

流量录制回放及CI/CD自动化脚本

录制篇

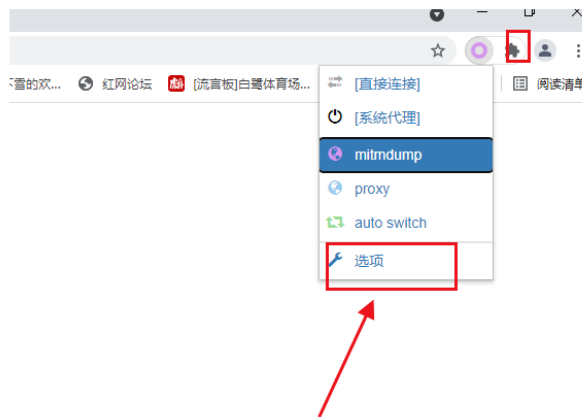
安装mitmproxy

1. pip install mitmproxy
2. 安装后提示需要路径加环境变量
C:\Users\Administrator\AppData\Roaming\Python\Python36\Scripts将这个路径添加到环境变量
3. 检查是否安装成功 mitmdump --version (windows 不支持mitmproxy) 以下截图表示安装成功

```
C:\Users\Administrator>mitmdump --version
Mitmproxy: 5.0.0
Python:    3.6.8
OpenSSL:   OpenSSL 1.1.0j  20 Nov 2018
Platform:  Windows-10-10.0.19041-SP0
```

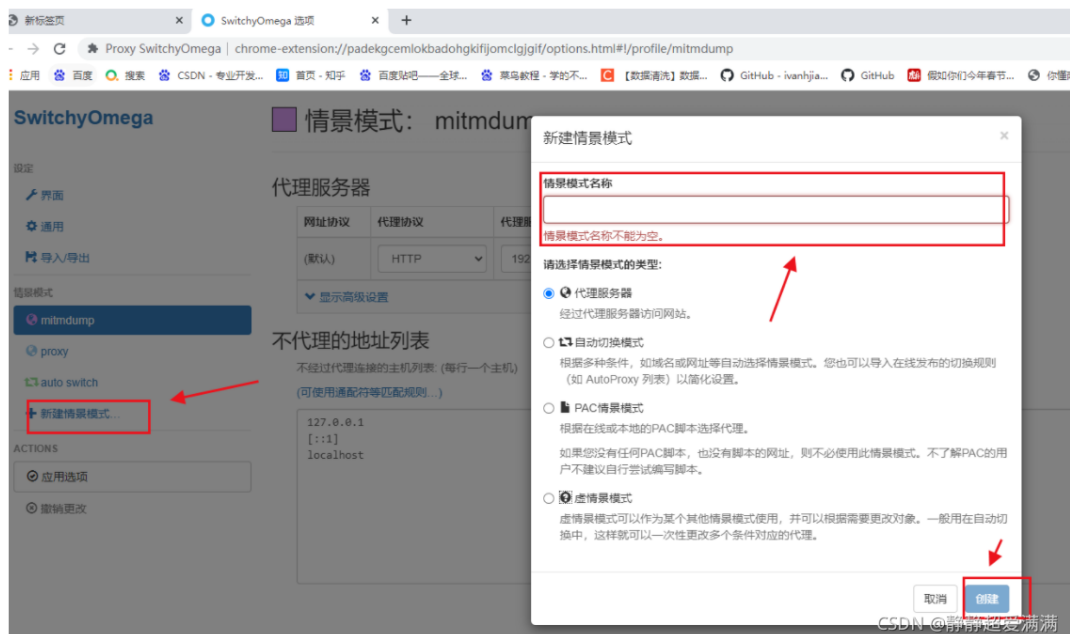
Chrome代理设置

1. SwitchyOmega插件下载 (通过chrome应用商城)
2. SwitchyOmega的使用
 1. 点击选项



CSDN @静静超爱满满

2. 新建情境模式



3. 设置代理

SwitchyOmega

情景模式: mitmproxy



代理服务器

网址协议	代理协议	代理服务器	代理端口
(默认)	HTTP	127.0.0.1	8080
http://	(同默认)	127.0.0.1	8080
https://	(同默认)	127.0.0.1	8080
ftp://	(同默认)	127.0.0.1	8080

不代理的地址列表

不经过代理连接的主机列表: (每行一个主机)
(可使用通配符等匹配规则...)

127.0.0.1
:::1
localhost

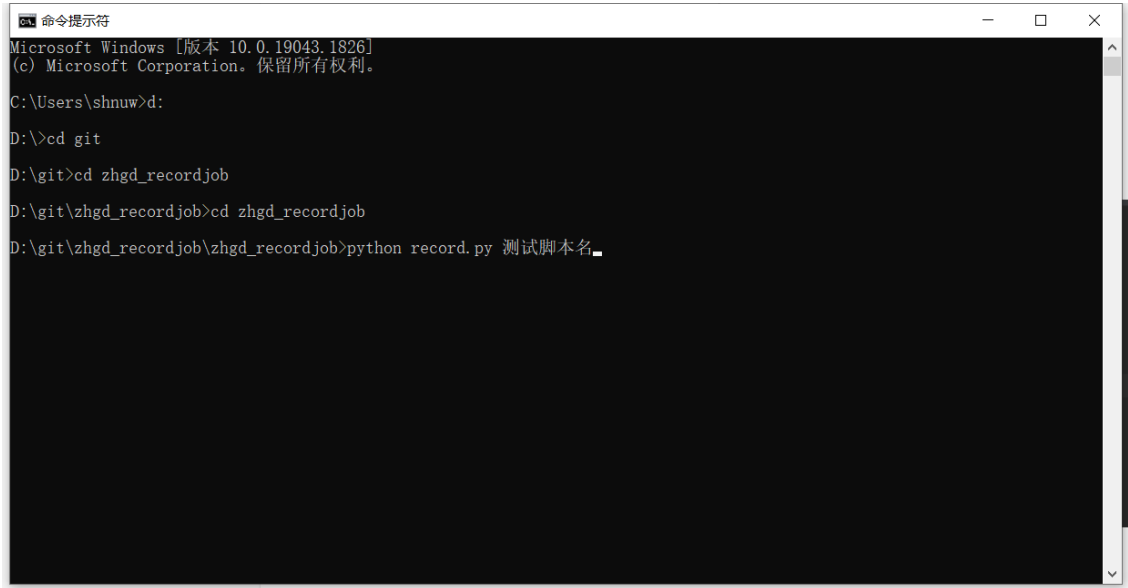
录制流程

1. pull git最新代码 git地址 http://git.luban.fit/wuguojun/zhgd_recordjob
2. 配置config, 找到config.yaml文件, 小tips, 如果不想往正式群发, Model可设置为test

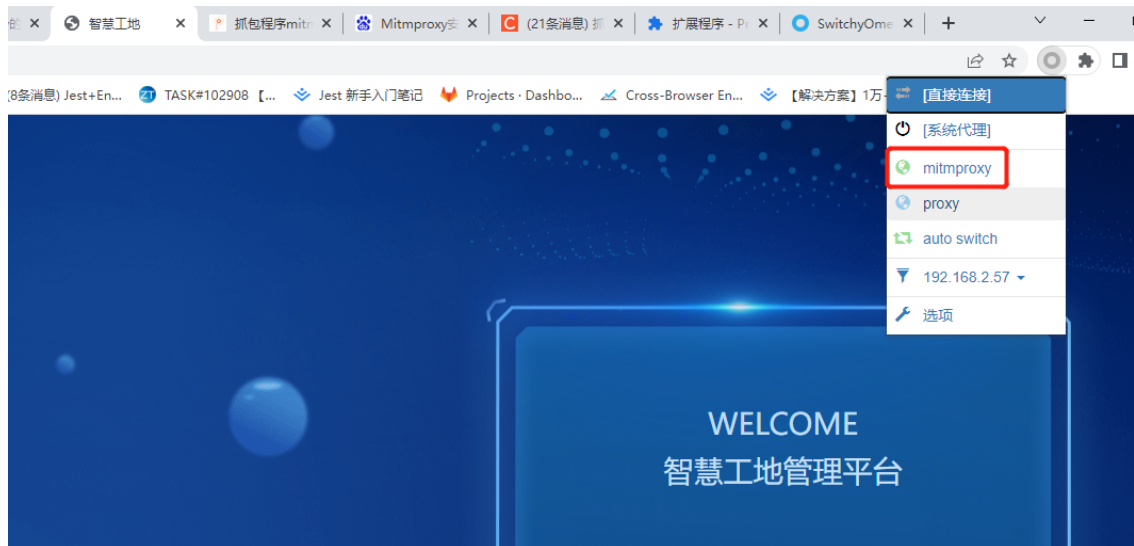


3. 通过cmd, 切换到record.py路径下, 执行命令

```
python record.py <测试脚本名>
```



4. 切换chrome浏览器模式，为代理模式



5. 手动点击页面进行操作，后台自动录制脚本（建议登录后，直接操作，脚本会重新登录防止token过期）
6. 录制完毕后，ctrl+c,会触发停止录制，同时会重新跑一次刚录制的脚本，即，录制完毕

回放篇

配置规则

1. 找到rules.yaml
2. 提取耦合字段，如id等

```
- Path : /things/v1/worker  
  Method : POST  
  Location : resp.data.id  
  Type: GETVALUE
```

Path : 接口url路径

Method: 接口类型

Location : 提取字段的路径, resp表示从response获取, 以'.'风格, 例如: resp.data.id,表示, 从response, data中找到id, 做提取

Type: GETVALUE 表示提取该字段

- 校验比对中的忽略校验

- 特定url的特定字段忽略校验

```
- Path: /things/v1/news/ui/list?pageSize=20&metaType=PARTY_NEWS&type=COMMONS&pageNo=1
  Method: POST
  Location: data_items_cover
  Type: IGNORE
  Contain: False
```

Path : 表示只有这个url, 才会触发忽略

Method : 配合Path, 确定唯一的接口

Location : 以“_”为分隔, 表示层级路径, data_items_cover, 表示忽略, response中, data下的items中的cover

Type : IGNORE 表示忽略校验

Contain : 是否模糊匹配字段, False表示精确匹配

- 特定url的模糊匹配字段忽略校验

```
- Path: /things/v1/dict/option/month
  Method: GET
  Location: data_label
  Type: IGNORE
  Contain: True
```

- 所有url的特定字段忽略校验

```
- Path : ALL
  Method : ALL
  Location : id
  Type: IGNORE
  Contain : False
```

Path : ALL, 表示, 任何接口只要是这个字段, 通通忽略校验

Method: ALL, 此时method也为ALL

- 所有url的模糊匹配字段忽略校验

```
- Path : ALL
  Method : ALL
  Location : time
  Type: IGNORE
  Contain : True
```

Contain: True,表示模糊匹配忽略校验

配置执行规则

1. 找到config.yaml文件

```
Setting :  
  Status : doc  
  Model : working  
  Url : http://192.168.2.57:9876/things/v1/login  
  User : testapi  
  PSW : 111111  
  Scope :  
    - 高支模-点位管理_测试脚本文件夹_1659421985
```

Status : doc, 则表示, 所有的用例都要

Model : working, 则表示, 在正式的企业微信群推送结果

```
Setting :  
  Status : debug  
  Model : working  
  Url : http://192.168.2.57:9876/things/v1/login  
  User : testapi  
  PSW : 111111  
  Scope :  
    - 高支模-点位管理_测试脚本文件夹_1659421985
```

Status : debug, 则表示, 只跑选定的用例, 用例范围在Scope中获取

Scope : 填入需要跑的用例名称

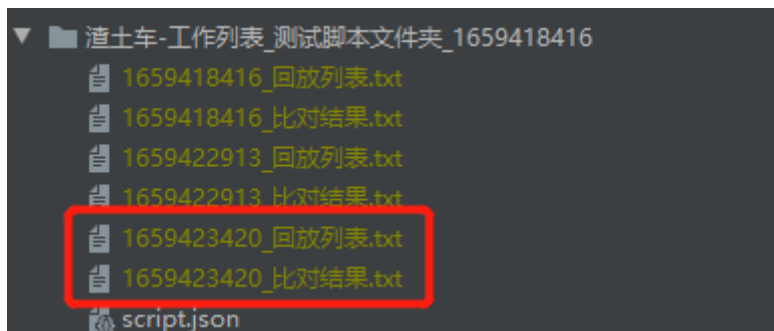
执行回放

执行命令

```
python createScript.py
```

查看log文件

- 每次执行会生成三个log文件



每个测试用例文件夹下, 都会有两个时间戳相同的文件

- 时间戳_回放列表

```

PDYd2z4yI, http://192.168.2.57:9876/things/v1/asset/tree/project, method: GET

Cy4y0t1XeK, http://192.168.2.57:9876/things/v1/component/ui/data?code=ElevatorDeviceManagement&deviceType=Elevator&opt=device-ts, method: GET

IbKI8GbW5, http://192.168.2.57:9876/things/v1/device/ui/list?type=Elevator&pageSize=20&pageNo=1, method: GET

mIIV70QZgX, http://192.168.2.57:9876/things/v1/worker/group/ui/form/tree, method: GET

cg0CYUbr0F, http://192.168.2.57:9876/things/v1/site/ui/select, method: GET

p8c1YUa2IG, http://192.168.2.57:9876/things/v1/device/schema/ui/select?type=Elevator, method: GET

CtuaIboS3D, http://192.168.2.57:9876/things/v1/worker/ui/list?siteStatus=1&pageSize=10&pageNo=1, method: GET

WLcgoX6Q9u, http://192.168.2.57:9876/things/v1/device/slim/ui/list?type=Camera&pageSize=10&pageNo=1, method: GET

TA4LDPXb88, http://192.168.2.57:9876/things/v1/device/image/upload?, method: POST

otKAc2WzIE, http://192.168.2.57:9876/things/v1/device, method: POST

n17N2H5n7a, http://192.168.2.57:9876/things/v1/device/ui/list?type=Elevator&pageSize=20&pageNo=1, method: GET

gC9fP8Ddgf, http://192.168.2.57:9876/things/v1/device/155402835379806208, method: GET

L5n0eaRHX0, http://192.168.2.57:9876/things/v1/worker/ui/list?siteStatus=1&value=1547872581941612544&op=loadOptions, method: GET

M46KWfPCgK, http://192.168.2.57:9876/things/v1/worker/group/ui/form/tree, method: GET

KDXwSeG6W, http://192.168.2.57:9876/things/v1/device/schema/ui/select?type=Elevator, method: GET

xsb9Q6kQy, http://192.168.2.57:9876/things/v1/site/ui/select, method: GET

oLYmosrMKW, http://192.168.2.57:9876/things/v1/device, method: POST

q20C9Am77e, http://192.168.2.57:9876/things/v1/device/ui/list?type=Elevator&pageSize=20&pageNo=1, method: GET

k23G2p17TH, http://192.168.2.57:9876/things/v1/device/ui/list?type=Elevator&pageSize=20&pageNo=1, method: GET

AIIgQae05g, http://192.168.2.57:9876/things/v1/device/ui/list?type=Elevator&pageSize=20&pageNo=1&name=testap11, method: GET

w9XbtC0EHK, http://192.168.2.57:9876/things/v1/device/155402835379806208, method: DELETE

ARfWtLULA9, http://192.168.2.57:9876/things/v1/device/ui/list?type=Elevator&pageSize=20&pageNo=1&name=testap11, method: GET

```

分别记录, id, 接口url, method, log的顺序是执行回放的顺序

- 时间戳_回放列表
 - 如果全部通过，展示如下

A screenshot of a terminal window with a dark background. The title bar at the top shows three open files: '913_比对结果.txt', '1659422721_比对结果.txt', and '基坑'. The terminal content shows the text 'Test PASS!' in a light blue font, with a cursor positioned at the end of the line.

- 如果执行失败，则会展示对应的报错

```
{
  "id": "F3ZgR1gTM",
  "url": "http://192.168.2.57:9876/things/v1/device/point",
  "method": "POST",
  "contentList": [
    {
      "key": "data_id",
      "expect": [
        "155435645194138429"
      ]
    }
  ]
},
{
  "id": "oKLSbGEMG",
  "url": "http://192.168.2.57:9876/things/v1/device/point",
  "method": "POST",
  "contentList": [
    {
      "key": "msg",
      "expect": [
        "\u64cd\u4f5c\u6210\u529f"
      ],
      "actual": [
        "\u64cd\u4f5c\u6210\u529f"
      ]
    }
  ]
},
{
  "id": "3uGfOKoHmF",
  "url": "http://192.168.2.57:9876/things/v1/device/point/u1/list?deviceType=HighMould&pageSize=20&pageNo=1",
  "method": "GET",
  "contentList": [
    {
      "key": "data_id",
      "expect": [
        "155435645194138429"
      ]
    }
  ]
},
{
  "id": "Aw18bsmpen",
  "url": "http://192.168.2.57:9876/things/v1/device/point/u1/list?deviceType=HighMould&pageSize=20&pageNo=1",
  "method": "GET",
  "contentList": [
    {
      "key": "data_id",
      "expect": [
        "155435645194138429"
      ]
    }
  ]
},
{
  "id": "8FlsYs4Uq4",
  "url": "http://192.168.2.57:9876/things/v1/device/point/u1/list?deviceType=HighMould&pageSize=20&pageNo=1&name=point2",
  "method": "GET",
  "contentList": [
    {
      "key": "data_id",
      "expect": [
        "155435645194138429"
      ]
    }
  ]
},
{
  "id": "LfMhC3mhLh",
  "url": "http://192.168.2.57:9876/things/v1/device/point/u1/list?deviceType=HighMould&pageSize=20&pageNo=1&name=gdeviceId=1554352336619659264",
  "method": "GET",
  "contentList": [
    {
      "key": "data_id",
      "expect": [
        "155435645194138429"
      ]
    }
  ]
},
{
  "id": "nup9tszeDN",
  "url": "http://192.168.2.57:9876/things/v1/device/point/1554354451941384192",
  "method": "DELETE",
  "contentList": [
    {
      "key": "msg",
      "expect": [
        "\u64cd\u4f5c\u6210\u529f"
      ],
      "actual": [
        "\u64cd\u4f5c\u6210\u529f"
      ]
    }
  ]
}
```

每一条展示的那个id下的url, 以及具体不同的字段, 具体形式如下

```
{
  'id': 'F3ZgR1gqTM', #唯一id
  'url': 'http://192.168.2.57:9876/things/v1/device/point',
  'method': 'POST',
  'contentList': [{
    'key': 'data_id', #错误字段的key, data_id, 表示data下的id字段不一
    'expect': ['1554354451941384192'], #预期值
    'actual': ['1554354590844149760'] #实际值
  }]
}
```

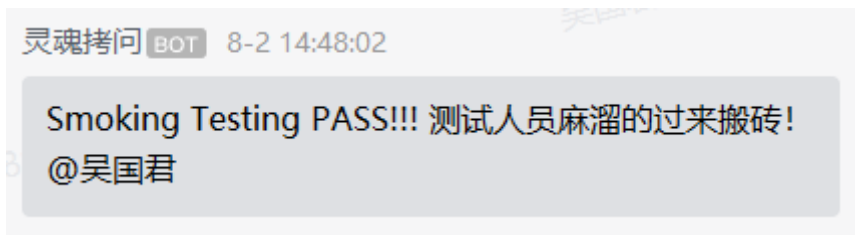
◦ ALLInfo_时间戳

- 本次所跑的所有脚本，记录本次所跑的脚本结果，和比对结果日志路径

```
{'Name': '停车-设备管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\停车-设备管理_测试脚本文件夹_1659337884\\1659422913_比对结果.txt'}
{'Name': '停车-车辆进出记录', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\停车-车辆进出记录_测试脚本文件夹_1658469158\\1659422913_比对结果.txt'}
{'Name': '党建-党员管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\党建-党员管理_测试脚本文件夹_1659344223\\1659422913_比对结果.txt'}
{'Name': '党建-党建新闻', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\党建-党建新闻_测试脚本文件夹_1659343617\\1659422913_比对结果.txt'}
{'Name': '劳务-人员管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-人员管理_测试脚本文件夹_1657879895\\1659422913_比对结果.txt'}
{'Name': '劳务-合同管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-合同管理_测试脚本文件夹_1658134794\\1659422913_比对结果.txt'}
{'Name': '劳务-工种管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-工种管理_测试脚本文件夹_1658134423\\1659422913_比对结果.txt'}
{'Name': '劳务-班组管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-班组管理_测试脚本文件夹_1658134268\\1659422913_比对结果.txt'}
{'Name': '劳务-规则配置', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-规则配置_测试脚本文件夹_1658298871\\1659422913_比对结果.txt'}
{'Name': '劳务-证书管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-证书管理_测试脚本文件夹_1658232267\\1659422913_比对结果.txt'}
{'Name': '劳务-进退场管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-进退场管理_测试脚本文件夹_1658134612\\1659422913_比对结果.txt'}
{'Name': '劳务-违纪管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-违纪管理_测试脚本文件夹_1658235890\\1659422913_比对结果.txt'}
{'Name': '劳务-黑名单管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\劳务-黑名单管理_测试脚本文件夹_1658235172\\1659422913_比对结果.txt'}
{'Name': '区域-区域管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\区域-区域管理_测试脚本文件夹_1659402014\\1659422913_比对结果.txt'}
{'Name': '升降机-传感器', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\升降机-传感器_测试脚本文件夹_1658814121\\1659422913_比对结果.txt'}
{'Name': '升降机-作业数据', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\升降机-作业数据_测试脚本文件夹_1658814050\\1659422913_比对结果.txt'}
{'Name': '升降机-台账列表', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\升降机-台账列表_测试脚本文件夹_1658814222\\1659422913_比对结果.txt'}
{'Name': '升降机-设备管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\升降机-设备管理_测试脚本文件夹_1659338141\\1659422913_比对结果.txt'}
{'Name': '卸料-传感器', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\卸料-传感器_测试脚本文件夹_1659417489\\1659422913_比对结果.txt'}
{'Name': '卸料-台账数据', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\卸料-台账数据_测试脚本文件夹_1659417486\\1659422913_比对结果.txt'}
{'Name': '卸料-设备管理', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\卸料-设备管理_测试脚本文件夹_1659417340\\1659422913_比对结果.txt'}
{'Name': '地磅-地磅台账', 'Status': 'Pass', 'location': 'D:\\git\\zhgd_recordjob\\zhgd_recordjob\\地磅-地磅台账_测试脚本文件夹_1659421386\\1659422913_比对结果.txt'}
```

脚本结果同时会通过webhook，发送到制定的企业微信群

- 脚本通过，展示如下



- 脚本失败，会展示报错的脚本，和提醒测试人员去查看问题，展示如下

灵魂拷问 BOT

```
{'Name': '塔吊-设备管理', 'Status': 'Failed'}
```

灵魂拷问 BOT

Smoking Testing Failed!!! 测试人员跑失败了，去瞅瞅！
@吴国君

CICD篇

本地调试完脚本后，上传至git

开发完成代码部署后，会执行测试脚本的docker服务

docker服务会先pull最新的git代码

执行代码，并将结果通过webhook发送到指定群

销毁docker容器，等待下次触发