# Postman使用MD5加密签名的两种方法

## 软件介绍

Postman是用户在开发、调试网络程序或者是网页时候来跟踪网络请求的，可以发送几乎所有类型的HTTP请求！Postman在发送网络HTTP请求方面可以说是网络调试类产品中的代表产品。



## 操作系统

Postman适用于不同的操作系统， Windows X32、Windows X64、Linux和Mac系统。

## 常用功能

1、Postman请求方式包含get、post、put等常用方式；

2、请求格式包含form-data、raw、binary等常用格式；

3、collection是类似文件夹，可以把相关项目的请求放在一起方便管理和测试；

4、Pre-request Script，主要用于编写数据发送前的自定义脚本，实现数据的自动化签名或设置，此功能为本篇文章的主要说明内容；

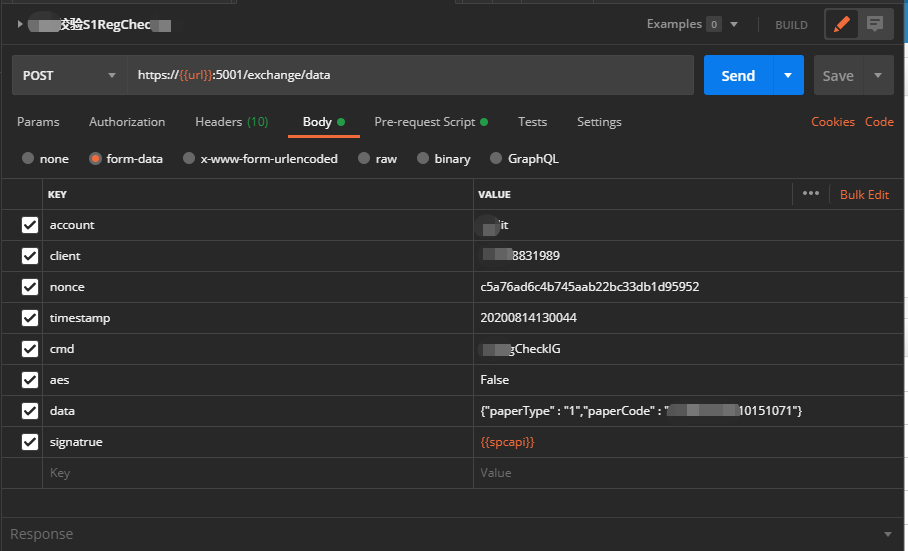
5、其它常用功能如设置环境变量、暗黑模式、导出文档等就不一一介绍了；

## MD5加密签名

Pre-request Script选项卡，主要用于编写数据发送前的自定义脚本，通过一下示例来说明一下。

**签名示例一：**

Body选项卡：



Pre-request Script选项卡：

var params = request.data;

var signKey = "415233197335409695";

var md5String = params.cmd +

    params.account +

    params.nonce +

    params.client +

    params.timestamp +

    params.data +

    signKey;

console.log("md5String---->" + md5String);

var md5 = CryptoJS.MD5(md5String).toString();

console.log("md5 digest---->" + md5);

pm.environment.set("apising", md5);

1、var params = request.data;声明可以从postman的body选项卡中获取参数，无需在单独声明，当body选项卡中的参数改变时脚本中数据会自动更新；

2、var signKey = "415233197335409695";声明签名密钥，由于数据中是不包含密钥信息的，所以需要单独声明；

3、var md5String声明了需要签名的信息包含哪些参数以及信息方式和顺序；

var md5String = params.cmd +

params.account +

params.nonce +

params.client +

params.timestamp +

params.data +

signKey;

4、console.log("md5String---->" + md5String);定义了输出的信息以及完整的日志，方便分析数据和调试；

5、var md5 = CryptoJS.MD5(md5String).toString();声明了md5信息并转换为字符串，方便和程序或服务端的签名信息核对，验证签名是否一致；

6、console.log("md5 digest---->" + md5);输出md5签名信息并记录日志；

7、pm.environment.set("apising", md5);通过postman内置的函数将以上签名结果设置为变量，名称为apising，在body的选项卡中将value值设置为｛｛apising｝｝即可，在调用调试接口前会自动执行脚本命令并获取到变量值。

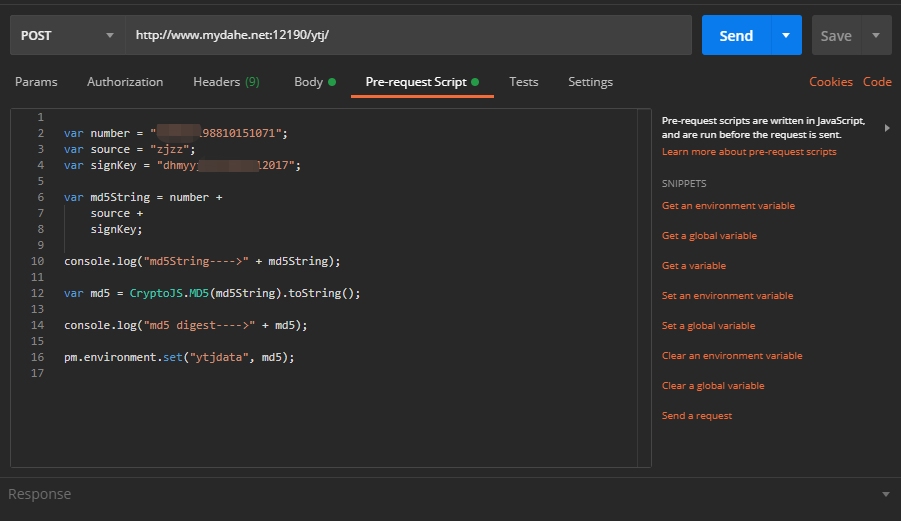
**签名示例二：**

输入license和Site Message，点击“next”；

Body选项卡：

Body选项卡内可以是任意格式的参数；

Pre-request Script选项卡：



1、var number = var "415233197335409695";

signKey = "415233197335409695";

……

声明参数、签名密钥等，可以把需要都单独声明一下；

2、var md5String声明了需要签名的信息包含哪些参数以及信息方式和顺序；

var md5String = params.number+

signKey;

4、console.log("md5String---->" + md5String);定义了输出的信息以及完整的日志，方便分析数据和调试；

5、var md5 = CryptoJS.MD5(md5String).toString();声明了md5信息并转换为字符串，方便和程序或服务端的签名信息核对，验证签名是否一致；

6、console.log("md5 digest---->" + md5);输出md5签名信息并记录日志；

7、pm.environment.set("ytjdata", md5);通过postman内置的函数将以上签名结果设置为变量，名称为ytjdata，在body的选项卡中将value值设置为｛｛ytjdata｝｝即可，在调用调试接口前会自动执行脚本命令并获取到变量值。

以上两种情况包含了自动获取签名参数和自定义需要签名的参数，也可以通过脚本命令实现一些其他的特殊参数、变量等。