

Zalo ZNS API用户对接指南

Step 1 登录“<https://www.nxcloud.com/login>”，获取“AccessKey”



Step 2 打开“<https://github.com/nxtele/http-api-document/wiki/API%E6%8E%A5%E5%8F%A3%E8%B0%83%E7%94%A8%E7%BA%A6%E5%AE%9A>”

1) 确保已有所有“公共参数”

公共参数

调用API接口时请求头Header中必须传入的参数，目前支持的公共参数有：

参数名称	参数类型	是否必填	示例值	参数描述
accessKey	String	是	fme2na3kdi3ki	用户身份标识
ts	String	是	1655710885431	当前请求的时间戳（单位是毫秒），牛信服务端允许用户端请求最大时间误差为60000毫秒
bizType	String	是	1	业务类型
action	String	是	send	API接口方法，参数值参照各API接口文档中的说明
sign	String	是	6e9506557d1f289501d333ee2c365826	API入参参数签名， 签名算法
algorithm	String	否	md5	签名哈希算法，可用md5, sha256（默认md5）

bizType参数说明：

参数值	业务描述
1	号码检测
2	whatsapp业务
3	短信
4	DID业务
5	隐私号
6	OTA
7	Viber
8	voice语音业务
9	Zalo 通知服务业务

1) 接下来，找到与您系统语言相匹配的语言。

例如，如果您的系统语言是Java，请参考以下代码片段：

复制此代码并填入Step 2 - 1) 中获得的所有公共参数。

计算签名代码示例

Java

```
/**
 * 生成接口参数签名demo
 */
@Test
public void generateSignDemo() {
    // header参数
    Map<String, String> headers = new HashMap<>(8);
    headers.put("accessKey", "fme2na3kdi3ki");
    headers.put("ts", "1655710885431");
    headers.put("bizType", "1");
    headers.put("action", "send");

    // 业务参数
    JSONObject postData = new JSONObject();
    postData.put("id", 10001);
    postData.put("name", "牛小信");
    String body = postData.toString();

    // accessKey对应的密码
    String accessSecret = "abciiko2k3";

    String sign = calcSign(headers, body, accessSecret);
    log.info("sign: {}", sign); // sign: 87c3560d3331ae23f1021e2025722354
}

/**
 * 计算sign签名
 *
 * @param headers 请求头中的公共参数
 * @param body body中的json字符串
 * @param accessSecret 密钥
 * @return
 */
private String calcSign(Map<String, String> headers, String body, String accessSecret) {
    StringBuilder raw = new StringBuilder();

    // step1: 拼接header参数
    raw.append("accessKey=").append(headers.get("accessKey")).append("&action=").append(headers.get("action"))
        .append("&bizType=").append(headers.get("bizType")).append("&ts=").append(headers.get("ts"));
    log.info("step1: {}", raw); // step1: accessKey=fme2na3kdi3ki&action=send&bizType=1&ts=1655710885431

    // step2: 拼接body参数
```

3) 然后，按“Enter”以获取“签名算法”。

```
log.info("step2: {}", raw); // step2: accessKey=fme2na3kdi3ki&action=send&bizType=1&ts=1655710885431&body={}&accessSecret=abciiko2k3

// step3: 拼接accessSecret
raw.append("&accessSecret=").append(accessSecret);
log.info("step3: {}", raw); // step3: accessKey=fme2na3kdi3ki&action=send&bizType=1&ts=1655710885431&body={}&accessSecret=abciiko2k3

// step4: MD5算法加密,结果转换成十六进制小写
String sign = DigestUtils.md5Hex(raw.toString());
log.info("step4: sign={}", sign); // step4: sign=87c3560d3331ae23f1021e2025722354

return sign;
}
```

Step 3 打开“<https://www.nxcloud.com/zh-developer-page>”

1) 确认您已在Step 2中获取了您的“签名算法”，然后继续下一步。

鉴权机制

鉴权规则请参考地址：[API 接口调用约定](#)

完成后即可放入“签名算法”

请求参数

header 参数：

参数名	类型	必选	示例值	说明
accessKey	String	是	fme2na3kdi3ki	用户身份标识
ts	String	是	1655710885431	当前请求的时间戳（单位是毫秒），牛信服务端允许用户端请求最大时间误差为 60 秒
bizType	String	是	9	Zalo 业务类型，取固定值“9”
action	String	是	mt	Zalo 业务操作，取固定值“mt”
sign	String	是	6e9506557d1f289501d333ee2c365826	API 入参参数签名， 签名算法

body 参数：

2) 发送 ZNS 通知时，请确保获取所有“header参数和body参数”

a) header参数

- i) accessKey，可从 NXCLOUD-客户端 首页获取。
- ii) ts，请参阅说明
- iii) bizType，请参阅说明，ZNS 是 9
- iv) action，mt
- v) sign，从“Step 2”复制代码

b) body参数

- i) appkey，NXCLOUD 提供给您的（请向我们的工作人员获取此信息）
- ii) phone，请参阅说明
- iii) function Type，请参阅说明
- iv) messageDate，请参阅说明
- v) templateID，NXCLOUD 提供给您的（请向我们的工作人员获取此信息）
- vi) templateDate，请参阅说明

header 参数:

参数名	类型	必选	示例值	说明
accessKey	String	是	fme2na3kdi3ki	用户身份标识
ts	String	是	1655710885431	当前请求的时间戳（单位是毫秒），牛信服务端允许用户端请求最大时间误差为 60 秒
bizType	String	是	9	Zalo 业务类型，取固定值“9”
action	String	是	mt	Zalo 业务操作，取固定值“mt”
sign	String	是	6e9506557d1f289501d333ee2c365826	API 入参参数签名， 签名算法

body 参数:

参数名	类型	必选	示例值	说明
appkey	String	是	pem28kje	应用 appkey
phone	String	是	86158xxxx1795	消息接收方的 zalo 号码，需要带国码。如 86158xxxx1795
functionType	String	是	template	zalo 消息功能类型，消息支持以下功能类型的发送： <code>template</code>
messageData	JsonObject	是	参照请求示例	zalo 消息体参数

3) 最后，启动消息服务方法发送 Zalo ZNS 消息

发送消息

通过 API 发起 zalo 号码消息发送服务

- URL: `https://api.nxcloud.com/v1/zalo/mt`
- Method: `POST`
- Content-Type: `application/json`
- 需要鉴权: `是`

监控日志