# 系统安全保护、配置用户环境、配置高级连接、防火墙策略管理

**for** 循环结构

```
 for   haha   in zhangsan   lisi   wangwu   tianqi
 do
       useradd $haha
 done


[root@server0 opt]# vim   /root/for.sh

#!/bin/bash
 for   i   in   stu01   stu02   stu03   stu04
 do
     useradd $i
     echo $i 创建成功
 done
```

```
##################################################
```

在 **server0** 上创建 **/root/batchusers** 脚本

**1)**此脚本要求提供用户名列表文件作为参数

```
[root@server0 ~]# vim /root/userlist
[root@server0 ~]# cat /root/userlist
duanwu
zhongqiu
yd
gq
```

**2)**如果没有提供参数,此脚本应该给出

提示 **Usage: /root/batchusers,**退出并返回相应值

**3)**如果提供一个不存在的文件,此脚本应该给出提

示 **Input file not found,**退出并返回相应值

**4)**新用户的登录 **Shell** 为 **/bin/false,**无需设置密码

```
[root@server0 ~]# cat /root/batchusers
#!/bin/bash
 if [ $# -eq 0 ];then
   echo 'Usage: /root/batchusers' >&2
   exit 1
```

```
 elif [ -f $1 ];then
  for i in `cat   /root/userlist`
  do
   useradd -s /bin/false $i   &> /dev/null
   echo $i 创建完成
  done
else
   echo 'Input file not found' >&2
   exit 2
fi
```

##########################################

# SELinux 安全机制,系统安全保护

- Security-Enhanced Linux
  - 美国 NSA 国家安全局主导开发,一套增强 Linux 系统安全的强制访问控制体系
  - 集成到 Linux 内核(2.6 及以上)中运行
  - RHEL7 基于 SELinux 体系针对用户、进程、目录和文件提供了预设的保护策略,以及管理工具

- **SELinux 的运行模式**
  - **enforcing(强制)、permissive(宽松)**
  - **disabled(彻底禁用)**

**任何状态变成 disabled(彻底禁用)都需要重起**

- 切换运行模式
  - 临时切换:setenforce 1|0
  - 固定配置:/etc/selinux/config 文件

虚拟机 Server:

```
[root@server0 ~]# getenforce
Enforcing
[root@server0 ~]# setenforce 0
[root@server0 ~]# getenforce
permissive

[root@server0 ~]# vim /etc/selinux/config
 SELINUX=permissive
```

虚拟机 desktop:

```
[root@desktop0 ~]# getenforce
Enforcing
```

[root@desktop0 ~]# setenforce 0
[root@desktop0 ~]# getenforce
permissive

[root@desktop0 ~]# vim /etc/selinux/config
SELINUX=permissive
##################################################

# 配置高级连接

一 、配置聚合连接 (也称为链路聚合　有称为网卡绑定)　　team 组队
HSRP 热备份路由协议

活跃 路由器　　备份 路由器

虚拟路由器

team 组队　　活跃　eth1　　备份 eth2

虚拟的网卡

- 作用 2:热备份(activebackup)连接冗余

**1. 创建一个新的虚拟网卡 team0,参考 man teamd.conf　按 大写的 G 从后向上**

# nmcli　connection　add　type　team
autoconnect　yes　con-name　team0　ifname　team0
config　'{"runner": {"name": "activebackup"}}'

添加一个类型为 team 的网卡
每次开机自动启用该网卡　配置文件的名字为 team0　网卡名为 team0
team0 工作的模式为'{"runner": {"name": "activebackup"}}'(热备方式)

# ifconfig | less　　#查看 team0 网卡信息
# ls /etc/sysconfig/network-scripts/ifcfg-team0　#生成配置文件
# cat /etc/sysconfig/network-scripts/ifcfg-team0

**2. 添加 奴隶,成员**

# nmcli connection add type team-slave con-name team0-1 ifname eth1 master team0

# nmcli connection add type team-slave con-name team0-2 ifname eth2 master team0

添加一个类型为 team-slave 的成员 配置文件为 team0-1
网卡为 eth1 添加到 team0

### 3. 配置 team0 的 IP 地址

# nmcli connection modify team0 ipv4.method manual ipv4.addresses 192.168.1.1/24 connection.autoconnect yes

### 4. 激活 team0

[root@server0 ~]# nmcli connection up team0
[root@server0 ~]# nmcli connection up team0-1
[root@server0 ~]# nmcli connection up team0-2
[root@server0 ~]# ifconfig team0

### 5. 删除重做

[root@server0 ~]# nmcli connection delete team0
[root@server0 ~]# nmcli connection delete team0-1
[root@server0 ~]# nmcli connection delete team0-2

### 6.专测试 team0 的命令

[root@server0 ~]# teamdctl team0 state

##########################################################

**配置永久的别名:**

用户家目录/.bashrc:用户配置文件,仅针对与用户本身,新开一个终端
/etc/bashrc:全局配置文件,所有用户均生效, 新开一个终端

[root@server0 ~]# vim /root/.bashrc
alias hello='echo hi'

[root@server0 ~]# vim /home/student/.bashrc
alias hi='echo hello'

[root@server0 ~]# vim /etc/bashrc
alias haha='echo xixi'

新开一个终端,分别用 root 用户与 student 用户验证

##########################################################

**配置 IPv6 地址**

IP 地址的作用:唯一标识一个网络节点的地址
IPv4: 32 个二进制数 方便使用 用 4 个十进制数来表示 ,以点
IPv6: 128 个二进制数 方便使用 分 8 段 每段用 4 个 16 进制数来表示,以冒号 :

虚拟机 server:
# nmcli connection modify 'System eth0' ipv6.method manual ipv6.addresses 2003:ac18::305/64 connection.autoconnect yes

# nmcli connection up 'System eth0'

# ifconfig | less

# ping6  2003:ac18::305
##################################################
防火墙策略管理
作用: 隔离  允许出站的请求,过滤入站的请求

分为：软件防火墙、硬件防火墙
##################################################
**确认 防火墙服务是否开启**

[root@server0 ~]# systemctl  status  firewalld
[root@desktop0 ~]# systemctl  status  firewalld
##################################################
**搭建 Web 服务**
服务端:server
**1.安装一个可以提供 Web 服务软件**
[root@server0 ~]# yum -y install httpd

**2.启动 httpd 服务**
[root@server0 ~]# systemctl restart httpd
[root@server0 ~]# systemctl enable httpd

**3.自己访问自己测试**
[root@server0 ~]# firefox 127.0.0.1  #可以看到一个测试页面

**4.书写网页文件,默认位置/var/www/html/index.html**
[root@server ~]# vim /var/www/html/index.html
<marquee><font color=red><h1>hahaxixihehelele

[root@server0 ~]# firefox  127.0.0.1

**搭建 FTP 服务**
服务端 server
**1.安装一个提供 FTP 功能软件**
[root@server ~]# yum -y install vsftpd

**2.启动 vsftpd 服务**

[root@server ~]# systemctl restart vsftpd
[root@server ~]# systemctl enable vsftpd

**3.自己访问自己测试**

[root@server0 ~]# firefox ftp://127.0.0.1　　#可以看到一个目录

**4. 默认共享目录/var/ftp**

##################################################
RHEL7 的防火墙体系

- 系统服务:firewalld
- 管理工具:firewall-cmd、firewall-config
- 根据所在的网络场所区分,预设保护规则集
  - **public:仅允许访问本机的 sshd 等少数几个服务**
  - **trusted:允许任何访问**
  - **block:阻塞任何来访请求　　#明确拒绝**
  - **drop:丢弃任何来访的数据包　　#直接丢弃不给回应**

防火墙的判断机制:

1.查看访问请求中的源 IP 地址,在所有区域中,哪一个区域有该 IP 地址的策略则进入哪一个区域

2.进入默认区域(管理员可以修改)

##################################################
虚拟机 server

```
# firewall-cmd --get-default-zone          #查看默认区域是什么
# firewall-cmd --zone=public --list-all   #查看区域规则
```

虚拟机 desktop

```
# ping 172.25.0.11     #可以通信
```

虚拟机 server

```
# firewall-cmd --set-default-zone=block   #修改默认区域
# firewall-cmd --zone=block --list-all      #查看区域规则
```

虚拟机 desktop

```
# ping 172.25.0.11     #不可以通信,有回应
```

虚拟机 server

```
# firewall-cmd --set-default-zone=drop   #修改默认区域
# firewall-cmd --zone=drop --list-all     #查看区域规则
```

虚拟机 desktop

```
# ping 172.25.0.11     #不可以通信,没有回应
```

##################################################
**添加服务与源 IP 地址**

虚拟机 server

```
# firewall-cmd --set-default-zone=public     #修改默认区域
# firewall-cmd --get-default-zone
```

虚拟机 desktop

```
# firefox 172.25.0.11              #不可以访问
# firefox ftp://172.25.0.11     #不可以访问
```

虚拟机 server

```
# firewall-cmd --zone=public   --add-service=ftp     #添加服务
# firewall-cmd --zone=public   --add-service=http    #添加服务
# firewall-cmd --zone=public   --list-all
```

虚拟机 desktop

```
# firefox 172.25.0.11              #可以访问成功
# firefox ftp://172.25.0.11      #可以访问成功
```

##########################################################

**配置规则的位置**

- 运行时(runtime)
- 永久(permanent)

虚拟机 server0

```
# firewall-cmd --reload                  #重新加载防火墙所有的配置
# firewall-cmd --zone=public   --list-all

# firewall-cmd --permanent --zone=public --add-service=http
# firewall-cmd --zone=public   --list-all
# firewall-cmd --reload
# firewall-cmd --zone=public   --list-all
```

默认区域的修改,默认就是永久的,不许要加上 --permanent

##########################################################

**虚拟机 server 实现本机的端口转发**

- 本地应用的端口重定向(5423 --> 80)
  - 从客户机访问 server 5423 的请求,自动映射到本机 80
  - 比如,访问以下两个地址可以看到相同的页面:

http://172.25.0.11:5423/  ------>  http://172.25.0.11:80

虚拟机 server0

```
# firewall-cmd --set-default-zone=public
# firewall-cmd --permanent --zone=public --add-service=http

# firewall-cmd --permanent --zone=public
  --add-forward-port=port=5423:proto=tcp:toport=80

# firewall-cmd --reload
# firewall-cmd --zone=public   --list-all
```

虚拟机 desktop0

```
# firefox http://172.25.0.11:5423
```