

1. 任意虚拟机编写脚本:

自动安装软件包 vsftpd;

自动创建一个系统账户 tomcat;

在 tomcat 家目录下创建一个测试文件/home/tomcat/test.txt(内容为 hello);

生成 1 个 8 位随机密码, 将 tomcat 账户的密码修改为该随机密码;

将该随机密码通过邮件的形式通知 root (给 root 发邮件, 内容是随机密码);

脚本自动启动 vsftpd, 设置开机自启。

真实机测试, 看看是否可以使用 tomcat 账户登陆虚拟机的 ftp, 并成功读写数据。

```
#vim test.sh
#!/bin/bash
yum -y install vsftpd
useradd tomcat
echo "hello" > /home/tomcat/test.txt
x="abcdefghijklmnopqrstuvwxyz0123456789"
for i in {1..8}
do
    tmp=${x:${RANDOM%62}:1}
    pass=$pass$tmp
done
echo $pass | passwd --stdin tomcat
echo $pass | mail -s password root
systemctl start vsftpd
systemctl enable vsftpd
```

2. 编写脚本:

循环生成 20 个 4 位随机字符串, 以这 20 个随机字符串为文件名, 创建 20 个文件, 扩展名为 jpg。

```
#vim test.sh
#!/bin/bash
x="abcdefghijklmnopqrstuvwxyz0123456789"
for i in {1..20}
do
    pass=""
    for j in {1..8}
    do
        tmp=${x:${RANDOM%62}:1}
        pass=$pass$tmp
    done
    touch /root/$pass.jpg
done
```

3. 编写脚本:

修改前面创建的 20 个随机文件名的文件, 新的文件名为 1.jpg, 2.jpg, 3.jpg 以此类推 (修改文件名)。

```
#vim test.sh
#!/bin/bash
x=1
for i in `ls *.jpg`
do
    mv $i $x.jpg
    let x++
done
```

4. 编写脚本:

将/etc/passwd 中所有的账户名称提取出来。

```
# vim test.sh
#!/bin/bash
n=$(cat /etc/passwd | wc -l)
for i in `seq $n`                #有多数行循环多数次
do
    A=`head -$i /etc/passwd | tail -1`    #逐次读取每一行
    echo ${A%%:*}                        #对读取的数据去尾
done
```

5. 给任意一台虚拟机添加 2 块 10G 磁盘, 编写脚本使用 fdisk 自动分区 (通过 expect 自动交互完成分区)、格式化、挂载。

```
#vim test.sh
#!/bin/bash
expect <<EOF
spawn fdisk /dev/vdb
expect: {send "n/r"}
expect: {send "p/r"}
expect: {send "1/r"}
expect: {send "/r"}
expect: {send "+1g/r"}
expect: {send "w/r"}
expect: {send "exit/r"}
EOF
mkfs.ext4 /dev/vdb1
mkdir /media/vdb1
mount /dev/vdb1 /media/vdb1
```

6. 真实主机安装 expect 软件，编写脚本：

使用 ftp 命令自动连接虚拟机的 FTP 服务器（题目 1 搭建的 ftp）；

结合 expect 自动输入登陆的帐户名 tomcat 及对应的密码；

脚本自动下载 test.txt 文件到真机。

备注：ftp 是需要通过 yum 安装的一个基于命令行的 ftp 客户端软件，语法格式：# ftp 服务器 IP

```
#vim test.sh
```

```
#!/bin/bash
```

```
yum -y install ftp
```

```
expect <<EOF
```

```
spawn ftp 192.168.4.100          #192.168.4.100 为 FTP 服务器的 IP 地址
```

```
expect Name      {send "tomcat/r"}    #tomcat 为用户名
```

```
expect Password  {send "x98ur/r"}     # x98ur 为密码
```

```
expect > {send "get test.txt/r"}
```

```
expect > {send "quit/r"}
```

```
EOF
```

7. 在真机通过 grep 命令过滤下列数据：

从/etc/hosts 中过滤以 127 或 172 开头的数据；

从/etc/passwd 中过滤所有解释器为 bash 的数据

从/etc/passwd 中过滤所有三位以及以上的数字

从/etc/chrony.conf 中过滤所有以 server 开头的内容

从/etc/ssh/sshd_config 中过滤包含 Port 的内容

从/etc/passwd 中过滤以 root 或者 adm 开头的内容

从/var/log/httpd/access_log 中过滤所有包含时间的内容

从/var/log/secure 中过滤包含 Failed 的内容

通过管道将 uptime 命令结果输出给 grep 命令，使用 grep 过滤所有包含小数点的内容

```
# egrep "(127|172)" /etc/hosts
```

```
# egrep "bash$" /etc/passwd
```

```
# egrep "[0-9]{3,}" /etc/passwd
```

```
# egrep "^server" /etc/chrony.conf
```

```
# egrep "Port" /etc/ssh/sshd_config
```

```
# egrep "^(root|adm)" /etc/passwd
```

```
# egrep "[0-9]{4}:[0-9]{2}:[0-9]{2}:[0-9]{2}" /var/log/httpd/access_log
```

```
# egrep "Failed" /var/log/secure
```

```
# uptime | grep "."
```

8. 判断系统中是否已经安装了 httpd, 如果已经安装则提示: 已经安装, 否则直接安装该软件, 并提示完成完成。

```
#vim test.sh
#!/bin/bash
rpm -q httpd &>/dev/null
if [ $? -eq 0 ];then
    echo "已经安装"
else
    yum -y install httpd
    echo "安装完成"
fi
```

9. 编写脚本:

编写函数, 函数的功能是给网卡配置 IP

检测某个网卡是否有 IP (通过 read 读取网卡名称, 如果用户未输入则默认为 eth0)

如果网卡没有 IP, 调用前面的函数给网卡配置 IP

如果网卡已经有 IP, 则提示该网卡已经有 IP 地址。

```
#vim test.sh
#!/bin/bash
myip(){
echo "DEVICE=$1
ONBOOT=yes
BOOTPROTO=static
IPADDR=$2" > /etc/sysconfig/network-scripts/ifcfg-$1
}
read -p "请输入需要配置 IP 的网卡名称:" net
ifconfig $net | ifconfig eth0 |grep -q "inet " #判断是否有 IP
if [ $? -ne 0 ];then
    read -p "请输入 IP:" ip
    myip $net $ip
    systemctl restart network
else
    echo "该网卡已经有 IP"
fi
```