

## RHCE(17-10-30-2)

在线/下载: <https://www.yangxiaoja.me>

RHCE:

两台虚拟机, 请认准机器再做。

本域是domainX.example.com,172.25.X.0/24;敌对域my133t.org, 172.26.X.0/24

\*\*\*\*\*开始练习前先运行, 预备环境 (考试时无需做这个)

练习准备lab teamdesk setup, desktop上做, 创建eno1和eno2网卡

练习准备lab teambridge setup, server上做, 创建eno1和eno2网卡

练习准备lab storageshares setup, server和desktop都需要做,将机器加入域, 能作krb5p

考试准备工作:

一次性放开本域的所有访问(server,desktop)

```
firewall-cmd --permanent --list-all
[root@server16 ~]# firewall-cmd --permanent --add-rich-rule 'rule family=ipv4 source
address="172.25.16.0/24" accept'
success
[root@server16 ~]# firewall-cmd --reload
success
[root@server16 ~]# firewall-cmd --list-all
rich rules:
  rule family="ipv4" source address="172.25.16.0/24" accept
```

如遇到阻止需求, 明确reject即可。

### 正式答题:

#### 1.配置 SELinux(server,desktop)

SELinux 必须在两个系统server 和 desktop中运行于 Enforcing 模式

```
[root@server16 ~]# setenforce 1
[root@server16 ~]# vim /etc/selinux/config
SELINUX=enforcing
```

在两个虚机都做

#### 2.配置SSH访问(server,desktop)

按以下要求配置SSH访问:

用户能够从域 domainX.example.com内的客户端通过SSH远程访问您的两个虚拟机系统,在域 my133t.org 内的客户端不能访问您的两个虚拟机系统

```
[root@server16 ~]# firewall-cmd --permanent --add-rich-rule 'rule family=ipv4 source
address="172.26.16.0/24" service name="ssh" reject'
success
[root@server16 ~]# firewall-cmd --reload
success
[root@server16 ~]# firewall-cmd --list-all
```

rich rules:

```
rule family="ipv4" source address="172.26.16.0/24" service name="ssh" reject
```

在两个虚拟机都做

### 3. 自定义用户环境(server,desktop)

在系统 server 和 desktop 上创建自定义命令名为 qstat 此自定义命令将执行以下命令:

```
/bin/ps -Ao pid,tt,user,fname,rsz
```

此命令对系统中所有用户有效。

```
[root@server16 ~]# vim /etc/bashrc
末尾加 alias qstat='/bin/ps -Ao pid,tt,user,fname,rsz'
[root@server16 ~]# . /etc/bashrc
[root@server16 ~]# qstat
```

在两个虚拟机都做

### 4. 配置端口转发

在系统 server 配置端口转发, 要求如下:

在 172.25.X.0/24 网络中的系统, 访问 server 的本地端口 5423 将被转发到80

此设置必须永久有效

```
[root@server16 ~]# firewall-cmd --permanent --add-rich-rule 'rule family=ipv4 source
address="172.25.16.0/24" forward-port port=5423 protocol=tcp to-port=80'
success
[root@server16 ~]# firewall-cmd --reload
success
[root@server16 ~]# firewall-cmd --list-all
    rule family="ipv4" source address="172.25.16.0/24" forward-port port="5423"
protocol="tcp" to-port="80"
```

### 5. 配置聚合链路(server,desktop)

在 server.domainX.example.com 和 desktop.domainX.example.com 之间按以下要求配置一个链路:

此链路使用接口 eno1 和 eno2

此链路在一个接口失效时仍然能工作

此链路在 server 使用下面的地址 172.16.X.25/255.255.255.0

此链路在 desktop 使用下面的地址 172.16.X.35/255.255.255.0

此链路在系统重启之后依然保持正常状态

[server]

```
[root@server16 ~]# nmcli con show --active
NAME          UUID                                  TYPE          DEVICE
System eth0    5fb06bd0-0bb0-7ffb-45f1-d6edd65f3e03  802-3-ethernet eth0
[root@server16 ~]# nmcli con add type team con-name team0 ifname team0 config
'{"runner":{"name":"activebackup"}}'
[root@server16 ~]# nmcli con modify team0 ipv4.method manual ipv4.addresses
"172.16.16.25/24"
[root@server16 ~]# nmcli con add type team-slave con-name team0-port1 ifname eno1
master team0
[root@server16 ~]# nmcli con add type team-slave con-name team0-port2 ifname eno2
master team0
[root@server16 ~]# teamdctl team0 state
```

```
setup:
  runner: activebackup
ports:
  eno2
    link watches:
      link summary: up
      instance[link_watch_0]:
        name: ethtool
        link: up
  eno1
    link watches:
      link summary: up
      instance[link_watch_0]:
        name: ethtool
        link: up
runner:
  active port: eno1
[root@server16 ~]#
```

[desktop]

desktop端只改变ip其他相同。

#### 6.配置IPv6地址(server,desktop)

在您的考试系统上配置接口 eth0 使用下列IPv6地址:

server 上的地址应该是 200e:ac18::e0a/64

desktop 上的地址应该是 200e:ac18::e14/64

两个系统必须能与网络 200e:ac18/64 内的系统通信。

地址必须在重启后依旧生效。

两个系统必须保持当前的IPv4地址并能通信。

```
[root@server16 ~]# nmcli con modify "System eth0" ipv6.method manual ipv6.addresses
'200e:ac16::e0a/64'
[root@server16 ~]# nmcli con down "System eth0"; nmcli con up "System eth0"
```

两台都做, 复制时注意修改ip;

#### 7.配置本地邮件服务(server,desktop)

在系统server 和 desktop 上 配置邮件服务, 满足以下要求:

这些系统不接收外部发送来的邮件

在这些系统上本地发送的任何邮件都会自动路由到 classroom.example.com

从这些系统上发送的邮件显示来自于 domainX.example.com

您可以通过发送邮件到本地用户 'dave' 来测试您的配置, 系统 classroom.example.com 已经配置把此用户的邮件转到下列URL [http://classroom.example.com/received\\_mail/X](http://classroom.example.com/received_mail/X)

```
[root@server16 ~]# postconf -e "relayhost= classroom.example.com"
[root@server16 ~]# postconf -e "inet_interfaces=loopback-only"
[root@server16 ~]# postconf -e "mynetworks=127.0.0.0/8 [::1]/128"
[root@server16 ~]# postconf -e "myorigin= domainX.example.com"
[root@server16 ~]# postconf -e "mydestination="
[root@server16 ~]# postconf -e "local_transport=error: local delivery disabled"
[root@server16 ~]# systemctl restart postfix
[root@server16 ~]# systemctl enable postfix
[root@server16 ~]# mail -s "Hello" root@server16.example.com
aaaaaaaaaaaaaaaaaaaa
EOT [ctrl+D]
[root@server16 ~]#
```

两台都做

## 8.通过 SMB 共享目录

在server上配置SMB服务

您的 SMB 服务器必须是 STAFF 工作组的一个成员

共享 /common 目录 共享名必须为 common

只有 domainX.example.com 域内的客户端可以访问 common 共享

common 必须是可以浏览的

用户 andy 必须能够读取共享中的内容, 如果需要的话, 验证的密码是 sestiver

[server]

```
[root@server16 ~]# yum -y install samba samba-client
[root@server16 ~]# vim /etc/samba/smb.conf
    workgroup = STAFF

    hosts allow = 127. 172.25.16.
    hosts deny = 172.26.16.
#===== Share Definitions =====
[common]
    path = /common
    browseable = yes
    read list = andy

[root@server16 ~]# mkdir /common
[root@server16 ~]# semanage fcontext -a -t samba_share_t '/common(/.*)?'
[root@server16 ~]# restorecon -vRF /common/
[root@server16 ~]# systemctl start nmbd smb
[root@server16 ~]# systemctl enable nmbd smb
[root@server16 ~]# useradd andy
[root@server16 ~]# smbpasswd -a andy
New SMB password:
Retype new SMB password:
Added user andy.
[root@server16 ~]# pdbedit -L
andy:1001:
```

[desktop]

```
[root@desktop16 ~]# yum install samba-client cifs-utils
[root@desktop16 ~]# smbclient -L //172.25.16.11
    Sharename      Type      Comment
    -----
    common         Disk
[root@desktop16 ~]# smbclient //172.25.16.11/common -U andy%sestiver
Domain=[STAFF] OS=[Unix] Server=[Samba 4.1.1]
smb: \>
smb: \> ls
.                               D            0   Sun Oct 29 14:51:24 2017
..                              D            0   Sun Oct 29 15:07:49 2017

    40913 blocks of size 262144. 28568 blocks available
smb: \> quit
```

## 9.配置多用户SMB 挂载



在server 共享通过SMB目录 /miscellaneous 满足以下要求:

共享名为 miscellaneous

共享目录 miscellaneous 只能被 domainX.example.com 域中的客户端使用

共享目录 miscellaneous 必须可以被浏览

用户 silene 必须能以读的方式访问此共享, 访问密码是 sestiver

用户 akira 必须能以读写的方式访问此共享, 访问密码是 sestiver

此共享永久挂载在 desktop.domainX.example.com 上的 /mnt/multi 目录, 并使用用户 silene 作为认证 任何用户可以通过用户 akira 来临时获取写的权限

[server]

```
[root@server16 ~]# vim /etc/samba/smb.conf
[miscellaneous]
    path = /miscellaneous
    browseable = yes
    read list = silene
    write list = akira
[root@server16 ~]# mkdir /miscellaneous
[root@server16 ~]# chcon -R -t samba_share_t /miscellaneous/
[root@server16 ~]# chmod o+w /miscellaneous/
[root@server16 ~]# ll -d /miscellaneous/
drwxr-xrwx. 2 root root 6 Oct 29 15:07 /miscellaneous/
[root@server16 ~]# systemctl restart nmb smb
[root@server16 ~]# useradd silene
[root@server16 ~]# useradd akira
[root@server16 ~]# smbpasswd -a silene
New SMB password:
Retype new SMB password:
Added user silene.
[root@server16 ~]# smbpasswd -a akira
New SMB password:
Retype new SMB password:
Added user akira.
[root@server16 ~]# systemctl restart nmb smb
[root@server16 ~]#
```

[desktop]

```
[root@desktop16 ~]# touch a b c
[root@desktop16 ~]# smbclient //172.25.16.11/miscellaneous -U silene%sestiver
Domain=[STAFF] OS=[Unix] Server=[Samba 4.1.1]
smb: \> put a
NT_STATUS_ACCESS_DENIED opening remote file \a
smb: \> quit
[root@desktop16 ~]# smbclient //172.25.16.11/miscellaneous -U akira%sestiver
Domain=[STAFF] OS=[Unix] Server=[Samba 4.1.1]
smb: \> put a
putting file a as \a (0.0 kb/s) (average 0.0 kb/s)
smb: \> ls
.                D          0   Sun Oct 29 15:25:29 2017
..               D          0   Sun Oct 29 15:07:49 2017
a                A          0   Sun Oct 29 15:25:29 2017

40913 blocks of size 262144. 28568 blocks available
smb: \>quit
[root@desktop16 ~]# vim /etc/fstab
[root@desktop16 ~]# vim /etc/fstab
//172.25.16.11/miscellaneous /mnt/multi cifs
```

```
credentials=/root/a.txt,multiuser,sec=ntlmssp 0 0
[root@desktop16 ~]# mkdir /mnt/multi
[root@desktop16 ~]# vim a.txt
username=silene
password=sestiver
username=akira
password=sestiver
[root@desktop16 ~]# mount -a
[root@desktop16 ~]# df -h
//172.25.16.11/miscellaneous 10G 3.1G 7.0G 31% /mnt/multi

[root@desktop16 ~]# useradd aaa
[root@desktop16 ~]# useradd bbb
[root@desktop16 ~]# su - aaa
[aaa@localhost ~]$ cd /mnt/
[aaa@localhost mnt]$ ll
ls: cannot access multi: Permission denied
total 0
drwxr-xr-x. 2 root root 6 Oct 29 11:21 data
?????????? ? ? ? ? ? multi
[aaa@localhost mnt]$ cifscreds add server16 -u silene
Password:
[aaa@localhost mnt]$ ll
total 0
drwxr-xr-x. 2 root root 6 Oct 29 11:21 data
drwxr-xrwx. 2 root root 0 Oct 29 15:25 multi
[aaa@localhost mnt]$ cd multi/
[aaa@localhost multi]$ ls
a
[aaa@localhost multi]$ touch aaa
touch: cannot touch 'aaa': Permission denied
[aaa@localhost multi]$ exit
logout
[root@desktop16 ~]# su - bbb
[bbb@localhost ~]$ cd /mnt/
[bbb@localhost mnt]$ ll
ls: cannot access multi: Permission denied
total 0
drwxr-xr-x. 2 root root 6 Oct 29 11:21 data
???????????? ? ? ? ? ? multi
[bbb@localhost mnt]$ cifscreds add server16 -u akira
Password:
[bbb@localhost mnt]$ ll
total 0
drwxr-xr-x. 2 root root 6 Oct 29 11:21 data
drwxr-xrwx. 2 root root 0 Oct 29 15:25 multi
[bbb@localhost mnt]$ cd multi/
[bbb@localhost multi]$ touch aaa
[bbb@localhost multi]$ ll
total 0
-rwxr--r--. 1 aaa aaa 0 Oct 29 15:25 a
-rw-r--r--. 1 aaa aaa 0 Oct 29 15:55 aaa
[bbb@localhost multi]$
```

## 11. 配置NFS服务

在 server 配置NFS服务，要求如下：

以只读的方式共享目录 /public 同时只能被 domainX.example.com 域中的系统访问

以读写的方式共享目录 /protected 能被 domainX.example.com 域中的系统访问

访问 /protected 需要通过Kerberos安全加密，您可以使用下面URL提供的密钥

[http://host.domainX.example.com/pub/materials/nfs\\_server.keytab](http://host.domainX.example.com/pub/materials/nfs_server.keytab).

目录 /protected 应该包含名为 confidential 拥有人为deepak 的子目录

域用户 deepak 能以读写方式访问 /protected/confidential

\*\*\*\*\*练习准备lab storageshares setup, server和desktop都需要做

```
[root@server16 ~]# mkdir /public
[root@server16 ~]# mkdir -p /protected/confidential
[root@server16 ~]# vim /etc/exports
/public 172.25.16.0/24(ro)
/protected 172.25.16.0/24(rw,sec=krb5p)
[root@server16 ~]# exportfs -r
[root@server16 ~]# cd /protected/
[root@server16 protected]# ll
total 0
drwxr-xr-x. 2 root root 6 Oct 29 16:10 confidential
[root@server16 protected]# chown ldapuser16:root confidential/
[root@server16 protected]# ll
total 0
drwxr-xr-x. 2 ldapuser16 root 6 Oct 29 16:10 confidential
[root@server16 protected]# cd
[root@server16 ~]# vim /etc/sysconfig/nfs
RPCNFSDARGS="-V 4.2"
[root@server16 ~]# wget -O /etc/krb5.keytab
http://172.25.254.254/pub/keytabs/server16.keytab
[root@server16 ~]# systemctl start nfs-server
[root@server16 ~]# systemctl start nfs-secure-server
[root@server16 ~]# systemctl enable nfs-server.service
[root@server16 ~]# systemctl enable nfs-secure-server.service
```

## 12.nfs客户端

开机自动挂载

public /mnt/nfsmount

protected /mnt/nfssecure

```
[root@desktop16 ~]# wget -O /etc/krb5.keytab
http://172.25.254.254/pub/keytabs/desktop16.keytab
[root@desktop16 ~]# vim /etc/fstab
172.25.16.11:/public /mnt/nfsmount nfs defaults 0 0
172.25.16.11:/protected /mnt/nfssecure nfs defaults,v4.2,sec=krb5p 0 0
[root@desktop16 ~]# systemctl start nfs-secure
[root@desktop16 ~]# systemctl enable nfs-secure.
[root@desktop16 ~]# mkdir /mnt/{nfsmount,nfssecure}
[root@desktop16 ~]# mount -a
[root@desktop16 ~]# df -h
172.25.16.11:/public          10G  3.1G  7.0G  31% /mnt/nfsmount
172.25.16.11:/protected      10G  3.1G  7.0G  31% /mnt/nfssecure
[root@desktop16 ~]#
```

## 13.配置web站点

server 上配置一个站点 <http://server.domainX.example.com> 然后执行下述步骤:

从 <http://172.25.254.254/content/rhce/web/station.html>

下载文件, 并且将文件重命名为 index.html 不要修改此文件的内容

将文件 index.html 拷贝到您的 web 服务器的 DocumentRoot 目录下

来自于 domainX.example.com 域的客户端可以访问此Web服务

来自于 my133t.org 域的客户端拒绝访问此Web服务

```
[root@server16 ~]# yum -y install httpd
[root@server16 ~]# cd /var/www/html/
[root@server16 html]# wget -O index.html
http://172.25.254.254/content/rhce/web/station.html
[root@server16 html]# ls
index.html
[root@server16 html]# cat index.html
stationX
[root@server16 html]# cd /usr/share/doc/httpd-2.4.6/
[root@server16 httpd-2.4.6]# ls
[root@server16 httpd-2.4.6]# cp httpd-vhosts.conf /etc/httpd/conf.d/13.conf
[root@server16 httpd-2.4.6]# cd /etc/httpd/conf.d
[root@server16 conf.d]# vim 13.conf
<VirtualHost *:80>
    DocumentRoot "/var/www/html"
    ServerName server16.example.com
</VirtualHost>
[root@server16 ~]# systemctl start httpd
[root@server16 ~]# systemctl enable httpd.service
[root@server16 ~]# netstat -ntlp
[root@server16 ~]# firewall-cmd --permanent --add-rich-rule 'rule family=ipv4 source
address=172.26.16.0/24 service name="http" reject'
[root@server16 ~]# firewall-cmd --reload
[root@server16 ~]# firewall-cmd --list-all

[desktop]
[root@localhost ~]# firefox http://server16.example.com
[root@localhost ~]# curl http://server16.example.com
stationX
```

#### 14 配置安全web服务

为站点 <http://server.domainX.example.com> 配置TLS加密 一个已签名证书从  
<http://host.domainX.example.com/pub/materials/server.crt> 获取 此证书的密钥从  
<http://host.domainX.example.com/pub/materials/server.key> 获取 此证书的签名授权信息从  
<http://host.domainX.example.com/pub/materials/domainX.crt> 获取  
[server]

```
[root@server16 ~]# yum -y install mod_ssl
[root@server16 ~]# vim /etc/httpd/conf.d/ssl.conf
DocumentRoot "/var/www/html" (59)
ServerName www.example.com:443 (60)
SSLCertificateFile /etc/pki/tls/certs/server16.crt (100)
SSLCertificateKeyFile /etc/pki/tls/private/server16.key (107)

[root@server16 ~]# cd /etc/pki/tls/
[root@server16 tls]# ls
[root@server16 tls]# cd certs/
[root@server16 certs]# wget http://172.25.254.254/pub/tls/certs/server16.crt
[root@server16 certs]# cd ../private/
[root@server16 private]# wget http://172.25.254.254/pub/tls/private/server16.key
[root@server16 certs]# cd
[root@server16 ~]# systemctl restart httpd.service
```

[desktop]

```
[root@localhost ~]# firefox https://server16.example.com
[root@localhost ~]# curl -k https://server16.example.com
stationX
```

## 15. 配置虚拟主机

在 server 上扩展您的 web 服务器，为站点 <http://www.domainX.example.com> 创建一个虚拟主机，然后执行下述步骤：

设置 DocumentRoot 为 /var/www/virtual

从 <http://classroom.example.com/pub/materials/www.html>

下载文件并重命名为 index.html 不要对文件 index.html 的内容做任何修改

将文件 index.html 放到虚拟主机的 DocumentRoot 目录下

确保 andy 用户能够在 /var/www/virtual 目录下创建文件

注意：原始站点 <http://server.domainX.example.com> 必须仍然能够访问，名称服务器 rhmgr.domainX.example.com 提供对主机名 www.domainX.example.com 的域名解析

[server]

```
[root@server16 ~]# cd /etc/httpd/conf.d/
[root@server16 conf.d]# ls
13.conf autoindex.conf README ssl.conf userdir.conf welcome.conf
[root@server16 conf.d]# cp 13.conf 15.conf
[root@server16 conf.d]# vim 15.conf
<VirtualHost *:80>
    DocumentRoot "/var/www/virtual"
    ServerName www16.example.com
</VirtualHost>
[root@server16 conf.d]# cd /var/www/
[root@server16 www]# mkdir virtual
[root@server16 www]# setfacl -m u:andy:rwX virtual/
[root@server16 www]# setfacl -m d:u:andy:rwX virtual/
[root@server16 www]# cd virtual/
[root@server16 virtual]# wget -O index.html
http://172.25.254.254/content/rhce/web/www.html
[root@server16 virtual]# cd
[root@server16 ~]# systemctl restart httpd.service
```

[desktop]

```
[root@localhost ~]# firefox http://www16.example.com
[root@localhost ~]# curl http://www16.example.com
wwwX
```

## 16 配置 web 内容的访问

在您的 server 上的 web 服务器的 DocumentRoot 目录下 创建一个名为 secret 的目录，要求如下：

从 <http://classroom.example.com/pub/materials/private.html> 下载一个文件副本到这个目录，并且重命名为 index.html。

不要对这个文件的内容做任何修改。

从 server 上，任何人都可以浏览 secret 的内容，但是从其它系统不能访问这个目录的内容

```
[root@server ~]# mkdir /var/www/html/secret
[root@server ~]# mkdir /var/www/virtual/secret
[root@server ~]# cd /var/www/html/secret
```

```
[root@server16 secret]# wget -O index.html
http://172.25.254.254/content/rhce/web/secret1.html
[root@server secret]# cd /var/www/virtual/secret
[root@server16 secret]# wget -O index.html
http://172.25.254.254/content/rhce/web/secret2.html
[root@server16 ~]# vim /etc/httpd/conf/httpd.conf
[root@server16 ~]# cd /etc/httpd/conf.d/
[root@server16 conf.d]# vim 16.conf
<Directory /var/www/html/secret>
    Require local
    Require all denied
</Directory>

<Directory /var/www/virtual/secret>
    Require local
    Require all denied
</Directory>
[root@server16 conf.d]# httpd -t
[root@server16 conf.d]# cd
[root@server16 ~]# systemctl restart httpd.service
[root@server16 ~]# firefox http://server16.example.com/secret
[root@server16 ~]# curl http://server16.example.com/secret/
secret1/
[root@server16 ~]# firefox http://www16.example.com/secret
[root@server16 ~]# curl http://www16.example.com/secret/
secret2/
```

[desktop]

```
[root@desktop16 ~]# firefox http://server16.example.com/secret/
[root@desktop16 ~]# firefox http://www16.example.com/secret/
```

## 17 实现动态Web内容

在 server 上配置提供动态Web内容，要求如下：

动态内容由名为dynamic.domainX.example.com的虚拟主机提供 #####教室练习请用  
webappX.example.com#####

虚拟主机侦听在端口 8998

从 <http://classroom.example.com/pub/materials/webapp.wsgi> 下载一个脚本，然后放在适当的位置，无论如何  
不要求修改此文件的内容

客户端访问 <http://dynamic.domainX.example.com:8998/> 时 应该接收到动态生成的web页面

此 <http://dynamic.domainX.example.com:8998/> 必须能被 domainX.example.com 域内的所有系统访问

[server]

```
[root@server16 ~]# yum install mod_wsgi
[root@server16 ~]# cd /etc/httpd/conf.d/
[root@server16 conf.d]# cp 13.conf 17.conf
[root@server16 conf.d]# vim 17.conf
Listen 8998
<VirtualHost *:8998>
    DocumentRoot "/var/www/html"
    ServerName server16.example.com
    WSGIScriptAlias / /var/www/html/webapp.wsgi
</VirtualHost>
[root@server16 conf.d]# httpd -t
[root@server16 ~]# cd /var/www/html/
[root@server16 html]# wget http://172.25.254.254/content/rhce/web/webapp.wsgi
```



```
[root@server16 ~]# semanage port -l | grep http
[root@server16 ~]# semanage port -a -t http_port_t -p tcp 8998
[root@server16 ~]# systemctl restart httpd.service
```

[desktop]

```
[root@localhost ~]# firefox http://webapp16.example.com:8998
```

## 18 shell编程

文件名/root/script

参数foo显示bar, 参数bar显示foo,

没有参数或错误显示"/root/script foo|bar"错误信息

```
[root@server ~]# vim /root/script
#!/bin/bash
if [ "$#" -eq 1 ]
then
if [ "$1" == "bar" ]
then
echo "foo"
elif [ "$1" == "foo" ]
then
echo "bar"
else
echo "/root/script foolbar" >&2
fi
else
echo "/root/script foolbar" >&2
fi
[root@server16 ~]# chmod a+x /root/script

[root@server16 ~]# ./script foo
bar
[root@server16 ~]# ./script bar
foo
[root@server16 ~]# ./script
/root/script foolbar
```

## 19.创建一个添加用户的脚本

在server上创建一个脚本, 名为/root/mkusers,此脚本能实现为server创建本地用户, 并且这些用户的用户名来自一个包含用户名列表的文件。同时满足下列要求:

此脚本要求提供一个参数, 此参数就是包含用户名列表的文件

如果没有提供参数, 此脚本应该给出下面的提示信息Usage:/root/mkusers filename然后退出并返回相应的值。

如果提供一个不存在的文件名, 此脚本应该给出下面的提示信息 Input file not found然后退出并返回相应的值

创建的用户登录shell是/bin/false

此脚本不需要为用户设置密码

您可以从下面的URL获取用户名列表作为测试用<http://classroom.example.com/pub/materials/userlist>

```
[root@server ~]# vim /root/mkusers
#!/bin/bash
if [ "$#" -ne 1 ]
then echo 'Usage:/root/mkusers  filename' >&2
exit 1
elif [ -f $1 ]
```



```

then
for i in `cat $1`
do
useradd -s /bin/fales $i
done
else
echo 'Input file not found' >&2
exit 1
fi
[root@server ~]# chmod a+x /root/mkusers

```

## 20.配置iscsi服务器

配置server提供一个iscsi服务磁盘名为iqn.2014-09.com.example.domain0:server, 并符合下列要求:

服务端口为3260

使用iscsi\_vol作其后端卷,其大小为3G

此服务只能被desktop.domain0.example.com访问

需要先创建名字叫做iscsi\_vol的逻辑卷。(分区至少3050M)

[server]

```

[root@desktop16 ~]# fdisk /dev/vdb
[root@desktop16 ~]# pvcreate /dev/vdb5
[root@desktop16 ~]# vgcreate vg0 /dev/vdb5
[root@desktop16 ~]# lvcreate -L 3G -n iscsi_vol vg0
[root@desktop16 ~]# lvs
[root@server16 ~]# yum install -y targetcli
[root@server16 ~]# systemctl start target
[root@server16 ~]# systemctl enable target.service
[root@server16 ~]# targetcli
/> /backstores/block create server16.disk1 /dev/vg0/iscsi_vol
/> /iscsi create iqn.2014-09.com.example.domain0:server16
/> /iscsi/iqn.2014-09.com.example.domain0:server16/tpg1/acls create iqn.2014-
06.com.example:desktop16
/> iscsi/iqn.2014-09.com.example.domain0:server16/tpg1/luns create
/backstores/block/server16.disk1
/> /iscsi/iqn.2014-09.com.example.domain0:server16/tpg1/portals create 172.25.16.11
/> saveconfig
/> exit

```

## 21 配置iscsi客户端

配置desktop使其能连接在server的上提供的iqn.2014-09.com.example.domain2:server并符合下列要求:

块设备iSCSI上包含一个大小为1700 MiB的分区,并格式化为xfs

此分区挂载在/mnt/data上同时在系统启动的期间自动挂载

[desktop]

```

[root@desktop16 ~]# vim /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.2014-06.com.example:desktop16
[root@desktop16 ~]# iscsiadm -m discovery -t st -p server16.example.com
172.25.16.11:3260,1 iqn.2014-09.com.example.domain0:server16
[root@desktop16 ~]# systemctl restart iscsid
[root@desktop16 ~]# systemctl restart iscsi
[root@desktop16 ~]# systemctl enable iscsid
[root@desktop16 ~]# systemctl enable iscsi
[root@desktop16 ~]# iscsiadm -m discovery -t st -p server16.example.com
172.25.16.11:3260,1 iqn.2014-09.com.example.domain0:server16
[root@desktop16 ~]# iscsiadm -m node -T iqn.2014-09.com.example.domain0:server16

```

```
[root@desktop16 ~]# iscsiadm -m node -T iqn.2014-09.com.example.domain0:server16 -l

[root@desktop16 ~]# fdisk -l
Disk /dev/sda: 3221 MB, 3221225472 bytes, 6291456 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 4194304 bytes
[root@desktop16 ~]# fdisk /dev/sda (sda1 1700M)
[root@desktop16 ~]# mkfs.xfs /dev/sda1
[root@desktop16 ~]# blkid /dev/sda1
[root@desktop16 ~]# cp /etc/fstab .
[root@desktop16 ~]# blkid /dev/sda1 >> /etc/fstab
[root@desktop16 ~]# vim /etc/fstab
UUID="9471c035-d171-4c48-bc53-d877ef9bf001" /mnt/data xfs defaults,_netdev 0 0
[root@desktop16 ~]# mkdir /mnt/data
[root@desktop16 ~]# mount -a
[root@desktop16 ~]# df -h
/dev/sda1      1.7G   33M   1.7G    2% /mnt/data
```

## 22. 配置一个数据库

在server上创建一个MariaDB 数据库,名为Contacts,并符合以下条件:

数据库应该包含来自数据库复制的内容,复制文件的URL

为<http://classroom.example.com/pub/materials/users.mdb>

数据库只能被localhost访问

除了root用户, 此数据库只能被Luigi查询。此用户密码为mishigs

root密码mishings,同时不允许空密码登录。

```
[root@server16 ~]# yum install mariadb-server
[root@server16 ~]# vim /etc/my.cnf
skip-networking=1 (2+)
[root@server16 ~]# systemctl start mariadb
[root@server16 ~]# systemctl enable mariadb.service
[root@server16 ~]# mysql
MariaDB [(none)]> \q
Bye
[root@server16 ~]# mysql_secure_installation
Enter current password for root (enter for none): [enter]
Set root password? [Y/n] y
New password:
Re-enter new password:
Remove anonymous users? [Y/n] y
Disallow root login remotely? [Y/n] y
Remove test database and access to it? [Y/n] y
Reload privilege tables now? [Y/n] y
[root@server16 ~]# mysql -uroot -pmishings
MariaDB [(none)]> create database Contacts;
MariaDB [(none)]> grant select on Contacts.* Luigi@localhost identified by
'mishigs';
MariaDB [(none)]> flush privileges;
MariaDB [(none)]> exit
[root@server16 ~]# mysql -uLuigi -pmishigs
MariaDB [(none)]> exit
[root@server16 ~]# wget http://172.25.254.254/pub/materials/mariadb/mariadb.dump
[root@server16 ~]# file mariadb.dump
mariadb.dump: ASCII text
[root@server16 ~]# vim mariadb.dump
[root@server16 ~]# mysql -uroot -pmishings Contacts < mariadb.dump
[root@server16 ~]# mysql -uLuigi -pmishigs Contacts
```

```
MariaDB [Contacts]> show databases
MariaDB [Contacts]> \q
[root@server16 ~]#
```

### 23. 数据库中的数据查询， 填空

数据库中有三张表，“User\_Names User\_Logins User\_Contacts”

在密码表中查出密码是forsook的用户叫什么名字

查出first name(看英文题)是Alan用户住址在Cupertino的用户个数

```
select User_Names.first_name from User_Names,User_Logins where
User_Names.user_id=User_Logins.id and User_Logins.User_Pass="forsook";
或select user.name from user inner join infor on user.id=infor.id where
infor.pass="tapacalaya";

select count(*) from User_Contacts,User_Names where
User_Contacts.Location="Cupertino" and User_Names.first_name="Alan" and
User_Contacts.id=User_Names.user_id;
或select count(*) from user inner join address on user.id=address.id where
user.name="john" and address.addr="Washion";
```