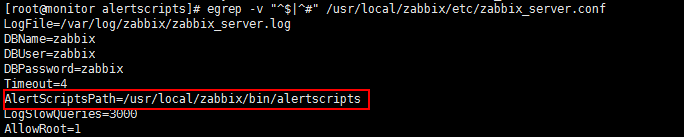
**Zabbix Python 邮件报警**

zabbix系统邮件告警配置有很多方式，如系统自带的mail，还有zabbix+msmtp+mutt，但是使用Python脚本应该来说还是一个非常简捷有效的方式。主要有如下步骤；

**1、设置zabbix\_server.conf文件**

AlertScriptsPath=/usr/local/zabbix/bin/alertscripts



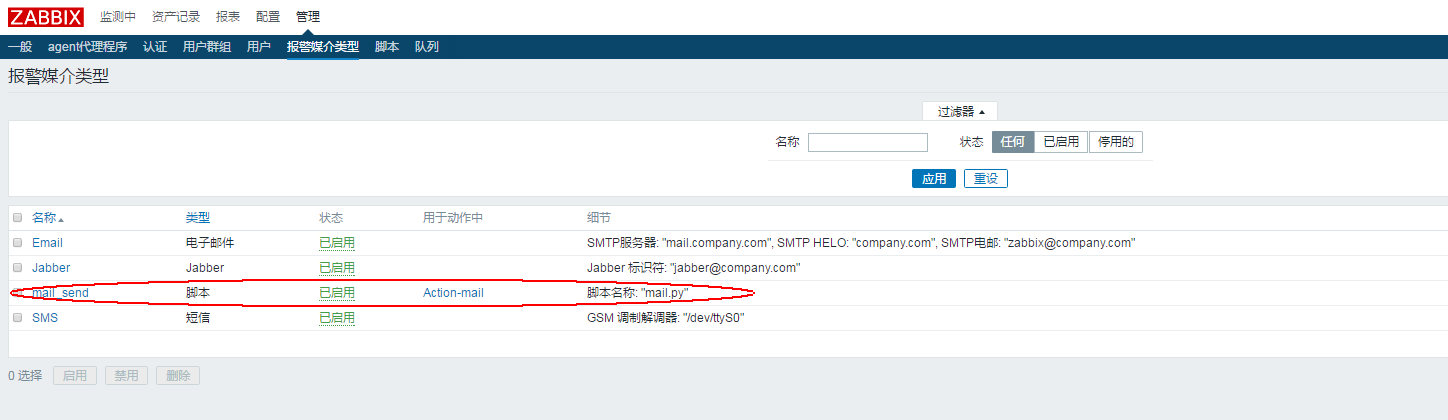
**2、建立告警脚本**

|  |
| --- |
| cd /usr/local/zabbix/bin/alertscripts  vim mail.py |

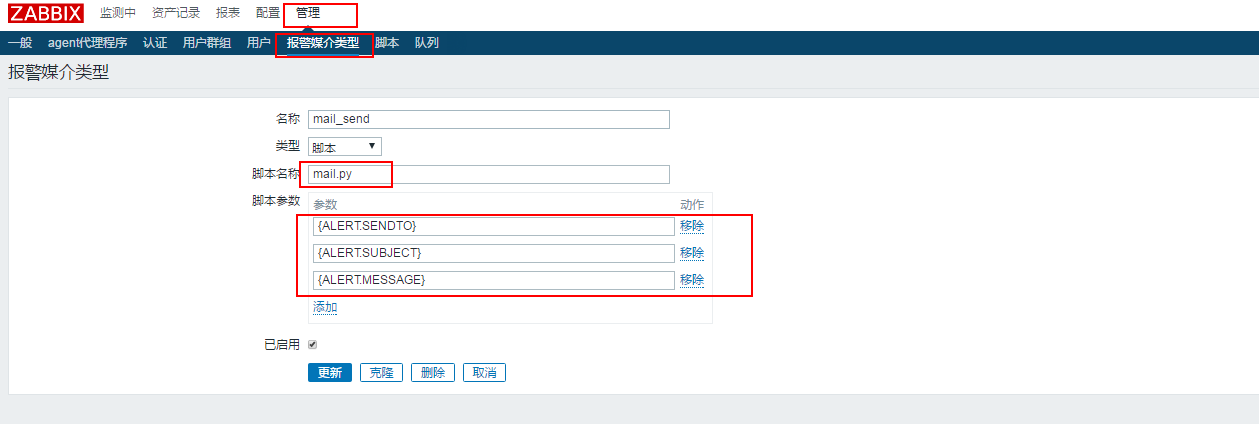
|  |
| --- |
| #!/usr/bin/python  #coding:utf-8  #author: itnihao  #mail: itnihao@qq.com  #url: https://github.com/itnihao/zabbix-book/edit/master/06-chapter/zabbix\_sendmail\_v2.py    import smtplib  from email.mime.text import MIMEText  import os  import argparse  import logging  import datetime    #QQ enterprise  #smtp\_server = 'smtp.exmail.qq.com'  #smtp\_port = 25  #smtp\_user = 'itnihao\_zabbix@itnihao.com'  #smtp\_pass = '1234567890'  #163 Mail  #smtp\_server = 'smtp.163.com'  #smtp\_port = 25  #smtp\_user = 'itnihao\_zabbix@163.com'  #smtp\_pass = '1234567890'  #GOOGLE Mail  smtp\_server ='smtp.gmail.com'  smtp\_port = 465  smtp\_user ='john@orientalgame99.com'  smtp\_pass ='zjh8108echo$?'  def send\_mail(mail\_to,subject,content):  msg = MIMEText(content,\_subtype='plain', \_charset='utf-8')  msg['Subject'] = unicode(subject,'UTF-8')  msg['From'] = smtp\_user  msg['to'] = mail\_to  global sendstatus  global senderr    try:  if smtp\_port == 465:  smtp = smtplib.SMTP\_SSL()  else:  smtp = smtplib.SMTP()  smtp.connect(smtp\_server,smtp\_port)  smtp.login(smtp\_user,smtp\_pass)  smtp.sendmail(smtp\_user,mail\_to,msg.as\_string())  smtp.close()  print 'send ok'  sendstatus = True  except Exception,e:  senderr=str(e)  print senderr  sendstatus = False    def logwrite(sendstatus,mail\_to,content):  logpath='/var/log/zabbix/alert'  if not sendstatus:  content = senderr  if not os.path.isdir(logpath):  os.makedirs(logpath)  t=datetime.datetime.now()  daytime=t.strftime('%Y-%m-%d')  daylogfile=logpath+'/'+str(daytime)+'.log'  logging.basicConfig(filename=daylogfile,level=logging.DEBUG)  os.system('chown zabbix.zabbix {0}'.format(daylogfile))  logging.info('\*'\*130)  logging.debug(str(t)+' mail send to {0},content is :\n {1}'.format(mail\_to,content))  if \_\_name\_\_ == "\_\_main\_\_":  parser = argparse.ArgumentParser(description='Send mail to user for zabbix alerting')  parser.add\_argument('mail\_to',action="store", help='The address of the E-mail that send to user ')  parser.add\_argument('subject',action="store", help='The subject of the E-mail')  parser.add\_argument('content',action="store", help='The content of the E-mail')  args = parser.parse\_args()  mail\_to=args.mail\_to  subject=args.subject  content=args.content  send\_mail(mail\_to,subject,content)  logwrite(sendstatus,mail\_to,content) |
| chown 755 mail.py |

**3、媒介配置**

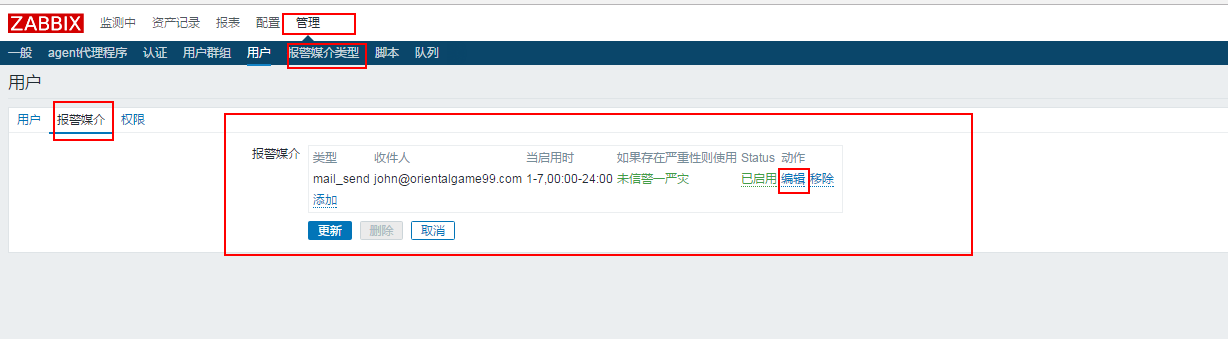
3.1 添加媒介

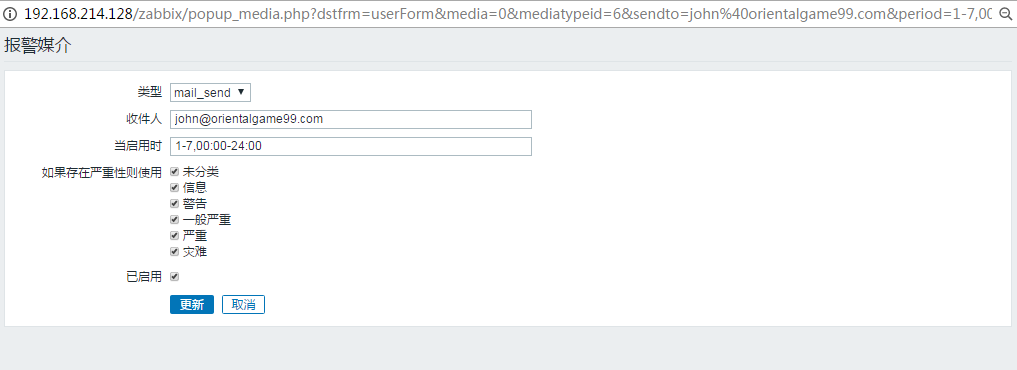


3.2设置媒介的名称和脚本的路径，已经需要传递的宏

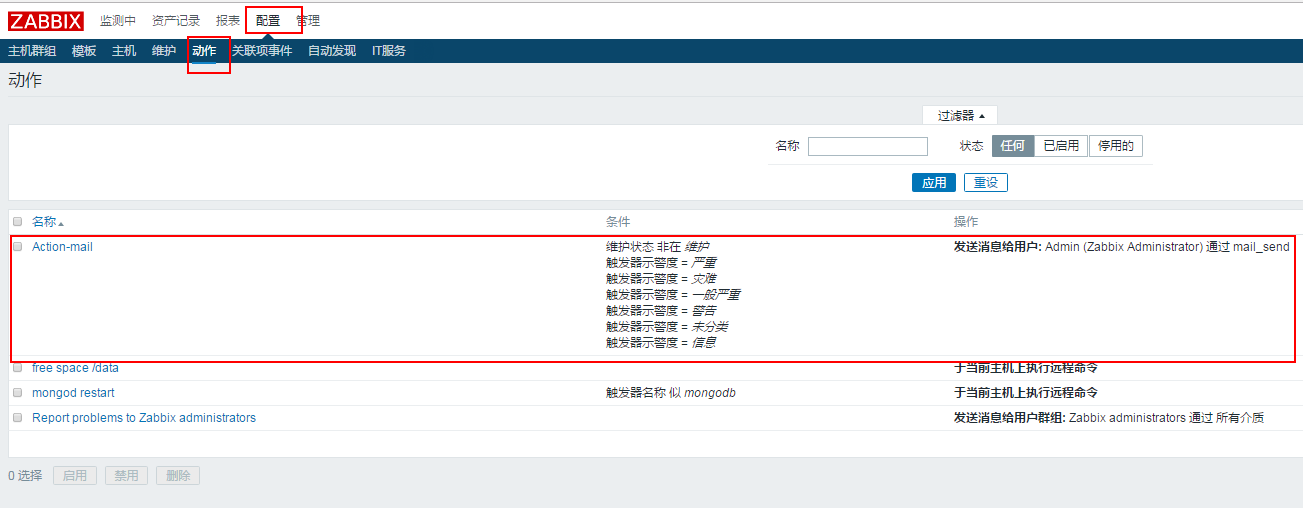


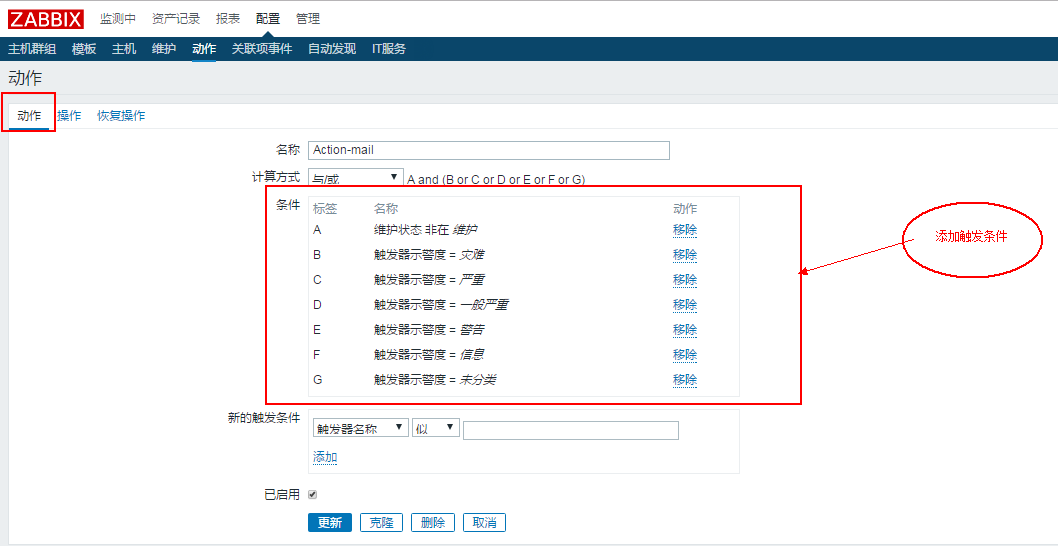
3.3 给用户添加我们新建的媒介

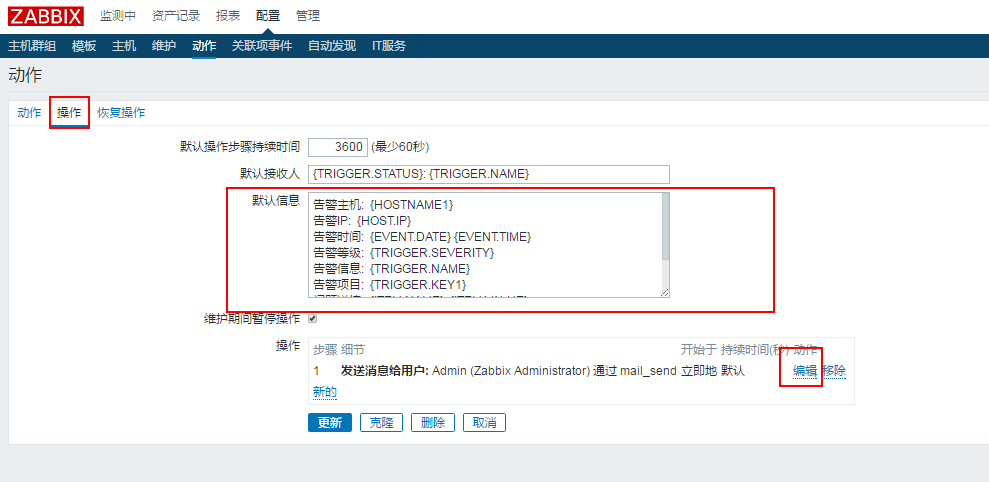




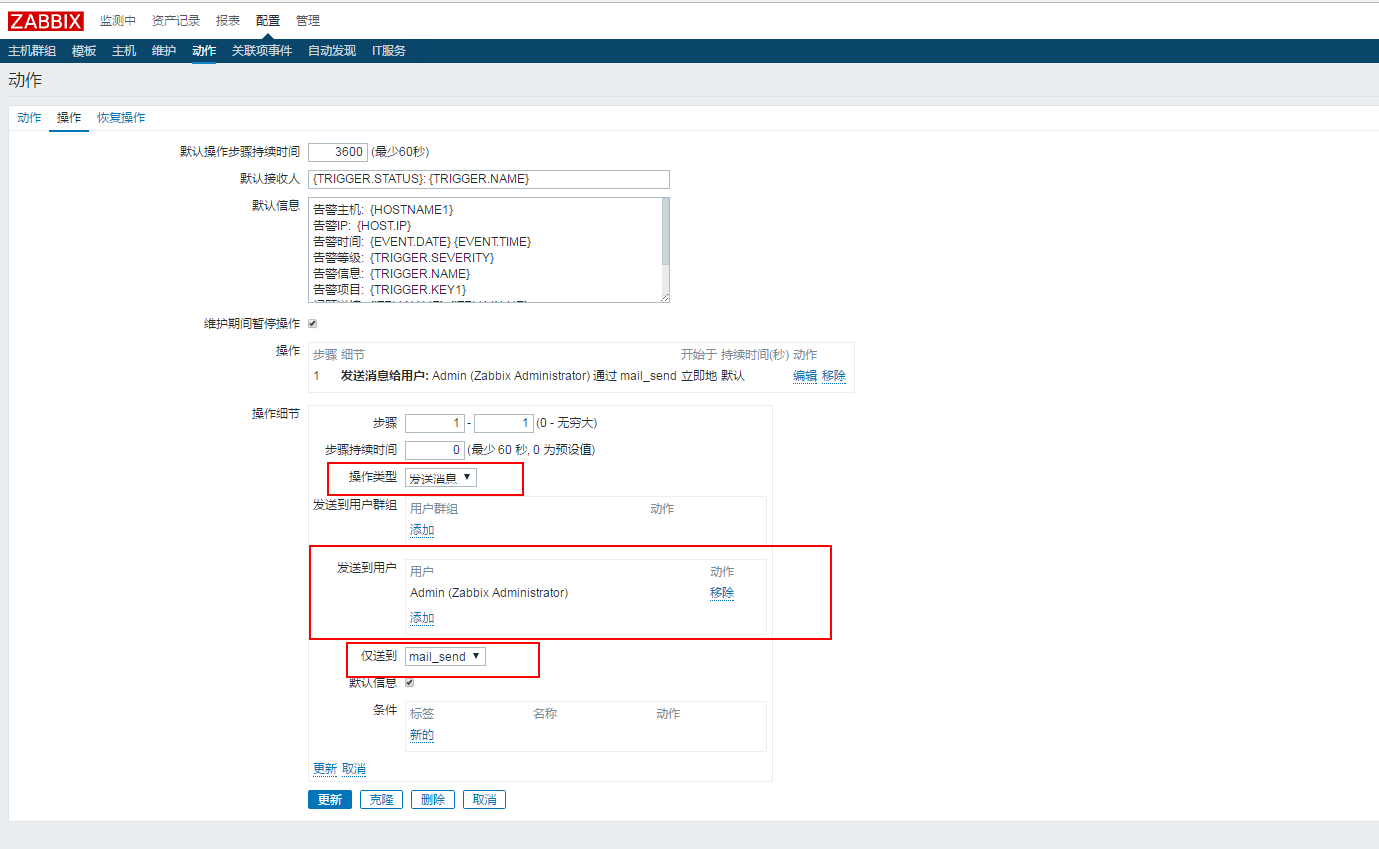
3.4 在动作的操作里面也添加选择我们新建的媒介（可以设置成只发送消息设置也可以带有远程命令，添加两个，就是当发生报警的时候远程命令帮我们执行命令）







|  |
| --- |
| 告警主机: {HOSTNAME1}  告警IP: {HOST.IP}  告警时间: {EVENT.DATE} {EVENT.TIME}  告警等级: {TRIGGER.SEVERITY}  告警信息: {TRIGGER.NAME}  告警项目: {TRIGGER.KEY1}  问题详情: {ITEM.NAME}: {ITEM.VALUE}  当前状态: {TRIGGER.STATUS}: {ITEM.VALUE1}  事件ID: {EVENT.ID} |



**解释**

默认的步骤是1-1,也即是从1开始到1结束。一旦故障发生，就是执行mail.py脚本发生报警邮件给Admin用户和zabbix administrator组。   
假如故障持续了1个小时，它也只发送一次。如果改成1-0，0是表示不限制.无限发送   
间隔就是默认持续时间60秒。那么一个小时，就会发送60封邮件。   
如果需要短信报警的话,可以再创建一条新的动作，选择短信脚本。

3.5 点击更新，等待邮件的到来

**4 附件**



**5 参考网址**

|  |
| --- |
| **<http://www.cnblogs.com/shenjianyu/p/6599364.html>**  **http://www.cnblogs.com/kevingrace/p/5961861.html** |