



链滴

zabbix 发送报警信息带图片

作者: [cuijiazhe](#)

原文链接: <https://ld246.com/article/1595330548851>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

zabbix发送报警内容调用zabbix的api生成原始图数据，然后将图片上传到远方图床取出图片URL进行展示；

```
#!/bin/env python3
import sys
import requests
import json
import os
import time
import re

url = 'http://192.168.51.202/api_jsonrpc.php'
headers = {'Content-Type': 'application/json-rpc'}
graph_path='/data/zabbix/images/' #定义图片存储路径
graph_url='http://192.168.51.202/chart.php' #定义图表的url
loginurl="http://192.168.51.202/index.php" #定义登录的url
def get_itemid(message):
    itemid = re.search(r'ITEMID:(\d+)',message).group(1)
    return itemid
def get_imgUrl(itemid):
    session = requests.Session()
    try:
        loginheaders = {
            "Host": "192.168.51.202",
            "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/png,*/*;q=0.8"
        }
        # 定义请求消息头

        payload = {
            "name": 'cuijianzhe',
            "password": 'password',
            "autologin": "1",
            "enter": "Sign in",
        }
        # 定义传入的数据
        login = session.post(url=loginurl, headers=loginheaders, data=payload)
        graph_params = {
            "from": "now-10m",
            "to": "now",
            "itemids": itemid,
            "width": "400",
        }
        # 定义获取图片的参数
        graph_req = session.get(url=graph_url, params=graph_params)
        # 发送get请求获取图片数据
        time_tag = time.strftime("%Y%m%d%H%M%S", time.localtime())
        graph_name = 'baojing_' + time_tag + '.png'
        # 用报警时间来作为图片名进行保存
        graph_name = os.path.join(graph_path, graph_name)
        # 使用绝对路径保存图片
        with open(graph_name, 'wb') as f:
            f.write(graph_req.content)
        # 将获取到的图片数据写入到文件中去
    except Exception as e:
        print(e)
```

```

    return graph_name
except Exception as e:
    print(e)
    return False

def uploadImg(imgpath):
    url_token = "https://tc.ltyuanfang.cn/api/token"
    payload = {
        "email": "myuser",
        "password": "mypassword",
    }
    login = requests.post(url=url_token, data=payload).text
    token = json.loads(login)['data']['token']
    url = "https://tc.ltyuanfang.cn/api/upload"
    headers = {'token': token}
    with open(imgpath,'rb') as file:
        files = {'image':file}
        upimg = requests.post(url=url, files=files,headers=headers)
    return upimg.json()['data']['url']
def send_msg(messages,imgurl):
    headers = {'Content-Type': 'application/json; charset=utf-8'}
    api_url = "https://oapi.dingtalk.com/robot/send?access_token=mytoken"
    json_text = {
        "msgtype": "markdown",
        "markdown": {
            "title": "网络报警通知",
            "text": messages + "![screenshot](%s)\n%imgurl"
        },
        "at": {
            "atMobiles": [
                ""
            ],
            "isAtAll": False
        }
    }
    return requests.post(api_url, json.dumps(json_text), headers=headers).content

if __name__ == '__main__':
    message = sys.argv[1]
    itemid = get_itemid(message)
    imgpath= get_imgUrl(itemid)
    imgurl = uploadImg(imgpath)
    send_msg(message,imgurl)

```

发送参数:

故障问题: {EVENT.NAME}

```

<font color="#dd0000 size=3 >故障问题: {TRIGGER.NAME}
</font>
> 网络设备端报警内容如下, 请及时处理!
> - ITEMID:{ITEM.ID}
> - 设备IP地址: {HOST.IP}
> - 当前时间: {DATE} {TIME}

```

```
> - 故障 ID: {EVENT.ID}  
{TRIGGER.URL}  
> - 故障设备: {HOST.NAME}  
>  
> - 状态: **{ITEM.LASTVALUE}**  
#### -
```

恢复问题: {EVENT.NAME}

```
<font color="#0000dd size=3>恢复问题: {TRIGGER.NAME}</font>  
> 网络设备端报警已恢复!  
> - ITEMID:{ITEM.ID}  
> - 设备IP地址: {HOST.IP}  
> - 恢复时间: {EVENT.RECOVERY.DATE} {EVENT.RECOVERY.TIME}  
> - 持续时长: {EVENT.AGE} minute  
> - 故障 ID: {EVENT.ID}  
> - 故障级别: {TRIGGER.SEVERITY}  
{TRIGGER.URL}  
> - 故障设备: {HOST.NAME}  
>  
> - 状态: **{ITEM.LASTVALUE}**  
#### -
```

效果如下:

