



1、群里自定义机器人，CURL调用或三方以WebHook回调机器人

(1) 开发指南

- <https://open.feishu.cn/document/ukTMukTMukTM/ucTM5Yjl3FTQ24yNxkjN>

(2) 开发步骤

- 只有在群聊中才能添加自定义机器人，首先先创建一个群聊



(3) 添加一个自定义机器人



添加机器人



Q 搜索



自定义机器人

通过webhook将自定义服务的消息推...

4

添加



多维表格助手

更多模板用法 <https://base-templ...>

添加



审批

简单、高效、开放的审批工具

添加



飞书OKR

简单实用的团队目标管理工具

添加



飞书问卷

极简又专业的「问卷调查、签到考勤...

添加



飞书提醒

有事我就会提醒你

添加



汇报

高效、智能、清晰、流畅的汇报软件

添加



飞书机器人助手

无需代码能力，3 分钟轻松搭建机器...

添加



飞书帮助中心

feishu.cn/hc 一键直击飞书用法最全...

添加



健康报备

及时报备情况，守护员工健康

添加

添加机器人

的消息推送至...



×

从群聊中移除

使用说明

取消

保存

描述*

通过webhook将自定义服务的消息推送至飞书

23/256

webhook 地址

https://open.feishu.cn/open-apis/bot/v2/hook/7a51e9...

复制

5

请保管好此 webhook 地址。不要公布在 Github、博客等可公开查阅的网站上。地址泄露后可能被恶意调用发送垃圾信息

安全设置

☐ 自定义关键词?

☐ IP 白名单?

☒ 签名校验?

rKmxgwCmGJbTDTygJnuTKg

重置

复制

6

密钥如上，签名方法详见 [帮助文档](#)

这一串是密钥，后续curl调用飞书机器人发通知时，curl需要通过密钥来生成签名(token)携带着，具体生成密钥的步骤见这里的帮助文档

7

(3) CURL调用飞书的自定义机器人给群里发消息

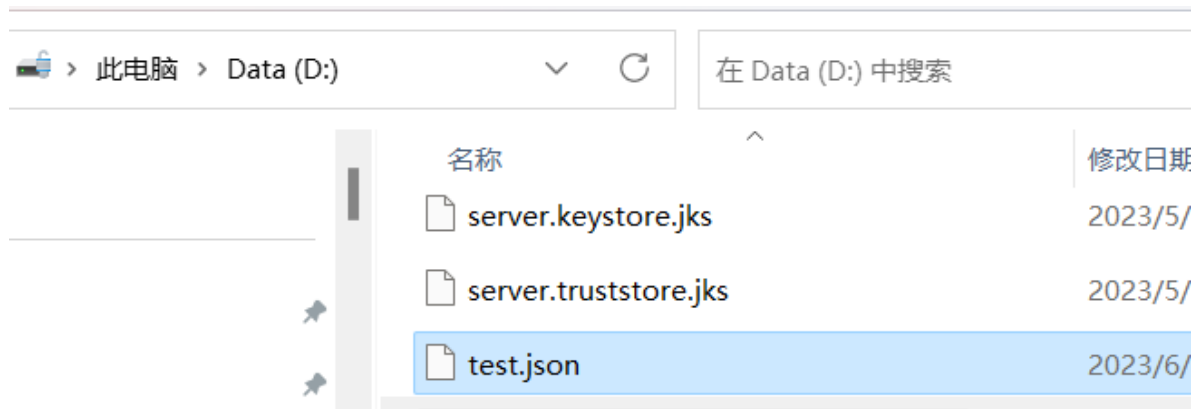
- kotlin代码：先通过自定义机器人的secert密钥生成sign签名

复制代码

```
1 fun main(args: Array<String>) {
2     val secret = "rKmxgwCmGJbTDTygJnuTKg" //飞书自定义机器人的密钥
3     val timestamp = 1685975648 //时间戳:秒,改成1个小时内的时间戳即可
4     System.out.printf("sign: %s", genSign(secret, timestamp.toString()))
5 }
6 private fun genSign(secret: String, timestamp: String): String {
7     //把timestamp+"\n"+密钥当做签名字符串
8     val stringToSign = timestamp.plus("\n").plus(secret)
9     //使用HmacSHA256算法计算签名
10    val mac: Mac = Mac.getInstance("HmacSHA256")
11    mac.init(SecretKeySpec(stringToSign.toByteArray(), "HmacSHA256"))
12    val signData: ByteArray = mac.doFinal(byteArrayOf())
13    return String(Base64.encodeBase64(signData))
14 }
```

java >

- 创建一个test.json文件，填充timestamp时间戳、sign签名



复制代码

```
1 {  
2   "timestamp": "1685975648",  
3   "sign": "A2Bi44VrSm+/b0ydcGMpZMeSLt5fKcoqLw/VQdSJDiU=",  
4   "msg_type": "text",  
5   "content": {  
6     "text": "request example"  
7   }  
8 }
```

java >

- 以CURL方式：调用飞书提供的自定义机器人的Webhook，即调用自定义机器人来给群里发消息

复制代码

```
1 //curl -X 跟请求方式 -H 跟请求头信息 --data 跟请求数据(这里数据是以文件的形式) 最后跟飞书自定义机器人的  
  webhook地址  
2 D:\>curl -X POST -H "Content-Type: application/json;charset=UTF-8" --data  
  @test.json https://open.feishu.cn/open-apis/bot/v2/hook/7a51e902-8066-4ce9-937b-e37582d71092  
3
```

java >

请求

```
D:\>curl -X POST -H "Content-Type: application/json;charset=UTF-8" --data @test.json
https://open.feishu.cn/open-apis/bot/v2/hook/7a51e902-8066-4ce9-937b-e37582d71092
{"StatusCode":0,"StatusMessage":"success","code":0,"data":{"msg":"success"}}
```

响应

- 飞书群里的通知消息



(4) SkyWalking集成Feishu Hook来给飞书的自定义机器人发消息

- <https://github.com/apache/skywalking/blob/v8.5.0/docs/en/setup/backend/backend-alarm.md>

```
1 feishuHooks:
2   textTemplate: |-
3     {
4       "msg_type": "text",
```

复制代码

```

5      "content": {
6          "text": "Apache SkyWalking Alarm: \n %s."
7      },
8      "ats": "feishu_user_id_1,feishu_user_id_2"
9  }
10 webhooks:
11     - url: https://open.feishu.cn/open-apis/bot/v2/hook/dummy_token
12       secret: dummysecret

```

java >






- Skywalking告警规则的设置位置
 - alarm-settings.yml中是配置一些默认的Scope（服务，实例，端点）的单个规则和组合规则

> Data (D:) > software > apache-skywalking-apm-9.4.0 > apache-skywalking-apm-bin > config

名称	修改日期	类型
ui-initialized-templates	2023/6/5 15:27	文件夹
zabbix-rules	2023/6/5 15:27	文件夹
alarm-settings.yml	2023/6/5 18:01	YML 文件
application.yml	2022/12/12 3:18	YML 文件

- 如果要自定义一些告警规则，需要先在core.oal中自定义Scope（服务，实例，端点）的指标名称

Data (D:) > software > apache-skywalking-apm-9.4.0 > apache-skywalking-apm-bin > config > oal

名称	修改日期	类
 browser.oal	2022/12/12 3:18	O
 core.oal	2023/6/5 15:48	O
 disable.oal	2022/12/12 3:18	O
 dotnet-agent.oal	2022/12/12 3:18	O
 java-agent.oal	2022/12/12 3:18	O

```
工作文件.txt x 每日任务记录.txt x 暂时记录.txt x README.txt x core.oal x alarm-settings.yml x application.yml x test.json x
8 // Service Instance Scope metrics
9 service_instance_sla = from(ServiceInstance.*).percent(status == true);
0 service_instance_resp_time = from(ServiceInstance.latency).longAvg();
1 service_instance_cpm = from(ServiceInstance.*).cpm();
2
3 // Endpoint scope metrics
4 endpoint_cpm = from(Endpoint.*).cpm();
5 endpoint_resp_time = from(Endpoint.latency).longAvg();
6 endpoint_sla = from(Endpoint.*).percent(status == true);
7 endpoint_percentile = from(Endpoint.latency).percentile(10); // Multiple values including p50,
p90, p95, p99
8 endpoint_mq_consume_latency = from((str->long)Endpoint.tag["transmission.latency"]).filter(type
RequestType.MQ).filter(tag["transmission.latency"] != null).longAvg();
9 endpoint_abnormal = from(Endpoint.*).filter(httpResponseStatusCode in [404,500,503]).count();
0 // Endpoint relation scope metrics
```

- 然后在alarm-settings.yml中引入core.oal自定义的指标名称，并配置告警规则，和告警后回调飞书自定义机器人

```
件.txt x 每日任务记录.txt x 暂时记录.txt x README.txt x core.oal x alarm-settings.yml x application.yml x test.json x
# http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
# Sample alarm rules.
rules:
  # Rule unique name, must be ended with `_rule`.
  endpoint_abnormal_rule:
    metrics-name: endpoint_abnormal
    threshold: 100
    op: "<"
    period: 10
    count: 1
    message: 在过去的2分钟到10分钟内 {name} 端点异常一次
feishuHooks:
  textTemplate: |-
    {
      "msg_type": "text",
      "content": {
        "text": "Apache SkyWalking Alarm: \n %s."
      }
    }
  webhooks:
    - url: https://open.feishu.cn/open-apis/bot/v2/hook/7a51e902-8066-4ce9-937b-e37582d71092
      secret: rKmxgwCmGJbTDTygJnuTKg
```

复制代码

```
1 # Sample alarm rules.
2 rules:
3   # Rule unique name, must be ended with `_rule`.
4   endpoint_abnormal_rule:
```

```
5    metrics-name: endpoint_abnormal
6    threshold: 100
7    op: "<"
8    period: 10
9    count: 1
10   message: 在过去的2分钟到10分钟内 {name} 端点异常一次
11
12   feishuHooks:
13     textTemplate: |-
14       {
15         "msg_type": "text",
16         "content": {
17           "text": "Apache SkyWalking Alarm: \n %s."
18         }
19       }
20   webhooks:
21     - url: https://open.feishu.cn/open-apis/bot/v2/hook/7a51e902-8066-4ce9-937b-e37582d71092
22       secret: rKmxgwCmGJbTDTygJnuTKg
```

java >

