# **Linux FTP服务和邮件发送**

# **[Linux下实现 OpenSSL 简单加密与解密字符串](https://www.cnblogs.com/linuxprobe-sarah/p/10129520.html)**

**场景**

shell脚本中存在明文密码  
客户要求禁止使用明文密码,密码做加密处理.

**方案**

[Linux](https://www.linuxprobe.com/" \o "linux" \t "https://www.cnblogs.com/linuxprobe-sarah/p/_blank) OpenSSL加密解密工具  
可以指定各种加密算法为字符,文件做加密处理.  
加密的案例比较多,解密的寥寥无几.  
有兴趣的可以去查下中文教程  
案例中使用加密算法 : AES

**实例**

##所以命令都在Linux中执行##

##查看是否安装了该工具,我的环境自带了,文章不做安装描述##

openssl

##加密(执行后获得加密码)##

echo abc | openssl aes-128-cbc -k 123 -base64

##解密(加密码太长,用\*表示了)##

echo \*\*\*\*\* | openssl aes-128-cbc -d -k 123 -base64

### **FTP的工作模式**

****主动模式：**** FTP 服务器主动向客户端发起连接请求。

****被动模式：**** FTP 服务器等待客户端发起连接请求（FTP 的默认工作模式）。

FTP: 　文件传输协议

软件包: 　vsftpd

FTP端口: 　控制端口21/tcp;数据端口20/tcp(主动模式)

配置文件：　/etc/vsftpd/vsftpd.conf

sed -ri '/^SELINUX=/cSELINUX=disabled' /etc/selinux/config #关闭SELinux，c表示整行替换

sed -rn '7c SELINUX=LIUZHAOCHEN2' config

sed -ri 's/^(SELINUX=).\*/\1disabled/' config

vsftpd 作为更加安全的文件传输的服务程序，允许用户以三种认证模式登录到 FTP 服务器上。

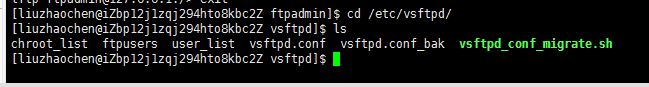
匿名开放模式： 是一种最不安全的认证模式，任何人都可以无需密码验证而直接登录到 FTP 服务器。

本地用户模式： 是通过 Linux 系统本地的账户密码信息进行认证的模式，相较于匿名开放模式更安全，而且配置起来也很简单。

虚拟用户模式： 是这三种模式中最安全的一种认证模式，它需要为 FTP 服务单独建立用户数据库文件，虚拟出用来进行口令验证的账户信息，而这些账户信息在服务器系统中实际上是不存在的，仅供 FTP 服务程序进行认证使用。

1. FTP部署，ftp服务的软件名vsftpd,服务名：vsftpd.service

配置文件：/etc/vsftpd/vsftpd.conf



安装软件：yum -y install vsftpd

启动服务：systemctl start vsftpd.service

系统显示如下信息：Created symlink from /etc/systemd/system/multi-user.target.wants/vsftpd.service to /usr/lib/systemd/system/vsftpd.service. 让这个程序自己的启动配置创建一个链接到系统的启动配置文件下，系统开机的时候就能运行 /usr/lib/systemd/system/ 文件下的配置文件对应的程序。

开机自启：Systemctl enable vsftpd.service

若不修改配置文件，默认路径：/var/ftp

**配置文件 /etc/vsftpd/vsftpd.conf**

anonymous\_enable=NO # 不允许匿名访问，禁用匿名登录,YES允许

local\_enable=YES # 允许使用本地帐户进行FTP用户登录验证

local\_umask=022 # 上传的文件的权限设置为：777-local\_umask

write\_enable=YES #允许写入，如需上传，则必须

download\_enable＝[YES|NO] #是否允许下载文件

userlist\_enable=[YES|NO] # 是否启用用户列表，YES为启用user\_list文件

userlist\_deny=[YES|NO] #为NO时，只允许user\_list文件内的用户名登录ftp，当为YES时，该文件内的用户不让登录ftp

#/etc/vsftpd/user\_list（白名单，只有里面的名单可以登录）

#/etc/vsftpd/ftpusers（黑名单，里面的名单不允许登录）

注：首先请明确一点：ftpusers不受任何配制项的影响，它总是有效，它是一个黑名单！

该文件存放的是一个禁止访问FTP的用户列表，通常为了安全考虑，管理员不希望一些拥有过大权限的帐号（比如root)登入FTP，以免通过该帐号从FTP上传或下载一些危险位置上的文件从而对系统造成损坏。

ftpusers和user\_list没有任何关系，ftpusers文件总是生效，user\_list则取决于userlist\_enable和userlist\_deny两项配置。

userlist\_enable和userlist\_deny两个选项联合起来针对的是：本地全体用户（除去ftpusers中的用户）和出现在user\_list文件中的用户以及不在在user\_list文件中的用户这三类用户集合进行的设置。

当且仅当userlist\_enable=YES时：userlist\_deny项的配置才有效，user\_list文件才会被使用；当其为NO时，无论userlist\_deny项为何值都是无效的，本地全体用户（除去ftpusers中的用户）都可以登入FTP

当userlist\_enable=YES时，userlist\_deny=YES时：user\_list是一个黑名单，即：所有出现在名单中的用户都会被拒绝登入；

当userlist\_enable=YES时，userlist\_deny=NO时：user\_list是一个白名单，即：只有出现在名单中的用户才会被准许登入(user\_list之外的用户都被拒绝登入)；另外需要特别提醒的是：使用白名单后，匿名用户将无法登入！除非显式在user\_list中加入一行：anonymous

local\_root=/var/ftp #本地用户的 FTP 根目录  
# FTP上本地的文件权限，默认是077，不过vsftpd安装后的配置文件里默认是022

max\_clients=0 #最大客户端连接数， 0 为不限制

max\_per\_ip=0 #同一 IP 地址的最大连接数， 0 为不限制

anon\_upload\_enable=YES # 允许虚拟用户和匿名用户上传

anon\_other\_write\_enable=YES #允许虚拟用户和匿名用户修改文件名和删除文件

xferlog\_enable=YES # 启用上传和下载的日志功能，默认开启。

connect\_from\_port\_20=YES

xferlog\_file=/var/log/vsftpd.log #日志存放位置

idle\_session\_timeout=600 #空闲连接超时

ftpd\_banner=Welcome to Holten FTP service #欢迎信息

guest\_enable=yes #允许虚拟用户

guest\_username=vsftpdguest #虚拟用户使用的系统账号virtual\_use\_local\_privs=YES #虚拟用户拥有本地系统权限

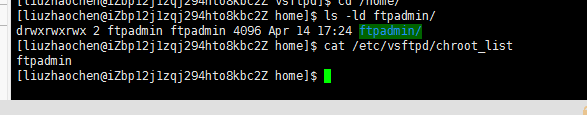
use\_localtime=YES # 使用本地时(自行添加)  
chroot\_local\_user=NO # 启用限定用户在其主目录下  
chroot\_list\_enable=YES #通常为YES、NO，（所有都限制在主目录）

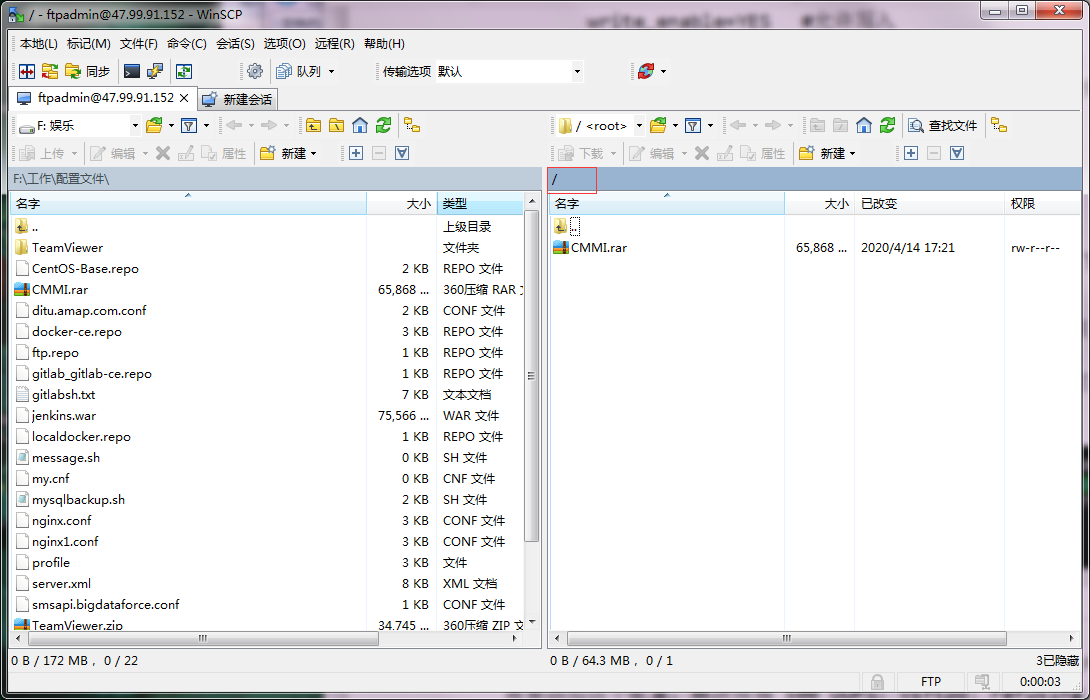
#NO、YES，只有chroot\_list\_file的限制在主目录下



#以上两行将虚拟用户限制在其目录下，不能访问其他目录，或者直接用

chroot\_list\_file=/etc/vsftpd/chroot\_list #该文件要自己创建，可以在里面填写ftp登录账号，文件里的账号登录ftp时，根目录就是家目录（/home/ftpadmin），而且切换不到其它目录下。若文件外的账号登录ftp时，可以切换到其它目录下，不安全。





　　　　allow\_writeable\_chroot=YES # 如果启用了限定用户在其主目录下需要添加这个配置，解决报错 500 OOPS: vsftpd: refusing to run with writable root inside chroot()

listen=yes #是否以独立运行的方式监听服务监听/被动模式

listen\_address=IP 地址 #设置要监听的 IP 地址

listen\_port=21 #设置ftp监听端口21

　　　　#pasv\_enable=no #禁用被动模式，默认启动

pasv\_min\_port=40000 #被动模式最小端口号

pasv\_max\_port=40080 #被动模式最大端口号

209服务器上的/etc/vsftpd/vsftpd.conf

12 anonymous\_enable=NO

16 local\_enable=YES

19 write\_enable=YES

23 local\_umask=022

37 dirmessage\_enable=YES

40 xferlog\_enable=YES

43 connect\_from\_port\_20=YES

57 xferlog\_std\_format=YES

83 ascii\_upload\_enable=YES

84 ascii\_download\_enable=YES

101 chroot\_local\_user=YES

102 chroot\_list\_enable=NO （限制在自己家目录，若为YES、NO，所有都限制，NO、YES只限制chroot\_list文件里的用户）

104 chroot\_list\_file=/etc/vsftpd/chroot\_list （自己创建并填写）

105 allow\_writeable\_chroot=YES

116 listen=NO

125 listen\_ipv6=YES

127 pam\_service\_name=vsftpd

128 userlist\_enable=YES

129 userlist\_enable=NO （YES、NO，代表user\_list的是白名单，且只有user\_list里的名单能登录ftp）

130 tcp\_wrappers=YES

有时候重启vsftpd时，报错：500 OOPS: bad bool value in config file for: allow\_writeable\_chroot

可能是配置项后面多加了“ ”空格，要观察一下。

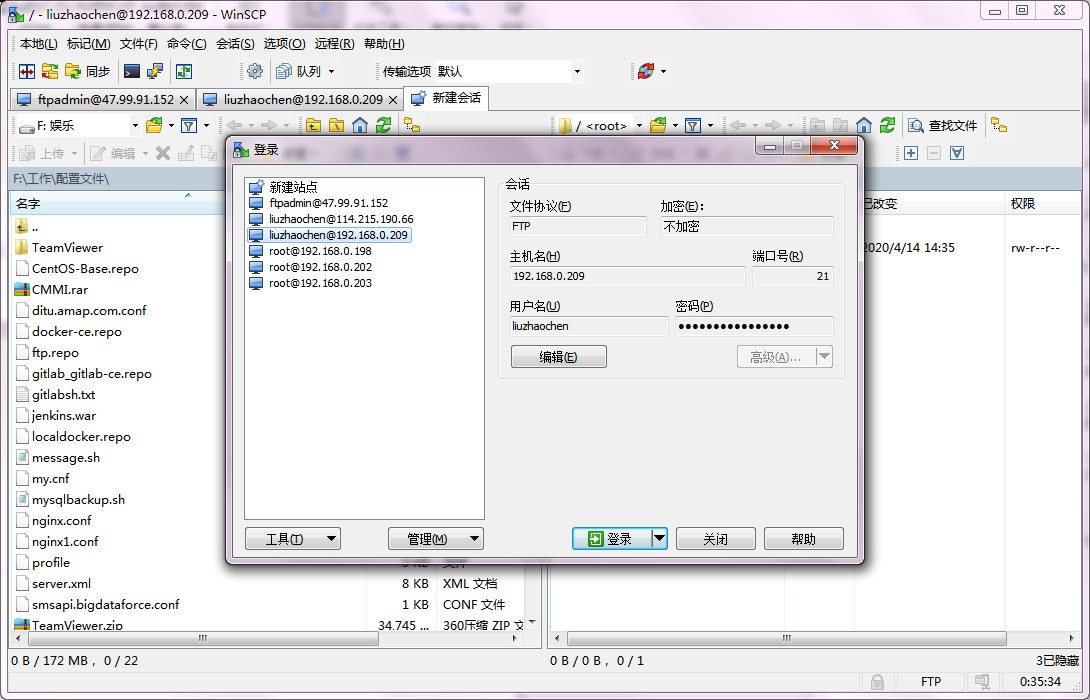
本地创建一个不能登录的账户：

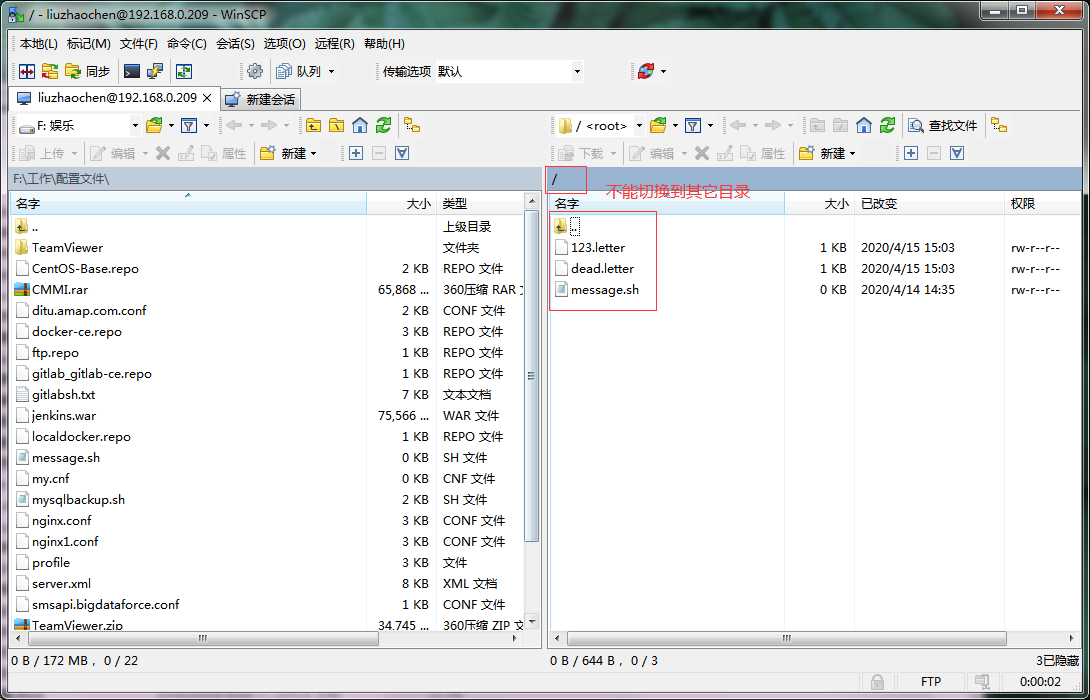
useradd -s /sbin/nologin liuzhaochen

passwd liuzhaochen （设置密码）

连接测试：

1、用WinScp连接





2、用lftp连接，要安装lftp软件：yum -y install lftp

配置文件： /etc/lftp.conf

连接命令：lftp [liuzhaochen@127.0.0.1](mailto:liuzhaochen@127.0.0.1) 输入密码即可

lftp -[fceup] user:passwd@host #passwd可选,此方式密码明文保存,history可查看

lftp 用户名:密码@ftp地址:传送端口（默认21）#在服务端未进行一些限制设置的时候可以直接通过ip进行登录访问

-f <file>           #执行文件中的命令后退出

-c <cmd>        #执行命令后退出

-e <cmd>         #登录后后执行命令

-u <user>[,<pass>]  #使用指定的用户名/口令进行验证

-p <port>           #连接指定的端口，默认21

cd #切换FTP目录

lcd #切换本地目录

ls  #列出FTP目录文件

!ls #列出本地目录文件

find ./ -name file #查找文件

mv #移动

rm #删除

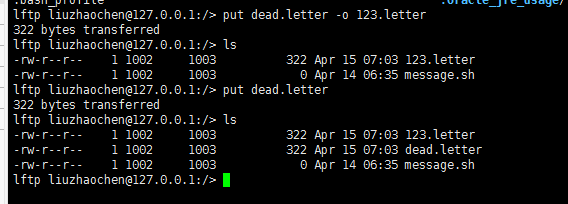
mkdir #新建目录

rmdir #删除目录

上传下载文件：

get [cEaO] file -o newfilename #下载文件

put [cEaO] file -o newfilename #上传文件，通常lcd切换到本地的目录，再上传下载，加了-o选项，相当于改名



pget -n #多线程下载

可选参数：

-c #支持断点续传

-E #下载后删除远端文件

-a #指定传输为ascii模式, 默认为二进制模式

-O #指定存放目录

上传下载目录

mirror [cRnr] remote [local]

可选参数：

-c #支持断点续传

-R #上传整个目录

-n #只下载较新的档案，可用于增量备份

-r  #只下载当前目录，不递归目录

妙用lftp多线程下载：

|  |  |
| --- | --- |
| 1 | lftp -c "pget -n 10 http://www.kernel.org/pub/linux/kernel/v2.6/linux-2.6.18.tar.gz" |

#### Linux发送外网邮箱

安装邮件相关软件

# yum -y install mailx

# rpm -qa|grep mail

# yum -y install sendmail

# yum -y install mutt

启动服务：systemctl restart sendmail

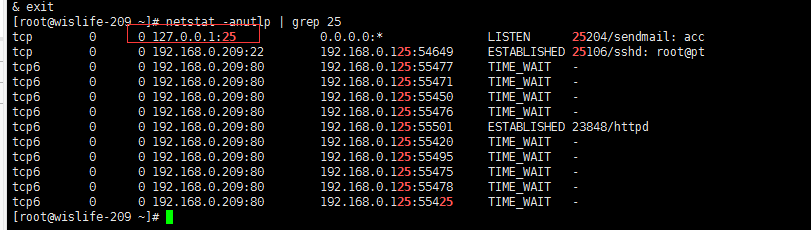
如果起不来，就需要停到postfix服务，sendmail与postfix只能用一个，不然冲突。

systemctl stop postfix

systemctl start sendmail

系统邮件信息路径：/var/mail /var/spool/mail

使用#ss   -tnl查看25端口是否监听在127.0.0.1上。默认就是监听的。



邮件发送命令：

mail -s ‘mail test’ [acb@qq.com,def@qq.com](mailto:acb@qq.com,def@qq.com) < mail.txt

echo “mail comment” | mail -s ‘mail test’ [abc@qq.com](mailto:abc@qq.com)

发送附件  
   uuencode 附件名 显示附件名 | mailx  -s "subject" root@localhost

注：uuencode是将二进制文件转换为文本文件的过程，转换后的文件可以通过纯文本e-mailx 进行传输，在接收方对该文件进行uudecode，即将其转换为初始的二进制文件。

设置外部SMTP来发送邮件：

vim /etc/mail.rc

添加：

set from=busername@sina.com

set smtp=smtp.sina.com

set smtp-auth-user=username 登录邮箱的名字

set smtp-auth-password=passwd 登录邮箱的密码或是SMTP的秘钥

set smtp-auth=login

set nss-config-dir=/root/.certs

set ssl-verify=ignore

from：发送的邮件地址，对方显示的发件人(对方收到邮件时显示的发件人)  
smtp：发生的第三方发邮件的smtp[服务器](https://www.funet8.com/tag/server" \o "查看\“服务器\”所有文章" \t "https://www.funet8.com/_blank)地址  
smtp-auth-user：第三方发邮件的smtp[服务器](https://www.funet8.com/tag/server" \o "查看\“服务器\”所有文章" \t "https://www.funet8.com/_blank)认证的用户名  
smtp-auth-password：邮箱授权码  
smtp-auth：邮件认证的方式,默认是login，也可以改成CRAM-MD5或PLAIN方式

nss-config-dir: SSL验证信息存放位置，需要后面步骤创建  
ssl-verify: SSL验证信息忽略

qq邮箱开启SMTP功能，并获取授权码



重启sendmail

systemctl restart sendmail

实际209服务器/etc/mail.rc配置：

set from=739136979@qq.com

set smtp=smtp.qq.com

set smtp-auth-user=739136979@qq.com

set smtp-auth-password=yonafivocvgbbaib

set smtp-auth=login

163的smtp邮件服务器：

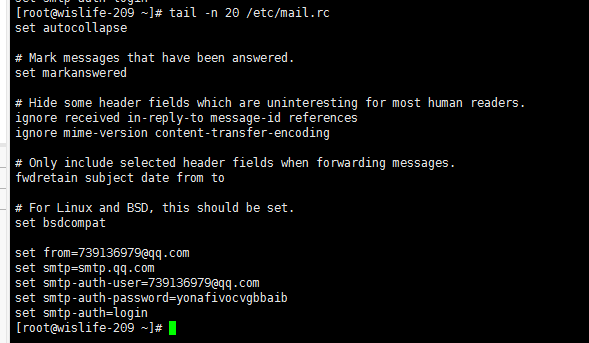
set from=xxx@163.com  这里和下面的smtp-auth-user一致

smtp=smtp.163.com           建议用  163的邮箱发，qq邮箱有点问题

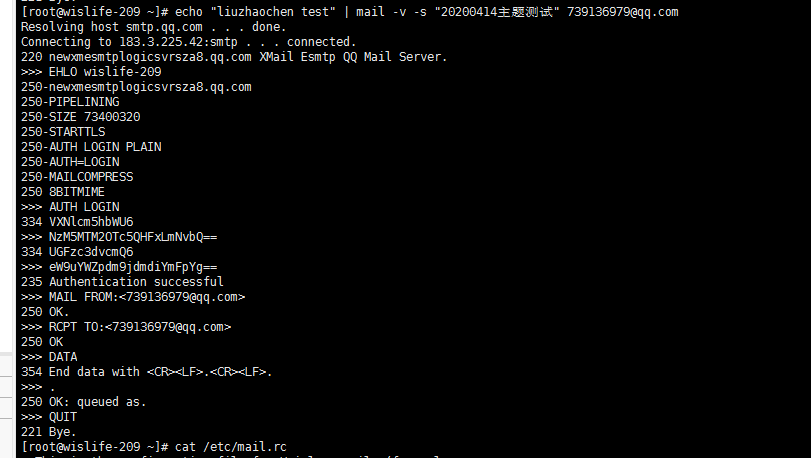
set smtp-auth-user=xxx@163.com         （企业和个人邮箱的服务器地址不同，不要搞混了）

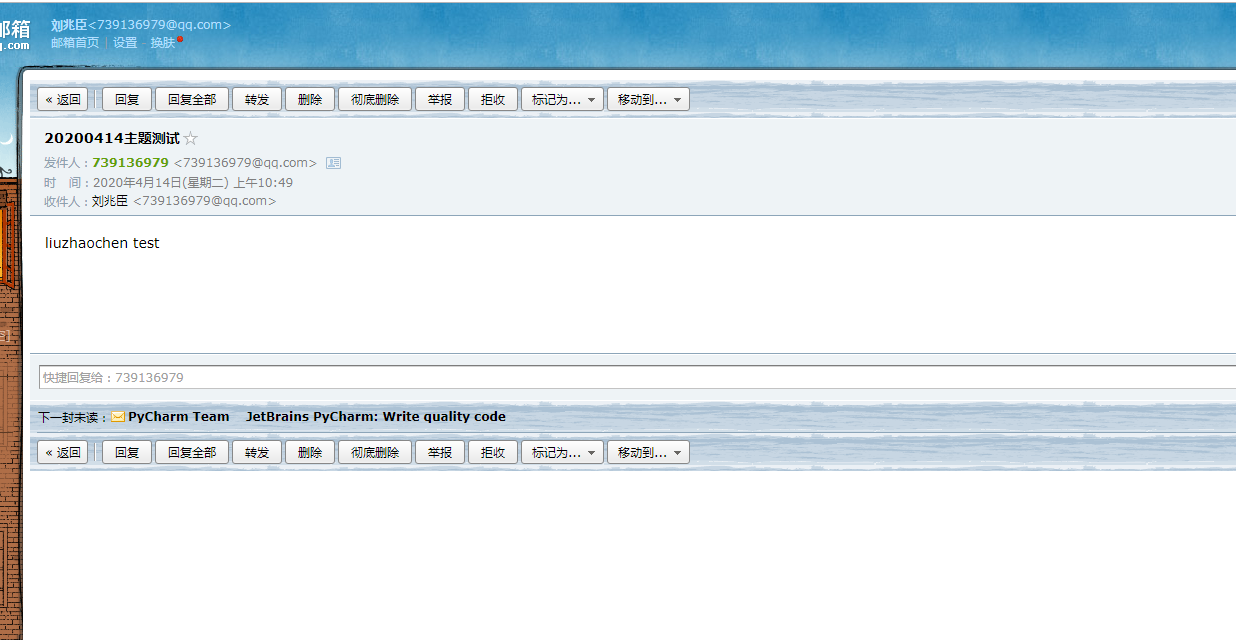
smtp-auth-password=你的密码（现在很多邮箱要授权码-----看看是否需要填写授权码来代替密码）163的授权码是自己设置的(自己百度下什么生成授权码)

set smtp-auth=login



测试：





#### Zabbix配置邮箱报警

前提：Zabbix监控服务器、客户端都已经部署完成，被监控主机已添加完成，Zabbix监控运行正常，上面的邮件能发送到外网。

**目的：**

Zabbix监控服务器设置邮件报警，当被监控主机宕机或达到触发器预设值进，会自动发送报警邮件到指定的邮箱。

**概念补充：**

什么是MTA？什么是MUA？

通俗的说MUA就是用户使用的客户端，而MTA则是真正发送邮件的发送进程，MUA编写的邮件最终也交由MTA发送至目标端。

mailx即为负责查看、编写邮件和向MTA发送邮件的MUA。

Sendmail和postfix即为负责邮件在网络上传输的MTA，将邮件从一个MTA传送至另一个MTA。

**标注：**使用zabbox服服务端本地邮箱账号发送邮件，需要安装两个软件达到报警邮件转发至指定邮箱。Sendmail或postfix（两个软件各选择一个，但不能同时运行）和mailx邮件客户端。

**Zabbix后台web介面配置**步骤：

编写邮件发送脚本sendmail.sh，209服务器上

[root@ssticentos65 ~]# **cd /usr/lib/zabbix/alertscripts**     #进入zabbix默认存放脚本路径

[root@ssticentos65 alertscripts]# **vim sendmail.sh**                                             #编写发送邮件脚本

**#!/bin/bash**

**messages=`echo $3 | tr '\r\n' '\n'`**

**subject=`echo $2 | tr '\r\n' '\n'`**

**echo "${messages}" | mail -s "${subject}" $1 >>/tmp/sendmail.log 2>&1**

[root@ssticentos65 ~]# **chown zabbix.zabbix /usr/lib/zabbix/alertscriptssendmail.sh**                       #sendmail.sh脚本修改所属用户和用户组

[root@ssticentos65 ~]# **chmod 755 /usr/lib/zabbix/alertscriptssendmail.sh**                                     #sendmail脚本赋予执行权限

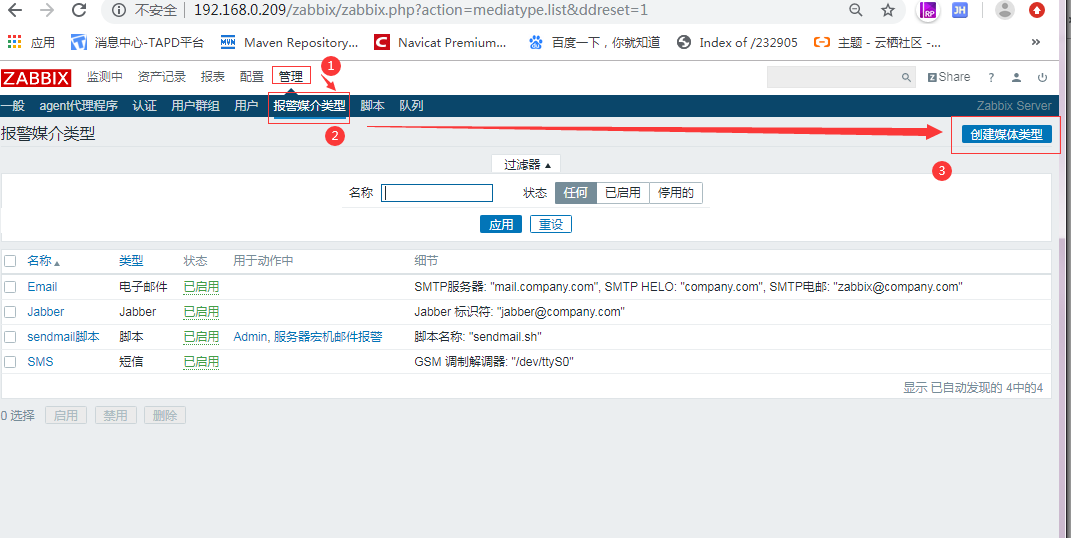
[root@ssticentos65 ~]#**ls -l /usr/lib/zabbix/alertscriptssendmail.sh**                                                 #查看sendmail.sh脚本信息

-rwxr-xr-x. 1 zabbix zabbix 154 Jan  7 05:12 /usr/lib/zabbix/alertscripts/sendmail.sh

[root@ssticentos65 ~]# **vim /etc/zabbix/zabbix\_server.conf**                                                                          #配置zabbix\_server.conf配置文件

**AlertScriptsPath=/usr/lib/zabbix/alertscripts**                                                                                     #这个修改成sendmail.sh脚本存放路径

1. 管理 ----  报警媒介类型 -----  创建媒介类型。



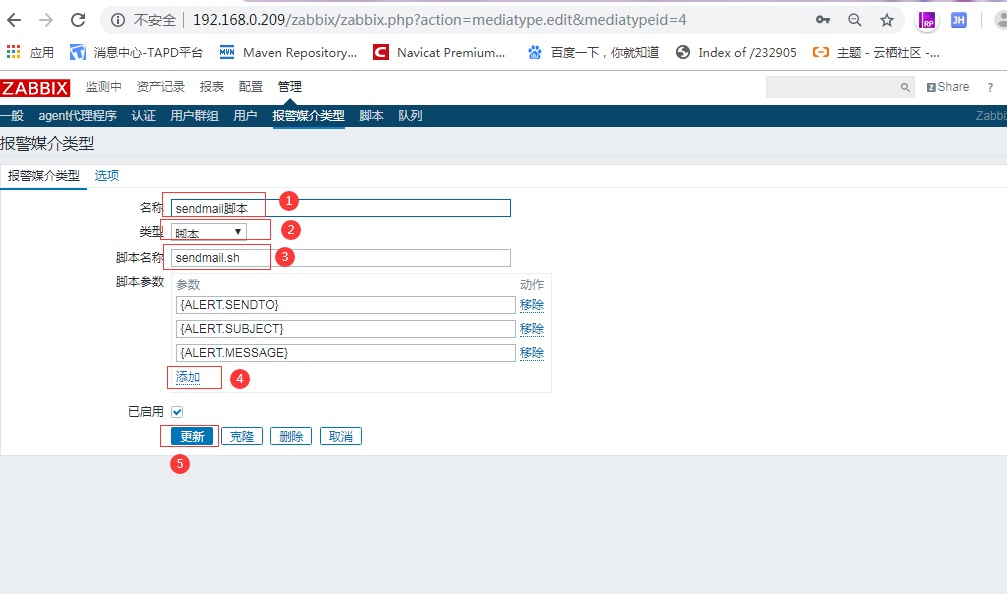
2、名称可以随便自己定义 ----- 类型选择脚本 ----- 脚本名称输入上一步编写的发送邮件的脚本名称 ----

再添加以下3个参数，分别对应脚本需要的3个参数：收件人地址、主题、详细内容：

{ALERT.SENDTO}

{ALERT.SUBJECT}

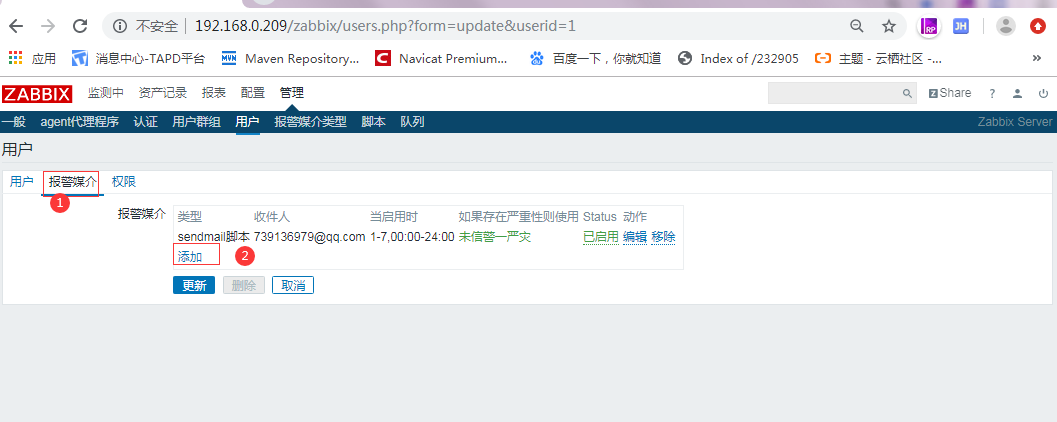
{ALERT.MESSAGE}



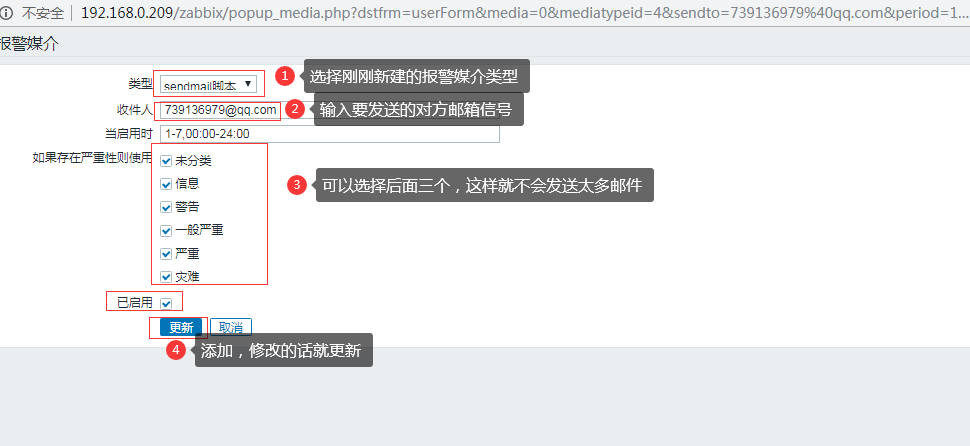
3、管理 ---- 用户 ----- 选择admin用户，点击进去。(自己创建一个用户也可以)



1. 报警媒介 ---- 添加。



1. 类型输入刚刚创建的报警媒介类型sendmail脚本 --- 输入个人邮箱用于接收报警邮件 --- 启动打勾 ---- 添加。



1. 选择更新。



7、权限设置(管理员用户不用修改，权限默认所有服务器，如果要分类权限需要在用户组进行控制)



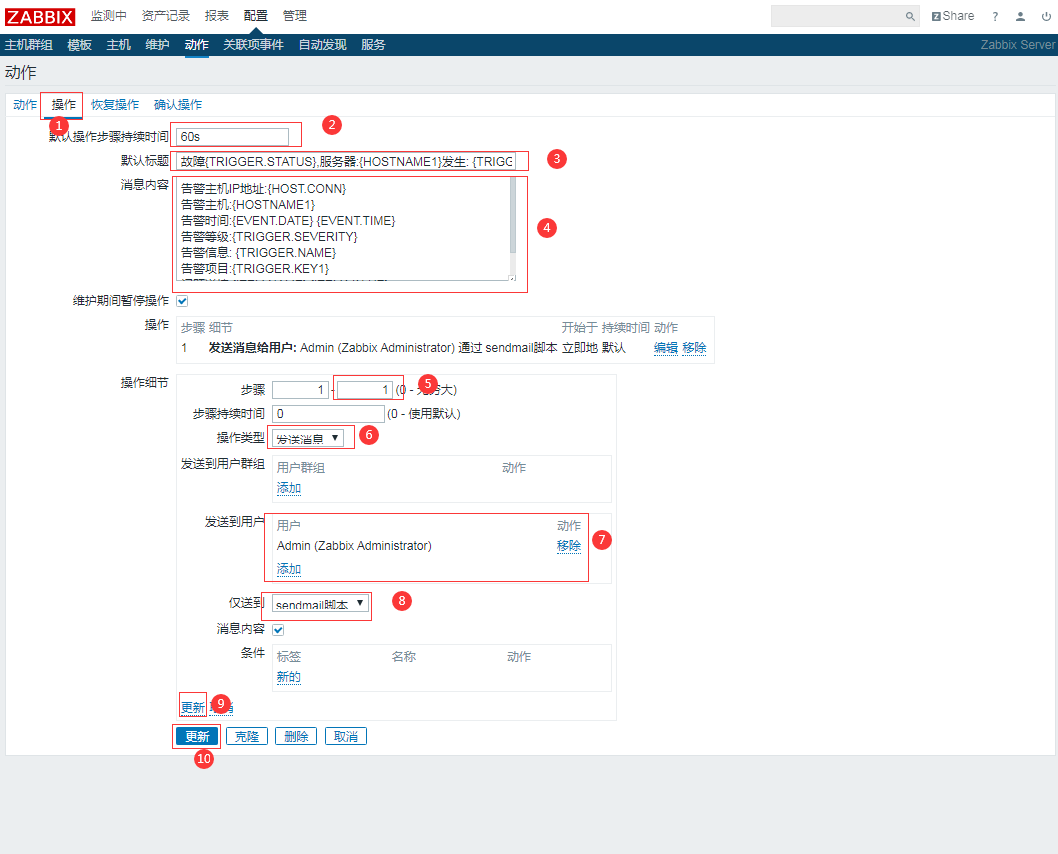
8、配置 ---- 动作 ---- 创建动作。



1. 动作 ---- 名称自定义 --- 设置一条新的触发条件 ---- 启用 ---- 添加。(触发条件需要根据实际情况设置，相当于正则表达式，一般不设置，相当于监控所有监控项)



10、定义操作。（操作细节的步骤1-1每次触发一次）



**默认标题：**故障{TRIGGER.STATUS},服务器:{HOSTNAME1}发生: {TRIGGER.NAME}故障!

**默认信息：**

告警主机IP地址:{HOST.CONN}

告警主机:{HOSTNAME1}

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}

系统默认的格式：

**默认标题：**Problem: {TRIGGER.NAME}

**默认信息：**

Problem started at {EVENT.TIME} on {EVENT.DATE}

Problem name: {TRIGGER.NAME}

Host: {HOST.NAME}

Severity: {TRIGGER.SEVERITY}

Original problem ID: {EVENT.ID}

{TRIGGER.URL}

11、定义恢复操作。



**默认标题：**恢复{TRIGGER.STATUS}, 服务器:{HOSTNAME1}: {TRIGGER.NAME}已恢复!

**恢复信息：**

告警主机IP地址:{HOST.CONN}

告警主机:{HOSTNAME1}

告警时间:{EVENT.DATE} {EVENT.TIME}

告警等级:{TRIGGER.SEVERITY}

告警信息: {TRIGGER.NAME}

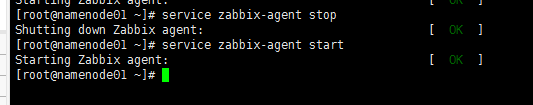
告警项目:{TRIGGER.KEY1}

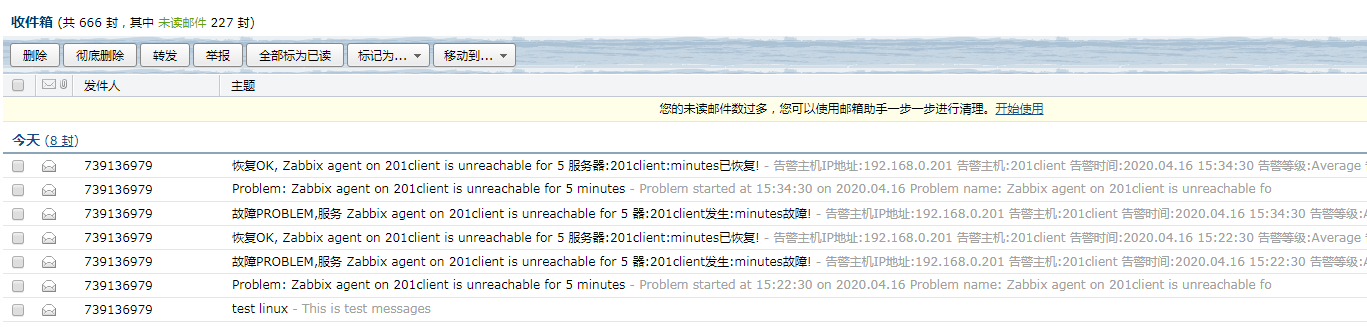
问题详情:{ITEM.NAME}:{ITEM.VALUE}

当前状态:{TRIGGER.STATUS}:{ITEM.VALUE1}

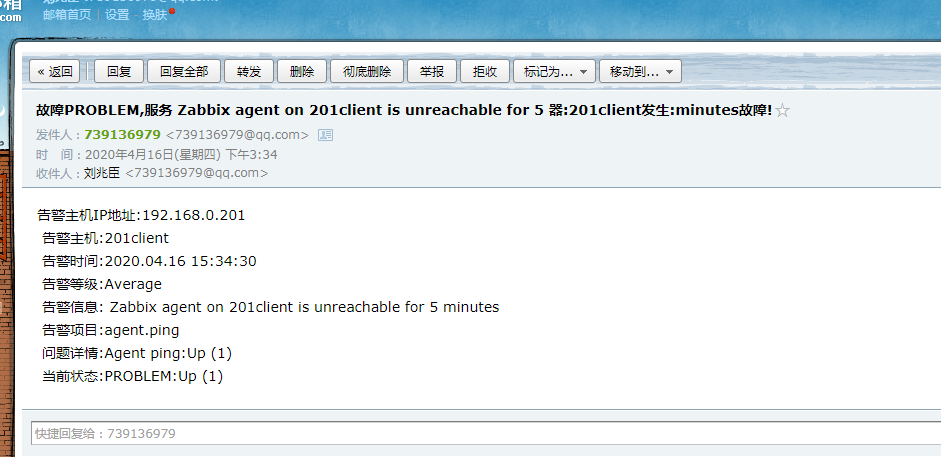
12、测试。

[root@namenode01 ~]# **service zabbix\_agent stop**              #201服务器暂停zabbix\_agentd客户端服务





报警1：



报警2（按不同格式的操作）：



恢复：

