

# 云计算应用管理

**NSD ENGINEER**

**DAY03**

# 内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	系统安全保护
	10:30 ~ 11:20	配置用户环境
	11:30 ~ 12:00	配置高级连接
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	防火墙策略管理
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑



## 系统安全保护



# SELinux安全机制

## SELinux概述

- 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运行模式的切换

知识讲解

- SELinux的运行模式
  - enforcing ( 强制 ) 、 permissive ( 宽松 )
  - disabled ( 彻底禁用 )
- 切换运行模式
  - 临时切换 : setenforce 1|0
  - 固定配置 : /etc/selinux/config 文件

```
[root@server0 ~]# getenforce           //查看当前模式
Disabled
```

```
[root@server0 ~]# vim /etc/selinux/config
SELINUX=enforcing                       //设置为强制启用
```

```
.. ..
[root@server0 ~]# reboot                 //重启系统以切换模式
```



## 案例1：启用SELinux保护

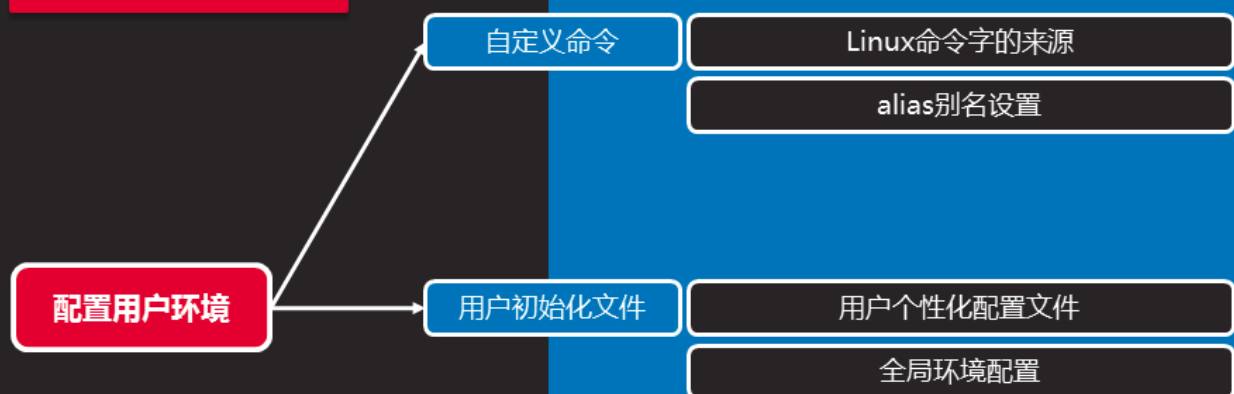
为虚拟机 server0、desktop0 配置SELinux

- 1) 确保 SELinux 处于强制启用模式
- 2) 在每次重新开机后，此设置必须仍然有效

课堂练习



## 配置用户环境



## 自定义命令

# Linux命令字的来源

知识讲解

- 如何指定命令字
  - 指令名：函数 > 别名 > 内部命令 > 外部命令
  - 可执行程序的路径
- 什么是别名
  - 在用户环境中，为一个复杂的、需要经常使用的命令行所起的短名称
  - 可用来替换普通命令，更加方便



## alias别名设置

知识讲解

- 查看已设置的别名
  - alias [别名名称]
- 定义新的别名
  - alias 别名名称= '实际执行的命令行'
- 取消已设置的别名
  - unalias [别名名称]

```
[root@server0 ~]# alias qstat='/bin/ps -Ao pid,tt,user,fname,rsz'
[root@server0 ~]# qstat
.. ..
```



# 用户初始化文件

## 用户个性化配置文件

- 影响指定用户的 bash 解释环境
  - `~/.bashrc` , 每次开启 bash 终端时生效

知识讲解

```
[root@server0 ~]# vim ~student/.bashrc
```

```
.. ..
```

```
alias ld='ls -lhd --color=auto'
```

```
[root@server0 ~]# su - student
```

//仅对 student 用户有效

```
[student@server0 ~]$ alias ld
```

```
alias ld='ls -lhd --color=auto'
```



# 全局环境配置

知识讲解

- 影响所有用户的 bash 解释环境
    - `/etc/bashrc` , 每次开启 bash 终端时生效
- ```
[root@server0 ~]# vim /etc/bashrc
.. ..
alias qstat='/bin/ps -Ao pid,tt,user,fname,rsz'

[root@server0 ~]# su - root           //对所有用户有效
[root@server0 ~]# qstat
.. ..
```



## 案例2：自定义用户环境

为系统 server0 和 desktop0 创建自定义命令

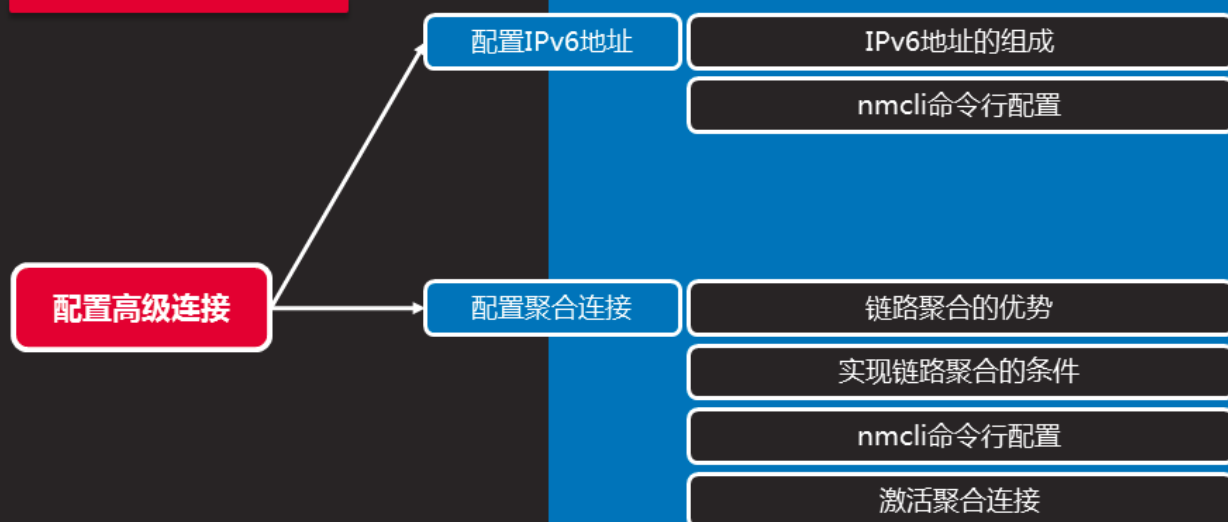
- 1) 自定义命令的名称为 qstat
- 2) 此自定义命令将执行以下操作：  
`/bin/ps -Ao pid,tt,user,fname,rsz`
- 3) 此自定义命令对系统中的所有用户都有效

课堂练习





## 配置高级连接



## 配置IPv6地址

# IPv6地址的组成

## 知识讲解

- IPv4 地址表示
  - 32个二进制位，点分隔的十进制数
  - 例如：172.25.0.11、127.0.0.1
- IPv6 地址表示
  - 128个二进制位，冒号分隔的十六进制数
  - 每段内连续的前置 0 可省略、连续的多个：可简化为 ::
  - 例如：2003:ac18:0000:0000:0000:0000:0000:0305  
2003:ac18::305



# nmcli命令行配置

## 知识讲解

- 基本配置方法

### 1) 使用命令行修改连接参数

```
[root@server0 ~]# nmcli con show //获知连接名称
```

| NAME        | UUID       | TYPE           | DEVICE |
|-------------|------------|----------------|--------|
| System eth0 | 5fb06.. .. | 802-3-ethernet | eth0   |

```
[root@server0 ~]# nmcli con mod "System eth0" ipv6.method manual ipv6.addresses 2003:ac18::305/64
```

### 2) 激活更改过的连接（必要时先down再up）

```
[root@server0 ~]# nmcli connection up "System eth0"
.. ..
```



## 案例3：配置IPv6地址

课堂练习

为两个虚拟机的接口 eth0 配置下列 IPv6 地址

- server0 上的地址应该是 2003:ac18::305/64
- desktop0 上的地址应该是 2003:ac18::306/64
- 两个系统必须能与网络 2003: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



## 配置聚合连接

## 链路聚合的优势

知识讲解

- team , 聚合连接 ( 也称为链路聚合 )
  - 由多块网卡 ( team-slave ) 一起组建而成的虚拟网卡 , 即 “组队”
  - 作用1 : 轮询式 ( [roundrobin](#) ) 的流量负载均衡
  - 作用2 : 热备份 ( [activebackup](#) ) 连接冗余

运行器的类型切换 ( 参考 [man teamd.conf](#) ) ——  
{"runner":{"name":"roundrobin"}}  
或者  
{"runner":{"name":"activebackup"}}



## 实现链路聚合的条件

知识讲解

- 网络接口的准备
  - 2块或2块以上的物理网卡

```
[root@server0 ~]# ifconfig -a | grep ^eth  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
```



# nmcli命令行配置

知识讲解

- 配置文件的准备
  - 为聚合连接提供配置（类型、连接名、运行器、IP地址）

```
[root@server0 ~]# nmcli con add con-name team0 type team
ifname team0 config '{ "runner":{ "name":"activebackup" } }'
```

```
.. ..
```

```
[root@server0 ~]# nmcli con mod team0 ipv4.method manual
ipv4.addresses '172.16.3.20/24' connection.autoconnect yes
```

- 为成员网卡提供配置（类型、连接名、主连接）

```
[root@server0 ~]# nmcli con add con-name team0-p1 type
team-slave ifname eth1 master team0
```

```
.. ..
```

```
[root@server0 ~]# nmcli con add con-name team0-p2 type
team-slave ifname eth2 master team0
```

```
.. ..
```



## 激活聚合连接

知识讲解

- 分别激活聚合连接、成员连接

```
[root@server0 ~]# nmcli con up team0
```

```
[root@server0 ~]# nmcli con up team0-p1
```

```
[root@server0 ~]# nmcli con up team0-p2
```

- 检查聚合连接状态

```
[root@server0 ~]# teamdctl team0 state
setup:
```

```
runner: activebackup
```

```
ports:
```

```
eth1
```

```
.....:
```

```
eth2
```

```
.....:
```



## 案例4：配置聚合连接

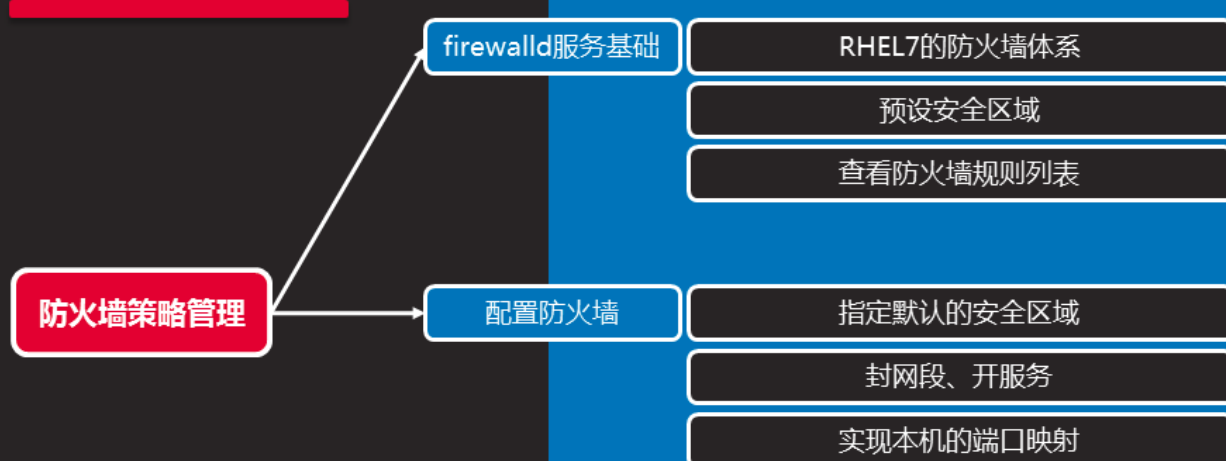
课堂练习

在两个虚拟机之间配置一个链路，要求如下：

- 此链路使用接口 eth1 和 eth2
- 此链路在其中一个接口失效时仍然能工作
- 此链路在 server0 上使用下面的地址  
172.16.3.20/255.255.255.0
- 此链路在 desktop0 上使用下面的地址  
172.16.3.25/255.255.255.0
- 此链路在系统重启之后依然保持正常状态



### 防火墙策略管理



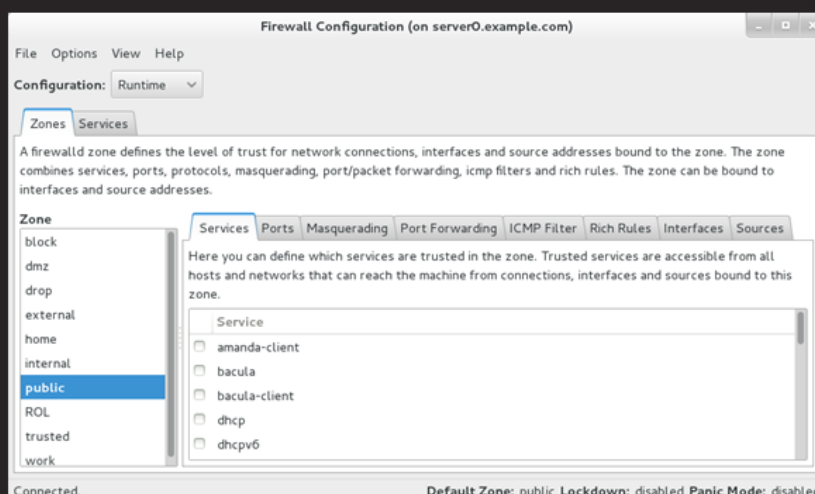
# firewalld服务基础

## RHEL7的防火墙体系

- 系统服务：firewalld
- 管理工具：firewall-cmd、firewall-config

```
[root@server0 ~]# systemctl restart firewalld  
[root@server0 ~]# firewall-config &
```

知识讲解



## 预设安全区域

知识讲解

- 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
  - **drop**：丢弃任何来访的数据包
  - .....
- 配置规则的位置
  - 运行时 ( runtime )
  - 永久 ( permanent )



## 查看防火墙规则列表

知识讲解

- 列表查看操作
  - `firewall-cmd --list-all [--zone=区域名]`
  - `firewall-cmd --list-all-zones`
  - `firewall-cmd --get-zones`
  - `firewall-cmd --get-services`
  - `firewall-cmd --get-default-zone`





# 配置防火墙

## 指定默认的安全区域

知识讲解

- 使用 `--set-default-zone=区域名`
  - 默认为 public , 限制较严格
  - 对于开放式环境, 建议将默认区域修改为 trusted
  - 针对 “运行时/永久配置” 均有效

```
[root@server0 ~]# firewall-cmd --get-default-zone    //修改前  
public
```

```
[root@server0 ~]# firewall-cmd --set-default-zone=trusted
```

```
[root@server0 ~]# firewall-cmd --get-default-zone    //修改之后  
trusted
```



## 封网段、开服务

知识讲解

- 若针对“永久配置”，需添加 `--permanent`
  - 使用 `--add-source=网段地址`
  - 使用 `--add-service=服务名`

```
[root@server0 ~]# firewall-cmd --permanent --zone=block --add-source=172.34.0.0/24
```

```
[root@server0 ~]# firewall-cmd --permanent --zone=public --add-service=http
```

```
[root@server0 ~]# firewall-cmd --permanent --zone=public --add-service=ftp
```

```
[root@server0 ~]# firewall-cmd --reload //重载配置
```



## 实现本机的端口映射

知识讲解

- 本地应用的端口重定向（端口1 --> 端口2）
  - 从客户机访问 端口1 的请求，自动映射到本机 端口2
  - 比如，访问以下两个地址可以看到相同的页面：

`http://server0.example.com:5423/`

`http://server0.example.com/`

```
[root@server0 ~]# firewall-cmd --permanent --zone=trusted --add-forward-port=port=5423:proto=tcp:toport=80
```

```
[root@server0 ~]# firewall-cmd --reload //重载配置
```



## 案例5：配置firewalld防火墙

课堂练习

为你的两个虚拟机配置防火墙策略

- 允许从 172.25.0.0/24 网段的客户机访问 server0、desktop0 的任何服务
- 禁止从 my133t.org 域 ( 172.34.0.0/24网段 ) 的客户机访问 server0、desktop0 的任何服务
- 在172.25.0.0/24网络中的系统，访问 server0 的本地端口5423将被转发到80
- 上述设置必须永久有效



### 总结和答疑



# 配置IPv6地址

## 问题现象

- 配置IPv6地址失败或异常
  - 问题1：配置了IPv6地址，但 ifconfig 看不到
  - 问题2：配置了IPv6地址以后，重启系统发现主机名变成了 localhost.localdomain



# 故障分析及排除

知识讲解

- 原因分析
  - 问题1：新修改的配置没有生效
  - 问题2：此主机未配置静态主机名，当存在有多个IP地址时，无法查询获知自身的主机名
- 解决办法
  - 问题1：使用 nmcli 重新激活连接（down，up）
  - 问题2：修改 /etc/hostname文件，设置固定主机名



## 配置聚合连接

## 问题现象

知识讲解

- 聚合连接 team0 运行异常
  - 新建的 team0 连接激活失败，也看不到IP地址

```
[root@server0 ~]# teamdctl team0 state
setup:
  runner: activebackup
ports:
```

```
[root@server0 ~]#
```



## 故障分析及排除

知识讲解

- 原因分析
  - 运行器配置 { ... } 编写有误，
  - 或者未激活成员连接
- 解决办法
  - 使用正确的 {"runner":{.. ..}} 配置
  - 依次激活 team0 以及 team0-p1、team0-p2 等成员连接



