求参数

名称	类型	是否必选	示例值	描述
templateCode	String	是	SMS73419576145	短信模板 CODE。您可以在控制台 模板管理 页面或 API 接口 AddSmsTemplate 的返回参数中获取短信模板 CODE。
action	String	否	DeleteSmsTemplate	系统规定参数。取值： DeleteSmsTemplate 。

4.4.2 返回参数

名称	类型	示例值	描述
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
message	String	OK	状态码的描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
templateCode	String	SMS73419576145	短信模板 CODE。

4.4.3 示例

请求示例

```
{
  "action": "DeleteSmsTemplate",
  "templateCode": "SMS13889962005"
}
```

复制

正常返回示例

JSON 格式

```
{
  "code": "OK",
  "message": "success",
  "requestId": "TxxfZdCz0sbhddVx"
}
```

复制

4.4.4 错误码

[查看本产品错误码](#)

4.5 修改模板

调用接口 `ModifySmsTemplate` 修改未通过审核的短信模板。

申请短信模板后，如果模板未通过审核，可以通过接口 **ModifySmsTemplate** 修改短信模板，并重新申请，提交审核。

模板内容需要符合[模板规范](#)。短信模板审核流程请参考[模板审核](#)流程。

说明： 一个自然日最多可以申请 100 个模板。

4.5.1 请求参数

名称	类型	是否必选	示例值	描述
templateType	Integer	是	1	短信类型。其中： 1： 验证码。 2： 短信通知。 3： 表示推广短信。
templateName	String	是	天翼云短信测试模板	模板名称，长度为 1~30 个字符。

名称	类型	是否必选	示例值	描述
templateContent	String	是	您正在申请手机注册，验证码为： \${code}，5分钟内有效！	模板内容，长度为 1~500 个字符。模板内容需要符合 模板规范 。 **说明:**修改模板时，请针对审核意见合理设计模板内容。
remark	String	是	当前的短信模板应用于双 11 大促推广营销	短信模板申请说明。请在申请说明中描述您的业务使用场景，长度为 1~100 个字符。
templateCode	String	是	SMS73419576145	短信模板 CODE。您可以在控制台 模板管理 页面或 API 接口 AddSmsTemplate 的返回参数中获取短信模板 CODE。

名称	类型	是否必选	示例值	描述
action	String	否	ModifySmsTemplate	系统规定参数。取值： ModifySmsTemplate 。

4.5.2 返回参数

名称	类型	示例值	描述
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
message	String	OK	状态码的描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
templateCode	String	SMS73419576145	短信模板 CODE。您可以使用模板 CODE 通过 API 接口 QuerySmsTemplate 或在控制台查看模板申请状态和结果。

4.5.3 示例

请求示例

```
{
  "action": "ModifySmsTemplate",
  "templateCode": "SMS13889962005",
  "templateType": 1,
  "templateContent": "您的验证码为${code},有效期为${time}分钟!",
  "templateName": "登陆验证码",
  "remark": "测试"
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success",
  "requestId": "xrfXI4nBFs6YN49U",
  "templateCode": "SMS13889962005"
}
```

复制

4.5.4 错误码

[查看本地产品错误码](#)

5. 签名

5.1 查询签名列表

申请短信签名后，您可以在控制台或通过接口 **QuerySmsSignList** 查看所有短信签名。

5.1.1 请求参数

名称	类型	是否必选	示例值	描述
action	String	否	QuerySmsSignList	系统规定参数。取值： QuerySmsSignList 。
pageIndex	Int	是	1	页码，即获取第几页的模板
pageSize	Int	是	10	个数，即每页包含多少条模板

5.1.2 返回参数

名称	类型	示例值	描述
code	String	OK	请求状态码。返回 OK 代表请求成功。其他错误码详见 错误码列表 。
message	String	OK	状态码描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
createTime	String	2022-04-15T03:15:06.178Z	短信签名的创建日期和时间。
data	Array		
data.reason	String	文件不能证明信息真实	审核备注。如果审核状态为 审

名称	类型	示例值	描述
		性，请重新上传	核通过 或 审核中 ，参数 Reason 显示为“无审核备注”。如果审核状态为 审核未通过 ，参数 Reason 显示审核的具体原因。
data.signName	String	天翼云	短信签名。
data.signPurpose	Int	1	签名用途 0：自用；2：他用；
data.signType	Int	1	签名类型 0：企事业单位的全称或简称；1：工信部备案网站的全称或简称；2：App 应用的全称或简称；3：公众号或小程序的全称或简称；4：电商平台店铺名的全称或简称；5：商标名的全称或简称
data.papersType	Int	1	证明类型 0：三证合一；1：企业营业执照；2：组织机构代码证书；3：社会信用代码证书；4：应用后台管理截图（个

名称	类型	示例值	描述
			人开发 APP) ;5: 网站备案后台截图 (个人开发网站) ;6: 小程序设置页面截图 (个人认证小程序) ;7: 商标注册书;8: 公众号设置页面截图 (个人认证公众号) ;
data.status	Int	0	签名审核状态。0: 审核中 1: 审核通过 2: 审核失败, 请在返回参数 Reason 中查看审核失败原因。
data.remark	String	测试	
data.reason	String	通过	

5.1.3 示例

请求示例

```
{
  "action": "QuerySmsSign",
  "pageIndex": 1,
  "pageSize": 10,
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success"
  "requestId": "cgudgor511imr7nh1ltg",
  "totalCount": 14,
  "data": [
    {
      "createTime": "2023-02-16T06:45:35.039Z",
      "remark": "测试 123",
      "signPurpose": 1,
      "signType": 3,
      "papersType": 0,
      "status": 2,
      "userId": "915e6709bc524b5b8285a0461c67ebc1",
      "reason": "ddd",
      "signName": "天翼云 7"
    }
  ]
}
```

复制

5.1.4 错误码

[查看本产品错误码](#)

5.2 添加签名

您可以通过短信服务 API 接口或云通信控制台申请短信签名，签名需要符合[签名规范](#)。

短信签名审核流程请参见[签名审核流程](#)。

说明： 个人用户每天最多可以申请一个短信签名。企业用户每天最多可以申请 100 个签名。

5.2.1 请求参数

名称	类型	是否必选	示例值	描述
action	String	是	AddSmsSign	系统规定参数。取值： AddSmsSign
signName	String	是	天翼云	签名名称。必须符合 签名规范 。
signPurpos	Integer	是	0	是否自用。0、自用； 1、他用
signType	Integer	是	1	签名来源。取值： 0：企事业单位的全称 或简称；1：工信部备 案网站的全称或简 称；2：App 应用的全 称或简称；3：公众号 或小程序的全称或简 称；4：电商平台店铺 名的全称或简称；5：

名称	类型	是否必选	示例值	描述
				商标名的全称或简称。 说明:签名来源为工信部备案网站的全称或简称时 ，请在申请说明中添加网站域名，加快审核速度。
papersType	Integer	是	0	证件类型: 0: 三证合一;1: 企业营业执照;2: 组织机构代码证书;3: 社会信用代码证书;4: 应用后台管理截图 (个人开发APP) ;5: 网站备案后台截图 (个人开发网站) ;6: 小程序设置页面截图 (个人认证小

名称	类型	是否必选	示例值	描述
				程序) ;7: 商标注册书;8: 公众号设置页面截图 (个人认证公众号)。
remark	String	是	当前的短信签名 应用于 双 11 大促推广营销	短信签名申请说明。 请在申请说明中详细描述您的业务使用场景，申请工信部备案网站的全称或简称，请在此处填写域名，长度不超过 200 个字符。
signFileList.N.FileSuffix	String	是	jpg	签名的证明文件格式，支持上传多张图片。当前支持 JPG、PNG 或 JPEG 格式的

名称	类型	是否必选	示例值	描述
				图片。个别场景下, 申请签名需要上传证明文件。详细说明请参见 签名规范 。 **说明: **他用也需要上传证明文件, 您可以上传营业执照附件或者其他相关的证明文件。
signFileList.N.FileContents	String	是		签名的资质证明文件经 base64 编码后的字符串。图片不超过 2 MB。个别场景下, 申请签名需要上传证明文件。详细说明请参见 签名规范 。

5.2.2 返回数据

名称	类型	示例值	描述
signName	String	天翼云	签名名称。
message	String	OK	状态码描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。

5.2.3 示例

请求示例

```
{
  "action": "AddSmsSign",
  "signName": "天翼云",
  "signPurpose": 0,
  "signType": 1,
  "papersType": 1,
  "remark": "测试",
  "signFileList": [
    {
      "FileSuffix": "jpg",
      "FileContents": ""
    }
  ]
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success",
  "requestId": "iesX5RP0v1ARaH3Y",
  "signName": "天翼云"
}
```

复制

5.2.4 错误码

[查看更多错误码。](#)

5.3 删除签名

调用接口 DeleteSmsSign 删除短信签名。

说明:

- 不支持删除正在审核中的签名。
- 短信签名删除后不可恢复，请谨慎操作。

5.3.1 请求参数

名称	类型	是否 必选	示例值	描述
signName	String	是	天翼云	短信签名。 **说明:**必须是本账号已申请的短信签名。

名称	类型	是否必选	示例值	描述
action	String	是	DeleteSmsSign	系统规定参数。取值： DeleteSmsSign 。

5.3.2 返回数据

名称	类型	示例值	描述
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
message	String	OK	状态码描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
signName	String	天翼云	短信签名。

5.3.3 示例

请求示例

```
{
  "action": "DeleteSmsSign",
  "signName": "天翼云"
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success",
  "requestId": "mxUo3xtDZ7bZJWLA",
  "signName": "天翼云"
}
```

复制

5.3.4 错误码

[查看更多错误码](#)。

5.4 修改签名

调用接口 `ModifySmsSign` 修改未审核通过的短信签名证明文件，并重新提交审核。

申请短信签名后，如果签名未通过审核，可以通过本接口修改短信签名证明文件，并重新申请，提交审核。

签名需要符合[签名规范](#)。短信签名审核流程请参见[签名审核流程](#)。

说明： 个人用户每天最多可以申请一个短信签名。企业用户每天最多可以申请 100 个签名。

5.4.1 请求参数

名称	类型	是否必选	示例值	描述
action	String	是	ModifySmsSign	系统规定参数。取值： ModifySmsSign 。
signName	String	是	天翼云	签名内容。 **说明：** 签名内容必须符合 签名规范 。
signPurpose	Integer	是	0	是否自用。 0、自用；1、他用
signType	Integer	是	1	签名来源。 取值：0：企事业单位的

名称	类型	是否 必 选	示例值	描述
				全 称 或 简 称;1:工信部 备案网站的 全 称 或 简 称;2:App 应 用的全称或 简称;3:公众 号或小程序 的全称或简 称;4:电商平 台店铺名的 全 称 或 简 称;5:商标名 的全称或简 称。 说明:签 名来源为工 信部备案网

名称	类型	是否必选	示例值	描述
				站的全称或简称 时，请在申请说明中添加网站域名，加快审核速度。
papersType	Integer	是	0	证件类型： 0：三证合一;1:企业营业执照;2:组织机构代码证书;3:社会信用代码证书;4:应用后台管理截图 (个人开发

名称	类型	是否必选	示例值	描述
				APP) ;5: 网站备案后台截图 (个人开发网站) ;6: 小程序设置页面截图 (个人认证小程序) ;7: 商标注册书;8: 公众号设置页面截图 (个人认证公众号)。
remark	String	是	当前的短信签名应用于双 11 大促推广营销	短信签名申请说明。请

名称	类型	是否必选	示例值	描述
				在申请说明中详细描述您的业务使用场景，申请工信部备案网站的全称或简称请在此处填写域名，长度不超过 200 个字符。
signFileList.N.FileSuffix	String	是	jpg	签名的证明文件格式，支持上传多张图片。当前支持 JPG、

名称	类型	是否必选	示例值	描述
				PNG、GIF 或 JPEG 格式的图片。个别场景下，申请签名需要上传证明文件。详细说明请参见 签名规范 。
signFileList.N.FileContents	String	是	R0lGODlhHAAMAKIHAKqqqsvLy0hISObm5vf394uLiwAA	签名的质证明文件经 base64 编码后的字符串。图片不超过 2 MB。 个别场景

名称	类型	是否必选	示例值	描述
				下，申请签名需要上传证明文件。 详细说明请参见 签名规范 。

5.4.2 返回数据

名称	类型	示例值	描述
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
message	String	OK	状态码描述。
requestId	String	F655A8D5D	请求 ID。
signName	String	天翼云	短信签名。

5.4.3 示例

请求示例

```
{
  "action": "ModifySmsSign",
  "signName": "天翼云",
  "signPurpose": 1,
  "signType": 2,
  "papersType": 1,
  "remark": "测试",
  "signFileList": [
    {
      "FileSuffix": "jpeg",
      "FileContents": "15235"
    }
  ]
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success",
  "requestId": "2BuGbZclnTraVxPK",
  "signName": "天翼云"
}
```

复制

5.4.4 错误码

[查看更多错误码。](#)

5.5 查询签名

调用接口 `QuerySmsSign` 查询短信签名申请状态。

申请短信签名后，您可以在控制台或通过接口 `QuerySmsSign` 查看短信签名的申请状

态。如果未通过审核，接口会返回未通过审核的原因，请针对具体原因修改签名并重新提交审核。

5.5.1 请求参数

名称	类型	是否必选	示例值	描述
signName	String	是	天翼云	签名名称。
action	String	是	QuerySmsSign	系统规定参数。取值： QuerySmsSign 。

5.5.2 返回参数

名称	类型	示例值	描述
code	String	OK	请求状态码。 返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
createTime	String	2019-01-08 16:44:13	短信签名的创建日期和时间。
message	String	OK	状态码描述。
reason	String	文件不能证明信息真实性， 请重新上传	审核备注。 如果审核状态为 审核通过 或 审核中 ，参数 Reason 显示

名称	类型	示例值	描述
			为“无审核备注”。 如果审核状态为 审核未通过 ，参数 Reason 显示审核的具体原因。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
signName	String	天翼云	短信签名。
signStatus	Integer	1	签名审核状态。其中： 0：审核中。 1：审核通过。 2：审核失败，请在返回参数 Reason 中查看审核失败原因。

5.5.3 示例

请求示例

```
{  
  "action": "QuerySmsSign",  
  "signName": "天翼云"  
}
```

复制

正常返回示例

```
{
```

```
"code": "OK",
"createTime": "2022-02-11T01:35:16.492Z",
"message": "success",
"reason": "",
"requestId": "uRzy63JqfvbAOMkh",
"signName": "天翼云",
"signStatus": 0
}
```

复制

5.5.4 错误码

[查看本产品错误码](#)

6.发送短信

SendSms 接口是短信发送接口，支持在一次请求中向多个不同的手机号码发送同样内容的短信。

调用该接口发送短信时，请注意：发送短信会根据发送量计费，价格请参见[产品定价](#)。

6.1 请求参数

名称	类型	是否必选	示例值	描述
action	String	是	SendSms	系统规定参数。取值： SendSms 。
phoneNumber	String	是	1381111*****	接收短信的手机号码。格式：

名称	类型	是否必选	示例值	描述
				国内短信:无任何前缀的 11 位手机号码, 例如 1381111****。
signName	String	是	天翼云	短信签名名称。请在控制台的 签名管理 页签下 签名名称 一列查看。 **说明: **必须是已添加、并通过审核的短信签名。
templateCode	String	是	SMS73419576145	短信模板 ID。请在控制台的 模板管理 页签下 模板 CODE 一列查看。 **说明: **必须是已添加、并通过审核的短信模板。
templateParam	String	否	{"code":"1111"}	短信模板变量对应的实际值, JSON 格式。 **说明: **如果 JSON 中需要带换行符, 请参照标准的 JSON 协议处理。

名称	类型	是否必选	示例值	描述
extendCode	String	否	90999	上行短信扩展码，上行短信，指发送给通信服务提供商的短信，用于定制某种服务、完成查询，或是办理某种业务等，需要收费的，按运营商普通短信资费进行扣费。 **说明： **无特殊需要此字段的用户请忽略此字段。
sessionId	String	否	123456	客户自带短信标识，在状态报告中会原样返回。 **说明： **无特殊需要此字段的用户请忽略此字段。

6.2 返回参数

名称	类型	示例值	描述
code	String	OK	请求状态码。

名称	类型	示例值	描述
			返回 OK 代表请求成功。 其他错误码详见 错误码列表 。
message	String	OK	状态码描述。
requestId	String	F655A8D5B967	请求 ID。

6.3 示例

请求示例

```
{
  "action": "SendSms",
  "signName": "京东",
  "phoneNumber": "13301110000",
  "templateCode": "SMS73419576145",
  "templateParam": "{\"code\":\"123456\",\"time\":\"1\"}",
  "extendCode": "123"
}
```

复制

正常返回示例

```
{
  "code": "OK",
  "message": "success",
  "requestId": "TxxfZdCz0sbhddVx"
}
```

7.查询发送记录

通过调用 QuerySendDetails 接口，可以根据短信发送日期查看发送记录和短信内容，也

可以添加发送流水号，根据流水号查询指定日期指定请求的发送详情。如果指定日期短信

发送量较大，可以分页查看。指定每页显示的短信详情数量和查看的页数，即可分页查看发送记录。

7.1 请求参数

名称	类型	是否必选	示例值	描述
action	String	是	QuerySendDetails	系统规定参数。取值： QuerySendDetails 。
currentPage	Long	是	0	分页查看发送记录，指定发送记录的当前页码。
pageSize	Long	是	10	分页查看发送记录，指定每页显示的短信记录数量。取值范围为 1~50。
phoneNumber	String	是	1590000****	接收短信的手机号码。格式： 国内短信：11 位手机号码，例如 1590000****。国际/港澳台消息：国际区号+号码，例如 8520000****。

名称	类型	是否必选	示例值	描述
sendDate	String	是	2022-03-28	短信发送日期，支持查询最近 30 天的记录。格式为 yyyy-MM-dd，例如 20181225。
signName	String	否	签名	对应发送内容的签名。
templateCode	String	否	SMS73419576145	对应发送内容的模板编号。
requestId	String	否	49667481793	对用发送时返回的流水号。

7.2 返回数据

名称	类型	示例值	描述
code	String	OK	请求状态码。返回 OK 代表请求成功。 其他错误码，请参见 错误码列表 。
message	String	OK	状态码的描述。
requestId	String	mOiogcaBxn6YgaEQ	请求 ID。
data	Array		短信发送明细。

名称	类型	示例值	描述
totalCount	String	1	短信发送总条数。

7.3 示例

请求示例

```
{
  "action": "QuerySendDetails",
  "phoneNumber": "18616065511",
  "sendDate": "2022-01-18",
  "currentPage": 1,
  "pageSize": 10,
  "signName": "天翼云",
  "templateCode": "SMS13889962005",
}
```

复制

正常返回示例

```
{
  "code": 0,
  "message": "success",
  "totalCount": 1,
  "data": [
    {
      "phone": "13301161312",
      "templateCode": "SMS81136825168",
      "signName": "天翼云",
      "total": 0,
      "index": 0,
      "type": 0,
      "content": "您的验证码为 123,有效期为 5 分钟!",
      "status": 3,
      "statusCode": "",
      "receiveCode": "0",
    }
  ]
}
```

```
    "receiveTime": "2022-03-28T06:29:34.819Z",  
    "deliverCode": "0",  
    "deliverTime": "2022-03-28T06:29:34.824Z",  
    "reportCode": "DELIVRD",  
    "reportTime": "2022-03-28T06:29:35.171Z",  
    "notifyTime": "2022-03-28T06:29:35.171Z"  
  }  
],  
  "requestId": "dNFgPWV5Db14vg38"  
}  
复制
```

7.4 错误码

访问[错误码列表](#)查看更多错误码。