Linux 学 习 笔 记 整 理

黄垣华

小奥

www.lapland.name

目　　录

1.Linux 系统安装.....3.15.1
2.Linux 常用命令.....3.16.1
3.Linux 常用命令(II).....3.17.1
4.Vim/ Vi 文本编辑器.....3.17.2
5.Linux 引导流程解析.....3.18.1
6.Linux 软件包管理.....3.21.1
7.Linux 用户管理.....3.22.1
8.Linux 进程管理.....3.23.1
9.Linux 文件系统管理.....3.24.1
10.Shell 编程.....3.28.1
11.Linux 网络设置.....3.29.1
12.Linux 文件共享服务.....3.29.2
13.LAMP 环境搭建.....3.31.1
14.APACHE 服务器配置.....4.01.1
15.LVM 逻辑卷管理.....4.01.2
16.RAID.....4.02.1
17.防火墙设置 Netfilter/Iptables.....4.02.2

3.15.1 Linux 系统安装

采用 WMware 虚拟机安装 linux；Linux 安装过程如下：

Redhat Linux 三种版本：AS（用于大型企业）ES（用于小型企业）WS（用于工作站以及台式机）

我们这里采用 redhat 社区服务器版：CentOS，最新稳定版为 5.5 。

Linux 的目录结构：
/ 分区 表示是 Linux 操作系统的根分区 这个分区里面一般都是操作系统和一些服务的配置文件.
swap 分区 这是 Linux 的交换分区类似于 Windows 的虚拟内存技术.交换分区一般是你物理内存的一到两倍.

安装界面：
1---图形界面：一般默认安装就是图形安装（直接按 enter 进入安装）
2---字符界面：输入 Linux text 安装
3---其他安装:可以输入 Linux askmethod 可以选择由软盘 或者是说硬盘进行安装包括 FTP 安装...

安装过程：
第一步 我们选择在图形化模式下安装 CENTOS 系统，所以点击【ENTER】进入下一步，此时出现窗口询问是否检查安装文件正确、完整，当我确定光盘正确，也为了节约时间我选在【SKIP】跳过该检测
第二步 接着我们正式进入到 CENTOS 的安装界面
第三步 选在安装过程中你希望使用的语言，这里我选在【简体中文】，单击【Next】
第四步 接下来为系统选择当前服务器键盘类型，我们选在【美国英语式】，点击【Next】，弹出窗口提示：驱动器上的分区无法被读取..是否初始化驱动器并删除所有数据，我们选在【是】，进入下一步

第五步 接着进入到磁盘分区设置界面，你可以为所需安装的 CENTOS 系统建了默认的磁盘分区，也可以手动进行磁盘分区；这里我使用手动分区，所以我选在【建立自定义的分区结构】，点击【下一步】进入具体的分区配置窗口
在这里我们只需创建一个根分区和一个交换分区 其他的分区我们可以挂载到其他目录下面去，一般我们的根分区的话要大一点.
（）fixed size 指定大小
（）fill maximum size of 自动调整大小.
（）fill all available space 自动将剩下的容量全分配到这个分区上面.
第六步 单击【下一步】,【下一步】进去网卡参数设置窗口，这里我们可以点击【编辑】配置 Ipv4 的 IP 地址和子网掩码,我们这里选择通过 DHCP 来分配自己的 IP 地址,直接单击【下一步】。

第七步　在下边窗口选在时区，我们选在【亚洲/上海】，单击【下一步】
第八步　接下来为根账户设置密码，如下，设置完成点击【下一步】
第九步　在接下来的窗口中，我们选在需要默认安装的部分入软件，根据实际安装需求进行选择后，然后点击【下一步】
第十步 进入安装确认界面，再次单击【下一步】进行安装
第十一步　安装进行时界面，这里我们只需等待，完成安装，系统将自动提示我们重启服务器，重新加载成功后，我们既成功完成 CENTOS 的系统安装工作。

使用 ls /proc 可以查看内核里面的一些文件.
more /proc/cpuinfo 查看你的 CPU 信息
more /proc/meminfo 查看你的内存情况.

df 命令是查看你挂载的磁盘使用情况
-a --all　包括全部的文件系统。
-B --block-size=<区块大小> 来指定的区块大小来显示区块数目
-h --human-readable 以兆字节的方式来查看分区的大小
-H 与-h 一样的功能...以兆字节来显示是 1000K 为一兆 并非 1024KB

用户的创建:
useradd 用户名
passwd 用户密码

注意：
密码设置：1.八位以上,大小写字母、数字、符号组合
　　2.要容易记忆
　　3.定期更换
防火墙禁用、SELINUX 禁用。

常识：I.Debian 系列和 Red hat 系列最大的区别：软件安装方式不同。
II.用户登录成功后，系统缺省的主提示符，普通用户为“$”; 超级用户 root 为 “#”。
III.用户退出系统的方法有两种：
1.键入命令：exit 或者 logout
2.在提示符下按 Ctrl+D 组合键
IV.SCSI 硬盘、SATA 硬盘、USB：sd
第一块硬盘：sda （第一分区：sda1，第二分区：sda2 ...）
第二块硬盘：sdb
IDE 硬盘：hd
第一块硬盘：hda
第二块硬盘：hdb
了解：windows 命名规则：8.3　文件名不超 8 位，后缀名 3 位。

知识点总结

- Linux 系统安装流程

- Linux 文件系统结构及分区设置
- Wmware 虚拟机软件应用
- 远程登录工具应用

练习

- 安装虚拟机软件 VMware
- 在 VMware 中安装 Linux 系统
- 设置 VMware 网络，使虚拟机 linux 可与本机 windows 网络连接

练习题

1、安装虚拟机软件
2、建立新的虚拟机安装 CentOS 5.5
3、设置网络，使用 Windows 客户端连接 linux （添加回环网卡）

- **LINUX 中所有的东西都是以文件的形式存在的。**
- **X window 是一个协议不是一个图形环境。**

3.16.1 Linux 常用命令

文件命名规则（linux 文件无后缀名规则、也没有长度限制。）
a. 除了/之外，所有的字符都合法。
b. 有些字符最好不用，如空格符、制表符、退格符和字符@#$&()-等。
（用 ″ ″来规则空格符）
c. 避免使用.作为普通文件名的第一个字符。(以. 开头的文件都是隐藏文件)
d. 大小写敏感，大小写严格区分:文件名、命令、命令选项、配置文件选项。

命令格式：命令 -选项 参数
说明：当有多格选项时，可以写在一起。
两个特殊的目录.和..，分别代表当前目录和当前目录的父目录。

查看 ifconfig （interface config） ipconfig
lo 回环地址 127.0.0.1
eth0 网卡名称 eth（ethernet） 第一块 0 第二块 1
inet addr ： IP 地址
HWaddr ： MAC 地址
更改 IP 地址： ifconfig eth0 新 IP 地址

1.文件处理命令 ls

命令名称：ls

命令英文原意：list
命令所在路径：/bin/ls
执行权限：所有用户
功能描述：显示目录文件
语法： ls 选项[-ald] [文件或目录]
-a （all）显示所有文件，包括隐藏文件
-l （long)详细信息显示

d rwxr-xr-x 2 root root 4096 mar 15 21:12 desktop
文件类型
d-目录 directory
- 二进制文件（命令、配置文件、图片、视频、网页、压缩包）
l-软链接文件 link
文件权限（以文件为单位，每个文件把用户分成三类）
rwx r-x r--
所有者 所属组 其他人
u g o
user group others

r－read 读
w－write 写
x－execute 执行

ls -l 第二个部分是：硬连接数
第三个部分是：所有者
第四个部分是：所属组
第五个部分是：文件大小（KB）针对目录的统计是不准确的，只统计了目录本身的大小
第六个部分是：文件创建时间或最后修改的时间
第七个部分是：文件名

ls -d （directory ）查看目录属性
ls -i 查看文件 i 节点

2.文件处理命令 cd

命令名称：cd
命令英文原意：change dirctory
命令所在路径：shell 内置命令
执行权限：所有用户
语法：cd [目录]

功能描述：切换目录
范例：$ cd / 切换到跟目录
$ cd .. 回到上一级目录

3.文件处理命令 pwd

命令名称：pwd
命令英文原意：print working directroy
命令所在路径：/bin/pwd
执行权限：所有用户
语法：pwd
功能描述：显示当前所在的工作目录
范例：$ pwd
/etc/rc5.d

4.文件处理命令 mkdir

命令名称：mkdir
命令英文原意：make direcrories
命令所在路径：/bin/mkdir
执行权限：所有用户
语法：mkdir [目录名]
功能描述：创建新目录
范例：$ mkdir newdir

5.文件处理命令 touch

命令名称：touch
命令所在的路径：/bin/touch
执行权限：所有用户
语法：touch [文件名]
功能描述：创建空文件
范例：$ touch newfile

6.文件处理命令 cp

命令名称：cp
命令英文原意：copy
命令所在路径：/bin/cp
执行权限：所有用户
语法：cp -R [源文件或目录] [目的目录]
-R 复制目录
功能描述：复制文件或目录
范例：$ cp file1 file2 dir1
将文件 file1、file2 复制到目录 dir1
$ cp -R dir1 dir2
将 dir1 下的所有文件及子目录复制到 dir2

7.文件处理命令 mv

命令名称：mv
命令英文原意：move
命令所在的路径：/bin/mv
执行权限：所有用户
语法：mv [源文件或目录] [目的目录]
功能描述：移动文件、更名
范例：$ mv file1 file3
将当前目录下的文件 file1 更名为 file3
$mv file2 dir2
将文件 file2 移动到目录 dir2 下

8.文件处理命令 rm

命令名称：rm
命令英文原意：remove
命令所在路径：/bin/rm
执行权限：所有用户
语法：rm -r [文件或目录]
-r 删除目录
功能描述：删除文件
范例：$ rm file3
删除文件 file3
$ rm -r dir1
删除目录 dir1
rm -f （force） 强制删除（不确认）
-i 提示确认

9.文件处理命令 cat

命令名称：cat
命令英文原意：concatenate and display files
命令所在路径：/bin/cat
执行权限：所有用户
语法：cat [文件名]
功能描述：显示文件内容
范例：$ cat /etc/issue
$ cat /etc/services

10.文件处理命令 more

命令名称：more
命令所在路径：/bin/more
执行权限：所有用户
语法：more [文件名]

（空格）或 f 显示下一页
（Enter） 显示下一行
Q 或 q 退出
功能描述：分页显示文件内容
范例：$ more /etc/services

11.文件处理命令 ln

命令名称：ln
命令英文原意：link
命令所在路径：/bin/ln
执行权限：所有用户
语法：ln -s [源文件] [目标文件]
-s 创建软链接
功能描述：产生链接文件
范例： $ ln -s /etc/issue /issue.soft
创建文件/etc/issue 的软链接/issue.soft
$ ln /etc/issue /issue.hard
创建文件/etc/issue 的硬链接/issue.hard

软链接文件： 类似于 Windows 快捷方式，只是一个符号链接
1、文件类型 l 、权限为 rwxrwxrwx
2、比较小，符号链接
3、创建时间
4、指向原文件 －－－>

硬链接文件： 拷贝+同步更新 （增加 i 节点映射）
1、原文件删除，硬链接依然可以访问
2、硬链接只能针对文件不能针对目录
3、硬链接不能跨分区

补充：echo "内容" >> 文件名 * 向文件末尾追加内容

12.权限管理命令 chmod

命令名称：chmod
命令英文原意：change thepermissions mode of a file
命令所在路径：/bin/chmod
执行权限：所有用户
语法：chmod [{ugo}{+-={rwx}] [文件或目录]
[mode=421] [文件或目录]
功能描述：改变文件或目录权限
范例： $ chmod g+w file1
赋予文件 file1 所属组写权限
$ chmod 777 dir1

设定目录 dir1 为所有用户具有全部权限

chmod u + r
g - w
o = x
a

chmod u+w chmod g-x chmod a+r chmod u+w,g-wx,o= /test

r=4 w=2 x=1

"=" 后面没有 为无权限

rwx r-x r-- 754 、 631 rw- -wx --x

补充：su - 用户名 【切换用户】

文件 r—cat、more w—vi x—command、script

目录 r—ls w—touch、mkdir、rm x—cd

文件权限总结

代表字符	权限	对文件的含义	对目录的含义
r	读权限	可以查看文件内容	可以列出目录中的内容
w	写权限	可以修改文件内容	可以在目录中创建、删除文件
x	执行权限	可以执行文件	可以进入目录

13.权限管理命令 chown

命令名称：chown

命令英文原意：change file ownership

命令所在路径：/bin/chown

执行权限：所有用户

语法：chown [用户] [文件或目录]

功能描述：改变文件或目录的所有者

范例: $ chown nobody file1

改变文件 file1 的所有者为 nobody

14. 权限管理命令 chgrp

命令名称：chgrp

命令英文原意：change file group ownership

命令所在路径：/bin/chgrp

执行权限：所有用户

语法：chgrp [用户组] [文件或目录]

范例：$ chgrp adm file1

改变文件 file1 的所属组为 adm

15. 权限管理命令 umask

命令名称：umask
命令所在路径：Shell 内置命令
执行权限：所有用户
语法：umask [-S]
　　-S 以 rwx 形式显示新建文件或目录缺省权限
功能描述：显示、设置文件的缺省权限
范例： $ umask
　　$ umask -S
0022 权限掩码值
0-特殊权限位
022 ----w--w-

777
- 022

755

750 rwxr-x--- umask (777-750) umask 027

16. 文件搜索命令 which

命令名称：which
命令所在路径：/usr/bin/which
执行权限：所有用户
语法：which [命令名称]
功能描述：显示系统命令所在目录
范例：$ which ls

查看命令所在目录
1、which 还会列出别名信息
2、whereis 还会帮助文档位置

17. 文件搜索命令 find

命令名称：find
命令所在路径：/usr/bin/find
执行权限：所有用户
语法：find [搜索范围] [匹配条件]
功能描述：查找文件或目录
范例： $ find /etc -name init
　　在目录/etc 中查找文件 init
　　$ find / -size +204800

在根目录下查找大于 100MB 的文件

$ find /home -user cooper

在目录/home 下查找所有者为 cooper 的文件

$ find /etc -cmin -5

在/etc 下查找 5 分钟内被修改过的属性的文件和目录

$ find /etc -size +163840 -a -size -204800

在/etc 下查找大于 80MB 小于 100MB 的文件

$ find /etc -name inittab -exec ls -l {} \;

在/etc 下查找 inittab 文件并显示其详细信息

补充：

find 搜索范围越小越好，匹配条件越精准越好

根据文件名查找 -name * 匹配任意字符 ？匹配单个字符

根据文件大小查找 -size （单位：数据块，1 个数据块＝512 字节） 数据块是最小的数据存储单位

100M ＝ 102400K ＝ 204800

+n 大于多少 n 等于多少 -n 小于多少

根据文件所有者查找 -user

根据时间值查找 time（天） min（分钟）

-atime -amin a-access 访问时间

-ctime -cmin c-change 改变文件属性（ls -l ，权限、所有者、所属组……）

-mtime -mmin m-modify 改变文件内容

根据文件类型查找 -type

f 二进制文件

l 软链接文件

d 目录

-a 逻辑与 两个条件都符合

-o 逻辑或 两个条件符合一个即可

-exec 命令 {} \;

-ok （询问确认）

{} 代表 find 查找的结果

\ 转义符（让它后面的命令或符号使用其本身的含义）

; 结束

find -inum 根据 i 节点查找

number

18. 文件搜索命令 grep

命令名称：grep

命令所在路径：/bin/grep

执行权限：所有用户

语法： grep [指定字串] [文件]

功能描述：在文件中搜寻字串匹配的行并输出
范例：# grep ftp /etc/services

- **Linux 中很多设置必须写入配置文件才能永久生效。**
- **每个文件必须有一个 i 节点。**
- **内核操作的对象必须用数字标识**
- **权限规定：缺省的二进制文件不能具有可执行权限（x）644 [755 －－ 644]**

常识：远程登陆服务：SSH（数据加密）
TELNET 不建议使用，因为它的密码明文传输。
文件 abc 隐藏:. abc

3.17.1 Linux 常用命令（II）

19. 帮助命令 man

命令名称：man
命令英文原意：manual
命令所在路径：/usr/bin/man
执行权限：所有用户
语法：man [命令或配置文件]
功能描述：获得帮助信息
范例： $ man ls
查看 ls 命令的帮助信息
$ msn services
查看配置文件 services 的帮助信息
/etc/passwd
帮助类型：1-命令 5-配置文件
只查看命令选项： 命令 --help

20. 帮助命令 info

命令名称：info
命令英文原意：information
命令所在路径：/usr/bin/info
执行权限：所有用户
语法：info [任何关键字]
功能描述：获得帮助信息
范例：$ info ls
查看 ls 指令的帮助信息

21. 帮助命令 help

命令名称：help
命令所在路径：Shell 内置命令
执行权限：所有用户
语法：help 命令
功能描述：获得 Shell 内置命令的帮助信息
范例： $ help umask
查看 umask 命令的帮助信息
查看 Shell 内置命令的帮助 help

压缩解压： *.gz *.tar *.zip *.bz2

.zip Linux 与 Windows 通用的格式
.bz2 压缩比很高

22. 压缩解压命令 gzip

命令名称：gzip
命令英文原意：GNU zip
命令所在路径：/bin/gzip
执行权限：所有用户
语法：gzip 选项 [文件]
功能描述：压缩文件
压缩后文件格式：.gz
注意：
1、不保留原文件
2、不能压缩目录

23. 压缩解压命令 gunzip

命令名称：gunzip
命令英文原意：GNU unzip
命令所在路径：/bin/gunzip
执行权限：所有用户
语法：gunzip 选项 [压缩文件]
功能描述：解压缩.gz 的压缩文件
范例：$ gunzip file1.gz

24. 压缩解压命令 tar

命令名称：tar
命令所在路径：/bin/tar
执行权限：所有用户
语法：tar 选项[cvf] [目录]
-c 产生.tar 打包文件
-v 显示详细信息
-f 指定压缩后的文件名
-z 打包同时压缩
功能描述：打包目录
压缩后文件格式：.tar.gz
范例：$ tar -zcvf dir1.tar.gz dir1
将目录 dir1 压缩成一个打包并压缩的文件
tar 命令解压索语法：
-x 解包.tar 文件
-v 显示详细信息
-f 指定解压文件
-z 解压缩
范例：$ tar -zxvf dir1.tar.gz

25. 压缩解压命令 zip

命令名称：zip
命令所在路径：/usr/bin/zip
执行权限：所有用户
语法：zip 选项[-r] [压缩后文件名称] [文件或目录]
-r 压缩目录
功能描述：压缩文件或目录
压缩后文件格式：.zip
范例：$ zip services.zip /etc/services
压缩文件
$ zip -r test.zip /test
压缩目录

26. 压缩解压命令 unzip

命令名称：unzip
命令所在路径：/usr/bin/unzip
执行权限：所有用户
语法：unzip [压缩文件]
功能描述：解压缩.zip 的压缩文件
范例：$ unzip test.zip

27.压缩解压命令 bzip2

命令名称：bzip2
命令所在路径：/usr/bin/bzip2
执行权限：所有用户
语法：bzip2 选项 [-k] [文件]
-k 产生压缩文件后保留原文件
功能描述：压缩文件
压缩后文件格式：.bz2
范例：$ bzip2 -k file1

28.压缩解压命令 bunzip2

命令名称：bunzip2
命令所在路径：/usr/bin/bunzip2
执行权限：所有用户
语法：bunzip2 选项 [-k] [压缩文件]
-k 解压缩后保留原文件
功能描述：解压缩
范例：$ bunzip2 -k file1.bz2

write、wall 使用必须是在线用户

29.网络通信指令 write

指令名称：write
指令所在路径：/usr/bin/write
执行权限：所有用户
语法：wall <用户名>
功能描述：向另外一个用户发信息，以 Ctrl+D 作为结束
范例：$ write cooper

30.网络通信指令 wall

指令名称：wall
指令所在路径：/usr/bin/wall
执行权限：所有用户
语法：wall [message] [文件名]
功能描述：向所有用户广播信息
范例：$ wall Hi,everybody!

31.网络通信命令 ping

命令名称：ping

命令所在路径：/bin/ping
执行权限：所有用户
语法：ping 选项 ip 地址
-c 指定发送次数
-s 指定数据包大小
功能描述：测试网络连通性
范例：# ping 192.168.2.2

补充：
ping 不通不一定代表网络不通 ICMP 请求包
1、防火墙
2、arp 地址错误
……
排查
1、ping 对方主机 IP 地址
2、ping 本机的 IP 地址 * 证明本机网卡问题
3、ping 回环地址（localhost/127.0.0.1） * 证明 TCP/IP 协议问题

32. 网络通信命令 ifconfig

命令名称：ifconfig
命令所在路径：/sbin/ifconfig
执行权限：root
语法：ifconfig 网卡名称 ip 地址
功能描述：查看和设置网卡信息
范例：# ifconfig eth0 192.168.2.2

重启网络服务 /etc/rc.d/init.d/network restart

33. 系统关机命令 shutdown

命令名称：shutdown
命令所在路径：/sbin/shutdown
执行权限：root
语法：shutdown
功能描述：关机
范例：# shutdown -h now

34. 系统关机命令 reboot

命令名称：reboot
命令所在路径：/sbin/reboot
执行权限：root
语法：reboot
功能描述：重启系统

范例：# reboot

Shell 应用技巧

① bash 应用技巧

命令补齐

命令补齐允许用户输入文件名起始的若干字母后，按<Tab>键补齐文件名。

命令历史

命令历史允许用户浏览先前输入的命令并重新调用它们，用 history 命令可以显示命令列表，按↑和↓可以查找以前执行过的命令。

② 命令别名

命令别名定义：
范例： alias copy=cp
alias xrm="rm -r"
查看别名信息：alias
删除别名：unalias copy
alias 别名=命令 or "命令组合"

③ 输入/输出重定向

同标准 I/O 一样，Shell 对于每一个进程预先定义 3 个描述字（0、1、2）。分别对应于：
0 （STDIN）标准输入； 【计算机中：键盘】
1 （STDOUT)标准输出； 【计算机中：显示器】
2 （STDERR）标准错误输出。 【计算机中：显示器】
> 或 >> 输入重定向
范例：ls -l /tmp > /tmp.msg
date >> /tmp.msg
< 输入重定向
范例：wall < /etc/motd
2> 或 2>> 错误输出重定向
范例：cp -R /usr /backup/usr.bak 2> /bak.error

输出重定向 > 、>> 追加 1
输入重定向 < 0
错误输出重定向 2> 、2>> 追加 2

④ 管道

管道：将一个命令的输出传送给另一个命令，作为另一个命令的输入。
使用方法：

命令 1 | 命令 2 | 命令 3 ..... | 命令 n

范例：

```
ls -l /etc | more
ls -l /etc | grep init
ls -l /etc | grep init | wc -l   [wc -l 统计文件行数]
```

⑤ 命令连接符

;
用; 间隔的各命令按顺序依次执行。
&&
前后命令的执行存在逻辑与关系，只有&&前面的执行成功后，它后面的命令才被执行。
||
前后命令的执行存在逻辑或关系，只有||前面的命令执行失败后，它后面的命令才被执行。

逻辑与

命令 1	&&	命令 2
1		1
0		0

逻辑或

命令 1	||	命令 2
1		0
0		1

⑥ 命令替换符

命令替换：将一个命令的输出作为另一个命令的参数。
格式为：命令 1 `命令 2`
范例：

```
ls -l `which touch`
```

知识点总结

Linux 文件命名规则

文件处理命令

Linux 权限；i 节点；软硬链接的区别及特点；对文件的创建、删除、拷贝、剪切、查看

权限管理命令

文件和目录的 rwx 权限的含义；修改权限；修改文件所有者、所属组；默认权限查看及设置。

文件搜索命令

查看命令路径、find 用法及实例、文件内容查找

帮助命令

查看命令和配置文件的帮助、Shell 内置命令的帮助

压缩解压命令

.gz、.tar.gz、.zip、.bz2，压缩命令的特点

网络通信命令

ping 的用法、ifconfig、用户通信

关机重启命令

Shell 应用技巧

命令补齐、历史记录、快捷键、命令别名、输入输出重定向、管道、命令连接符、命令替换符、转义符。

练习

- 练习操作所有命令
- 重点练习：
  权限练习
  软硬链接文件练习
  find 查找练习
  命令技巧练习

练习题

1、用 root 用户登录 linux，创建目录/perm ，在/ perm 目录下创建文件 newfile，授予/perm 目录所有用户都有 rwx 权限；创建普通用户 testuser，切换到 testuser 执行“rm /perm/newfile”是否可以执行
2、在 root 目录下创建文件 newfile2，移动文件 newfile2 到/perm 目录下同时改名为 file01；改变/perm/file01 文件的所有者为系统用户 adm，改变其所属组为系统用户组 games；改变/perm/file01 文件权限为“rwxrw-r--”；在/perm 目录下，分别给 file01 生成一个软连接文件 file01.soft 和一个硬连接文件 file01.hard；删除/perm 目录
3、查找命令 ifconfig 的绝对路径并判断此命令哪些用户可以执行；更改本机 IP 地址为 192.168.9.250（联系后改变回来）
4、查看用户配置文件/etc/passwd 的帮助信息；查看 cd 的帮助信息
5、在/etc 目录下查找 5 分钟内被改变过内容的文件；在/boot 目录下查找文件名为 grub.conf 的文件并同时列出文件的详细信息；在跟目录查找系统中大于 100MB 小鱼 150MB 的文件
6、创建目录/comp，拷贝文件/etc/services 到 /comp 目录下，分别对 services 文件进行压缩 ，生成.gz .zip .bz2 三种格式的压缩包；拷贝目录/etc 到/comp 目录下，把 etc 目录压缩成 etc.tar.gz，把 service 文件所有压缩包使用 rm 删除 （只用一条 rm 命令，非执行三次 rm 操作）；在 /comp 目录下创建文件 hidefile，并设置为隐藏
7、设置命令“cp -R”的命令别名为 dircp，把当前 linux 系统时间广播给所有在线用户（使用命令替换符）
8、依次把命令 date 执行的结果写入文件 /root/cmd.msg ； 之后再把命令 wc -l /etc/passwd 执行结果追加到文件 /root/cmd.msg
9、查看/etc 目录的详细信息（权限、大小等）；查看 /etc 目录下文件的详细信息时实现分页浏览；查看 /etc 目录下文件名包含.conf 的文件有多少个；统计 /etc 目录下有多少个子目录
10、ping 本机地址测试，要求发送 10 次 ICMP 包且包大小为 1000byte

- **Linux 默认的 shell：bash （echo $SHELL）**
- **系统中安装的 shell ：/etc/shells （/sbin/nologin 不可用，伪用户使用）**
- **who 查看在线用户 （终端：tty 本地终端 pts 远程终端）**
- **ctrl+l 清屏 ，等同于 clear 命令**
- **ctrl+u 清除光标前字符**
- **ctrl+c 终止命令执行**

- 左键选择复制、右键粘贴
- **mail** 查看和发送邮件

3.17.2 Vim/Vi 文本编辑器

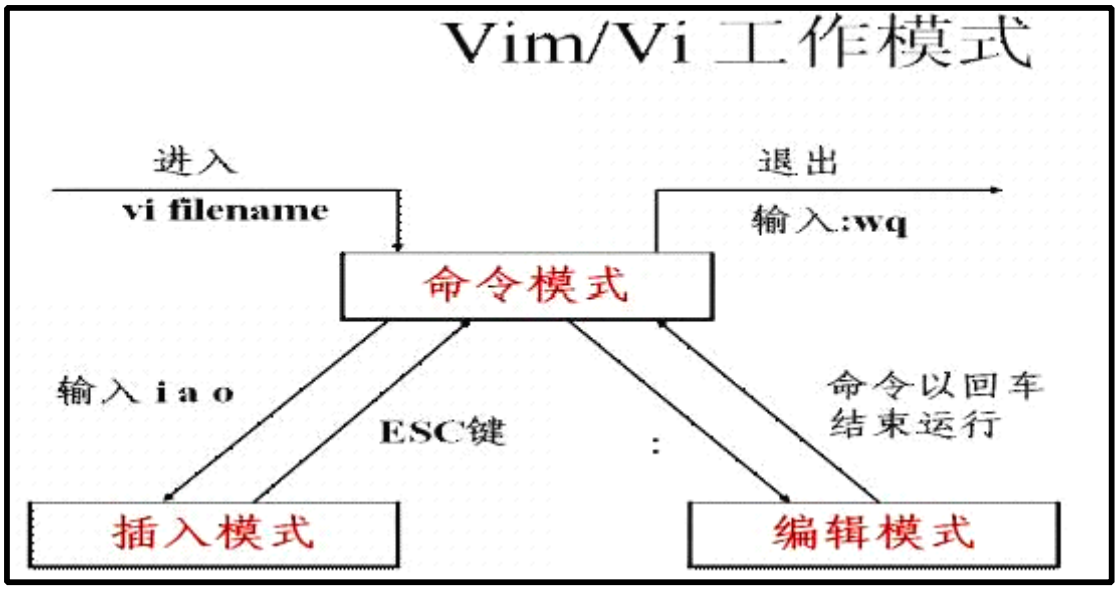
Vim/Vi 没有菜单，只有命令。

★ vi 的模式（命令模式、插入模式、编辑模式）

命令模式：

在我们刚刚通过 vi 新建或打开一个已经存在的文件时，首先默认被读取的模式就是“命令模式”，命令模式的特征就是，在编辑器窗口左下角的位置上没有任何的提示标语。

在这个模式下，如果不通过相应的命令，我们无法输入新的文本到这个文件中。确切说 vi 命令模式下的命令，只是计算机键盘上的某个按键或某些按键的组合，当我们在命令模式下敲下键盘一个按键的时候，相应命令就已经被执行。



插入命令

命令	作用
a	在光标所在字符后插入
A	在光标所在行末插入
i	在光标所在字符前插入
I	在光标所在行首插入
o	在光标下插入新行
O	在光标上插入新行

定位命令

命令	作用
h, 方向左键	左移一个字符
j, 方向下键	下移一行
k, 方向上键	上移一行
j, 方向右键	右移一个字符
$	移至行尾
0	移至行首
H	移至屏幕上端

M	移至屏幕中央
L	移至屏幕下端
:set nu	设置行号
:set nonu	取消行号
gg	到第一行
G	到最后一行
nG	到第 n 行
:n	到第 n 行

删除命令

命令	作用
x	删除光标所在处字符
nx	删除光标所在处后 n 个字符
dd	删除光标所在行，ndd 删除 n 行
dG	删除光标所在行到文件末尾内容
D	删除光标所在处到行尾内容
:n1,n2d	删除指定范围的行

复制和剪切命令

命令	作用
yy、Y	复制当前行
nyy、nY	复制当前行以下 n 行
dd	剪切当前行
ndd	剪切当前行以下 n 行
p、P	粘贴在当前光标所在行下或行上

替换和取消命令

命令	作用

r	取代光标所在处字符
R	从光标所在处开始替换字符，按 Esc 结束
u	取消上一步操作

搜索和替换命令

命令	作用
/string	向前搜索指定字符串，搜索时忽略大小写:set ic
n	搜索指定字符串的下一个出现位置
:%s/old/new/g	全文替换指定字符串
:n1,n2s/old/new/g	在一定范围内替换指定字符串

保存和退出命令

命令	作用
:w	保存修改
:w new_filename	另存为指定文件
:wq	保存修改并退出
ZZ	快捷键，保存修改并退出
:q!	不保存修改退出
:wq!	保存修改并退出（文件所有者及 root 可使用）

应用实例

- 导入命令执行结果 :r !命令
- 定义快捷键 :map 快捷键 触发命令
- 范例：:map ^P I#<ESC>
- :map ^B 0x
- 连续行注释 :n1,n2s/^/#/g
- :n1,n2s/^#//g
- :n1,n2s/^\/\//g
- 替换 :ab coopermail 493630800@qq.com

补充：# 开头表示注释

Ctrl + P 自动跳到光标所在行行首 #

Ctrl + v　　Ctrl + p　　/　　Ctrl + v + p

^ 表示行首

Vi 配置文件 ~/.vimrc (~用户宿主目录 root---/root　　cooper---/home/cooper)

【写的是编辑模式命令】

知识点总结

- Vim/Vi 工作模式
- Vim/Vi 基本使用
- 插入、定位、删除、复制、剪切、替换、取消、搜索、保存、退出
- Vim/Vi 应用技巧

练习

- 熟悉 Vim/Vi 工作模式切换
- 熟练 Vim/Vi 常用操作
- 练习 Vim/Vi 定义快捷键、连续行替换、导入命令执行结果等技巧
- 设定 Vim/Vi 配置文件

练习题

1、创建目录 /vi.test，拷贝文件 /etc/inittab 到 /vi.test 目录下，使用 vi 编辑 /vi.tets/inittab 文件，把其中的第 23 行到 29 行注释（使用连续行注释命令）

2、编辑 /vi.test/inittab 文件，把第 10 行到 20 行之间的字符串 init 都替换为 boot ；并且在文件末尾导入当前编辑文件的时间

3、定义快捷键 Ctrl+e 可以输入兄弟连网址 bbs.lampbrother.net 并自动返回命令行模式

4、设置 vim 配置文件，使每次使用 vi 编辑任何文件都自动显示行号；并定义在 vi 中是要输入 mymail 就自动转换为 admin@lampbrother.net

5、编辑 /vi.test/inittab 文件，定位到第 35 行，光标跳到行尾添加“www.lampbrother.net”，然后复制此行到第 40 行后

6、删除 /vi.test/inittab 文件的第 1 行到第 20 行

7、编辑 /vi.test/inittab 定位到屏幕下端，把光标所在处字符替换为“x”，查找字符串“NOT”，找到后替换为 “YES”，保存退出

3.18.1 Linux 引导流程解析

系统引导流程

固件 firmware （CMOS/BIOS） → POST 加电自检

↓

自举程序 BootLoader （GRUB） → 载入内核

↓

载入内核 Kernel → 驱动硬件

↓

启动进程 init

↓

读取执行配置文件/etc/inittab

GRUB 载入内核

1、驱动硬件

2、启动引导进程 init

init 是系统启动的第一个进程 PID 恒为 1 ，是系统所有进程的父进程

父进程终止子进程自然终止

1、父进程终止，子进程没有终止，子进程成为孤儿进程

系统会将孤儿进程的父进程指向 init

2、子进程终止，父进程没有切断进程联系，子进程成为僵尸进程 Zombie

僵尸进程手动终止

PPID parent process ID

PID=0 内核调度器 作用：分配 CPU 时间、进程间切换

了解：Windows：NTLDR 、配置文件 boot.ini

软硬件时钟设置

软件时钟（系统时钟） date

硬件时钟（CMOS 时钟） hwclock

```
[root@localhost ~]# date
Thu Mar 17 03:25:39 CST 2011
[root@localhost ~]# hwclock
Thu 17 Mar 2011 03:25:53 AM CST  -0.048293 seconds
[root@localhost ~]#
[root@localhost ~]# hwclock --hctosys
[root@localhost ~]#
[root@localhost ~]# date
Thu Mar 17 03:26:43 CST 2011
[root@localhost ~]#
[root@localhost ~]# hwclock --set --date="3/19/2011 14:06:00"
[root@localhost ~]# hwclock
Sat 19 Mar 2011 02:06:12 PM CST  -0.627060 seconds
[root@localhost ~]#
[root@localhost ~]# hwclock --help_
```

init 的工作

init 启动后读取 inittab 文件，执行缺省运行级别，从而继续引导过程。在 linux 系统中，init 是第一个可以在存在的进程，它的 PID 恒为 1，但它也必须向一个更高级的功能负责：PID 为 0 的内核调度器（Kernel scheduler），从而获得 CPU 时间。

```
[root@localhost ~]# ps -le | head -n 5
F S   UID   PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S     0     1     0  0  78   0 -   518 -      ?        00:00:02 init
1 S     0     2     1  0 -40   - -     0 migrat ?        00:00:00 migration/0
1 S     0     3     1  0  94  19 -     0 ksofti ?        00:00:00 ksoftirqd/0
5 S     0     4     1  0 -40   - -     0 watchd ?        00:00:00 watchdog/0
```

inittab 文件剖析

运行级的切换：

1.查看当前运行级 [详见：附 运行级别]

```
[root@localhost ~]# runlevel
N 3
```

2.运行级的切换

#inti [0123456Ss]

```
[root@localhost ~]# init 1
INIT: Sending processes the TERM signal
Shutting down smartd:                                      [  OK  ]
Shutting down Avahi daemon:                                [  OK  ]
Stopping yum-updatesd:                                     [  OK  ]
Stopping atd:                                              [  OK  ]
Stopping cups:                                             [  OK  ]
Stopping hpiod:                                            [  OK  ]
Stopping hpssd:                                            [  OK  ]
Shutting down xfs:                                         [  OK  ]
Shutting down console mouse services:                      [  OK  ]
```

在 inittab 中，所有条目采取以下格式：

id : run-levels : action : process

id：标识符，一般为两位字母或数字

run-levels：指定运行级别，可以指定多个

action：指定运行状态、要求

process：指定要运行的脚本、命令

action 常用取值：

initdefault: 指定系统缺省启动的运行级别

sysinit：系统启动任何运行级别都执行 process 中指定命令

wait：执行 process 中执行的命令，并等其结束再运行其他命令

once：执行 process 中指定的命令，不等待其结束

ctrlaltdel：按下 Ctrl + Alt + Del 时执行 process 指定的命令
powerfail：当出现电源错误时执行 process 指定的命令，不等待其结束
powerokwait：当电源恢复时执行 process 指定的命令
respawn：一旦 process 指定的命令中止，便重新运行该命令

```
id:3:initdefault:
```

指定系统默认运行级别为 3，如果想系统启动后自动运行 X Window，则将上面的 3 改为 5.

```
# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit
```

启动脚本 /etc/rc.d/rc.sysinit，完成系统服务程序启动，如系统环境变量设置、设置系统时钟、加载字体、检查加载文件系统、生成系统启动信息日志文件等。

```
l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6
```

判断默认运行级别，调用 /etc/rc.d/rc 脚本，执行相应运行级别目录中的服务程序，完成相应运行级别的初始化设置。

/etc/rc.d/init.d

该目录下包含各个运行级别的服务程序脚本

```
[root@localhost ~]# ls /etc/rc.d/init.d
acpid                dovecot     killall         nscd              single
anacron              dund        krb524          ntpd              smartd
apmd                 firstboot   kudzu           oddjobd           smb
atd                  functions   lvm2-monitor    pand              spamassassin
auditd               gpm         mcstrans        pcscd             squid
autofs               haldaemon   mdmonitor       portmap           sshd
avahi-daemon         halt        mdmpd           psacct            syslog
avahi-dnsconfd       hidd        messagebus      rawdevices        tcsd
bluetooth            hplip       microcode_ctl   rdisc             tux
capi                 hsqldb      multipathd      readahead_early   vncserver
conman               httpd       named           readahead_later   vsftpd
cpuspeed             ibmasm      netconsole      restorecond       wdaemon
crond                innd        netfs           rpcgssd           winbind
cups                 ip6tables   netplugd        rpcidmapd         wpa_supplicant
cups-config-daemon   iptables    network         rpcsvcgssd        xfs
dc_client            irda        NetworkManager  rwhod             xinetd
dc_server            irqbalance  nfs             saslauthd         ypbind
dnsmasq              isdn        nfslock         sendmail          yum-updatesd
```

/etc/rc.d/rc[0123456].d

分别存放对应于运行级别的服务程序脚本的符号连接，连接到 init.d 目录中的相应脚本

```
[root@localhost ~]# ls /etc/rc.d/rc3.d
K01dnsmasq          K35winbind          S04readahead_early  S26acpid
K02avahi-dnsconfd   K50ibmasm           S05kudzu            S26apmd
K02NetworkManager   K50netconsole       S08ip6tables        S26haldaemon
K02oddjobd          K50tux              S08iptables         S26hidd
K05conman           K50vsftpd           S08mcstrans         S28autofs
K05innd             K69rpcsvcgssd       S09isdn             S50hplip
K05saslauthd        K73ypbind           S10network          S55sshd
K05wdaemon          K74nscd             S11auditd           S56cups
K10dc_server        K74ntpd             S12restorecond      S56rawdevices
K10psacct           K85mdmpd            S12syslog           S56xinetd
K10tcsd             K87multipathd       S13cpuspeed         S80sendmail
K12dc_client        K87named            S13irqbalance       S85gpm
K15httpd            K88wpa_supplicant   S13portmap          S90crond
K20nfs              K89dund             S14nfslock          S90xfs
K20rwhod            K89netplugd         S15mdmonitor        S95anacron
K24irda             K89pand             S18rpcidmapd        S95atd
K25squid            K89rdisc            S19rpcgssd          S97yum-updatesd
K30spamassassin     K91capi             S22messagebus       S98avahi-daemon
K35dovecot          K99readahead_later  S25bluetooth        S99firstboot
K35smb              S00microcode_ctl    S25netfs            S99local
K35vncserver        S02lvm2-monitor     S25pcscd            S99smartd
[root@localhost ~]# _
```

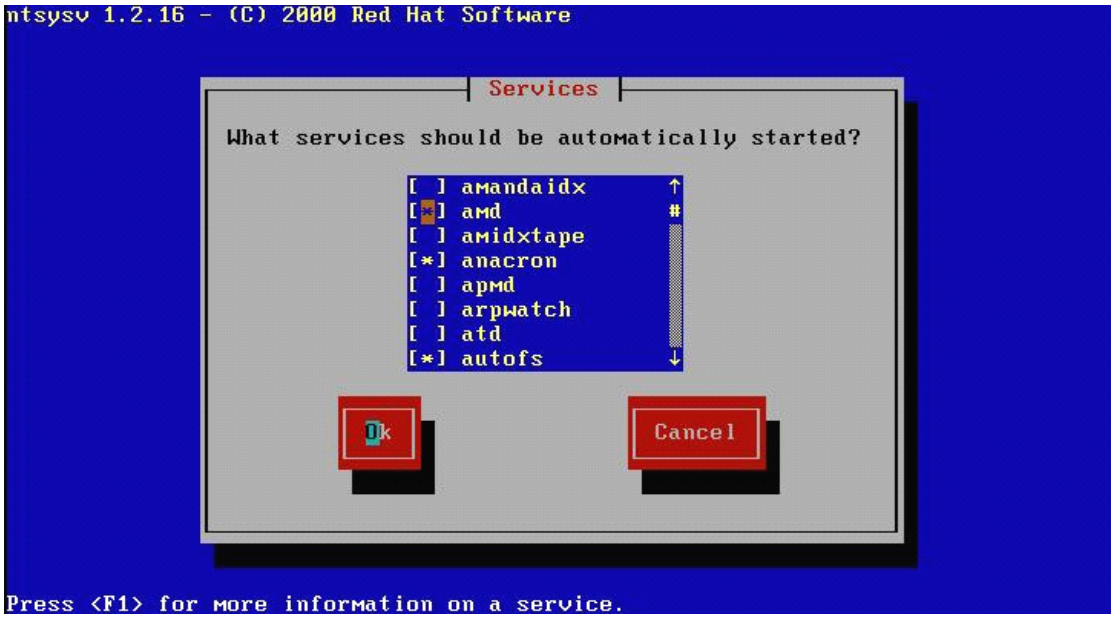
设置自启动程序

- ls -s
- chkconfig
- ntsysv

查看服务在各个运行级别启动状态
chkconfig --list 服务名称
chkconfig --level 运行级别 服务名称 on/off

```
[root@localhost ~]# ln -s /etc/rc.d/init.d/ntpd /etc/rc.d/rc3.d/S100ntpd
[root@localhost ~]# ls -l /etc/rc.d/rc3.d/S100ntpd
lrwxrwxrwx 1 root root 21 Mar 17 05:32 /etc/rc.d/rc3.d/S100ntpd -> /etc/rc.d/ini
t.d/ntpd
[root@localhost ~]# chkconfig --help
chkconfig version 1.3.30.2 - Copyright (C) 1997-2000 Red Hat, Inc.
This may be freely redistributed under the terms of the GNU Public License.

usage:   chkconfig --list [name]
         chkconfig --add <name>
         chkconfig --del <name>
         chkconfig [--level <levels>] <name> <on|off|reset|resetpriorities>
[root@localhost ~]#
[root@localhost ~]# chkconfig --list named
named           0:off   1:off   2:off   3:off   4:off   5:off   6:off
[root@localhost ~]# chkconfig --level 3 named on
[root@localhost ~]# chkconfig --list named
named           0:off   1:off   2:off   3:on    4:off   5:off   6:off
[root@localhost ~]# _
```

ntsysv --level 运行级别



相关命令及日志

❖ 运行 dmesg 检查引导期间的错误

```
[root@localhost ~]# dmesg | grep hda
    ide0: BM-DMA at 0x10c0-0x10c7, BIOS settings: hda:pio, hdb:pio
[root@localhost ~]# dmesg | grep eth0
eth0: registered as PCnet/PCI II 79C970A
eth0: link up
[root@localhost ~]# _
```

❖ 检查系统日志 /var/log/messages，查找可能被 dmesg 忽略的应用程序错误

```
[root@localhost ~]# grep syslogd /var/log/messages
Mar 16 04:55:51 localhost syslogd 1.4.1: restart.
Mar 16 05:01:29 localhost syslogd 1.4.1: restart.
Mar 15 21:07:37 localhost syslogd 1.4.1: restart.
Mar 15 21:10:48 localhost syslogd 1.4.1: restart.
Mar 15 21:15:39 localhost syslogd 1.4.1: restart.
Mar 16 02:48:15 localhost syslogd 1.4.1: restart.
Mar 16 04:29:35 localhost syslogd 1.4.1: restart.
Mar 16 22:30:07 localhost syslogd 1.4.1: restart.
Mar 17 00:36:13 localhost syslogd 1.4.1: restart.
Mar 17 05:15:55 localhost syslogd 1.4.1: restart.
Mar 17 05:17:42 localhost syslogd 1.4.1: restart.
[root@localhost ~]# _
```

GRUB

GRUB 的配置文件默认为

/boot/grub/grub.conf

ls -l /etc/grub.conf （为链接文件）

GRUB 配置：
/etc/grub.conf ----> /boot/grub/grub.conf
/etc 配置文件
/boot 启动程序文件

版本号：2.6.18
6－偶数正式版，奇数测试版

/etc/inittab tab-table

GRUB 配置选项

default 定义缺省启动系统
timeout 定义默认等待时间 （值为 0 则不等待）
splashimage 定义 GRUB 界面图片
hiddenmenu 隐藏菜单
title 定义菜单项名称
root 定义系统内核所在分区
kernel 指定内核文件所在位置
initd 指定镜像文件所在位置

单用户模式应用

案例：root 密码忘记

进入单用户模式重新设置 root 密码，方法如下：
开机进入 GRUB 界面，按 e 进入编辑模式，选中 kernel 行，再次按 e 键，在行后输入 1，回车保存后按 b 键引导后，即可进入单用户模式。

***进入单用户模式不需要密码**

设置 GRUB 密码

1、生成 GRUB 密码
grub-md5-crypt
password:
1JyPZ00$X5ZX0JQfLiErYIlM0qHUB.
2、写入 GRUB 配置文件

passwd --md5 1JyPZ00$X5ZX0JQfLiErYIlM0qHUB.
--md5 参数表示口令使用 MD5 加密。

GRUB 命令模式修复

启动后进入 grub 界面但没有菜单，只剩下一个 grub>提示符，错误解决方法：
grub>cat /boot/grub/grub.conf（为查看错误原因）
grub>boot (hd0,0)
grub>kernel(hd0,0)/vmlinuz-2.6.18-194.e15 ro root=LABEL=/
grub>initrd(hd0,0)/initrd-2.6.18-14.img
grub>boot

Linux 光盘修复模式

进入光盘修复模式
1、把安装盘放到光驱，然后重新启动机器，在 BIOS 中把系统设置为光驱引导。
2、在 boot 提示符下输入：linux rescue
进入修复模式

修复：先尝试单用户模式-->光盘修复模式

/ --- /mnt/sysimage
/etc/inittab --- /mnt/sysimage/etc/inittab

***光盘修复模式下的原系统根分区被挂载到光盘 linux 的/mnt/sysimage 的目录下。**

附：运行级别

0－关机
1－单用户模式（类似 Windows 安全模式） 1）没有图形界面 2）默认只能 root 登录
2－多用户模式（不启动 NFS），没有图形界面
NFS-Network File System ，网络文件系统，实现 Linux/UNIX 系统之间的文件共享 RPC/UDP
3－多用户模式，没有图形界面
4－自定义
5－图形界面的多用户模式 X X11 X11R6 xfree86 ＝X Window
6－重启

runlevels 为空，表示所有运行级别都要执行 process

切换运行级别：init 运行级别　　s/S－single，单用户模式
查看运行级别：runlevel　　前面：之前运行级别　后面：当前运行级别

initdefault　定义系统的缺省运行级别
　　|
/etc/rc.d/rc.sysinit　启动系统服务程序
　　|
/etc/rc.d/rc　判断缺省运行级别，启动对应目录下的服务程序
　　|
/etc/rc.d/rcN.d　　N=运行级别　　/etc/rcN.d
　　|
/etc/X11/prefdm　启动 Xwindow

S55sshd
S-启动 start　改名　s55sshd
K-关闭 kill　切换运行级别
55－优先级 ，数字越小优先权越大
sshd－服务名称

/etc/rc.d/init.d　服务启动脚本存放目录
start 启动　stop 关闭　restart 重启　reload 重新加载配置文件
condrestart 重启（重启前确认下进程是否启动）　status 服务状态

service 服务名称 start/stop ……

知识点总结

- 掌握 Linux 引导过程的每个步骤
- 了解固件设置，掌握软硬件时钟设置及同步
- 掌握使用帮助命令的方法和习惯
- 掌握 GRUB 的配置文件及原理
- 掌握 Linux 运行级别知识
- 掌握 inittab 文件的引导设置及相关知识点
- 掌握如何设置服务自启动及手工启动
- 掌握如何判断引导期间硬件加载及服务启动是否出现错误
- 掌握如何进入 Linux 单用户模式进行修复
- 了解如何设置 GRUB 密码
- 掌握使用 GRUB 命令行操作修复启动
- 掌握如何使用 Linux 光盘修复模式

firmware（CMOS/BIOS）－－检测硬件　* 软硬件时钟的设置和同步
　　|
MBR-BootLoader 自举程序　* GRUB 配置、内核存放目录
　　|
　kernel　－－驱动硬件 dmesg
　　|
　 init　 PID=1 、父子进程的关系
　　|
 /etc/inittab　* inittab 文件格式
　　|
　initdefault　* 运行级别有几种代表什么含义
　　|
/etc/rc.d/rc.sysinit
　　|
 /etc/rc.d/rc
　　|
 /etc/rc.d/rcN.d　* 如何设置一个服务不自启动　/etc/rc.d/init.d　手动启动关闭服务
/var/log/messages
　　|
 /etc/X11/prefdm

(hd0,0) ===　hda1

练习

- 设置及同步软硬件时钟
- 试验更改缺省运行级别并通过 GRUB 命令方式或 Linux 系统修复方式更正
- 练习更改服务自启动及手工启动
- 假设 root 密码忘记进入单用户模式更改
- 设置 GRUB 密码
- 假设 root 密码和 GRUB 密码都忘记的情况，进如 Linux 系统修复模式更改

练习题

1、修改 linux 缺省启动运行级别为 3；设置服务 sshd 只在系统运行级别 3 自启动，其他运行级别不启动
2、修改系统的硬件时钟为 2011 年 2 月 26 日 8 时 18 分 18 秒，然后通过硬件时钟同步软件时钟为此时间值
3、关闭系统中正在运行的 sendmail 服务
4、切换 linux 当前的运行级别为 1 （单用户模式）
5、假设 root 密码忘记，无法使用 root 登录系统，在 linux 启动时进入单用户模式修改 root

密码
6、假设 linux 中添加了第二块 SCSI 硬盘，简述新硬盘的检测识别流程
7、简述 linux 的系统引导流程及每个启动步骤的作用
8、设置 GRUB 密码
9、把 grub.conf 文件的 title 删掉，重启 linux 后，系统进入 GRUB 命令行模式，练习通过 GRUB 命令进入系统后修复
10、创建/backup 目录，备份 /etc/inittab 文件到 /backup，删除/etc/inittab 文件，重新启动 linux 后系统无法引导，练习使用光盘修复模式恢复系统

3.21.1 Linux 软件包管理

一、RPM 包管理

二进制包＝RPM 包（RedHat）　　Debian DEB 包

RPM 软件包的一个例子：
　　Sudo-1.7.2pl-5.el5.i386.rpm
其中包括软件名（sudo），版本号（1.7.2pl），发行号（5.el5）,和硬件平台（i386）.
i386 i686 alpha ppc

（1）卸载
卸载 rpm -e 软件名
　　#rpm -e sudo
注意：如果其他软件包有依赖关系，卸载时会产生提示信息，可使用--nodeps 强行卸载。
强行卸载（忽略依赖关系）rpm -e --nodeps 软件名
卸载 1、关闭进程　2、rpm -e 卸载

（2）安装
　　#rpm -ivh sudo-1.7.2pl-5.el5.i386.rpm
安装 -i　显示详细信息 -v　显示进度条 -h
rpm -i --prefix＝安装路径
挂载光盘：
mkdir /mnt/cdrom
mount /dev/cdrom /mnt/cdrom

/dev/sda1　/home
/dev/cdrom　软链接，原文件/dev/hdc ，光盘修复模式不能/dev/cdrom
/mnt　存放临时文件系统（光盘、U 盘）的挂载点
mount 物理设备名 挂载点
/mnt/cdrom/CentOS 存放了 RPM 包　　/mnt/cdrom/RedHat/PRMS

查询：
rpm -q sudo

rpm -qa | grep samba
查询软件是否安装和软件包信息　rpm -q 软件名　　rpm -qa　查询所有软件包
查询软件包安装文件 rpm -ql

安装选项：
--test
只对安装进行测试，并不实际安装
--replacepkgs
覆盖安装
--replacefiles
文件冲突时替换安装

安装文档那个中示例或文档信息，一般名为 example　sample
etc 配置文件　bin/sbin　命令　libexec/lib　库文件　doc　文档　man 帮助　var 临时文件

（3）升级
#rpm -Uvh sudo-1.8.0pl-5.el5.i386.rpm
升级 -U

安装、下载、升级：yum
查询、校验、从 RPM 包提取文件：rpm

一．一 YUM 包管理

运用 yum 的好处
- 自动解决软件包依赖关系
- 方便的软件包升级

- ✧ 安装 yum install
- ✧ 检测升级 yum check-update
- ✧ 升级 yum update
- ✧ 软件包查询 yum list
- ✧ 软件包信息 yum info
- ✧ 卸载 yum remove
- ✧ 帮助 yum --help 、man yum

保存默认 yum 源配置文件 /etc/yum.repos.d/CentOS-Base.repo

修改 yum 源为安装光盘

#vi /etc/yum.repos.d/Centos-Media.repo 编辑配置文件

[c5-media]
name=CentOS-$releasever - Media
baseurl=file:///mnt/cdrom
# file:///media/cdrom
# file:///media/cdrecorder/
gpgcheck=1
enabled=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-5
#mv /etc/yum.repos.d/CentOS-Base.repo .. 移至其他位置

（4）查询

选项：

-a 查询所有已安装的软件包

-f 查询文件所属软件包

-p 查询软件包

-i 显示软件包信息

-l 显示软件包中所有文件

-d 显示文档文件

-c 显示配置文件

RPM 查询应用案例：

1、查询文件隶属的软件包：rpm -qf

2、查询软件包信息：rpm -qi、rpm -qip

3、查询软件包安装文件：rpm -ql、rpm -qlp

4、查询软件包帮助文档：rpm -qd、rpm -qdp

5、查询软件包配置文件：rpm -qc、rpm -qcp

（5）校检

#rpm -V 软件名称

- 5 文件的 md5 校检值
- S 文件大小
- L 连接文件
- T 文件时间值
- D 设备文件
- U 文件所有者
- G 文件所属组
- M 文件权限

missing 文件丢失

（6）软件包文件提取

解压所有文件到当前目录

rpm2cpio initsripts-8.45.30-2.el5.centos.i386.rpm | cpio idv

解压指定文件到当前目录

rpm2cpio initscropts-8.45.30-2.el5.centos.i386.rpm | cpio idv ./etc/inittab

提取文件　rpm2cpio RPM 包 | cpio -id 文件名
i-提取 d-保持提取文件目录结构

切换到根目录 cd /
文件提取恢复
rpm2cpio /mnt/cdrom/CentOS/initscripts-8.45.30-2.el5.centos.i386.rpm | cpio -idv ./etc/inittab

光盘修复模式通过 RPM 包提取恢复 inittab 文件
1、改变目录结构为原有硬盘 Linux 结构
chroot /mnt/sysimage
2、挂载光盘
mount /dev/hdc /mnt/cdrom　　* 使用光盘原始设备名非软链接
3、提取恢复文件
rpm -qf /etc/inittab
cd /
rpm2cpio /mnt/cdrom/CentOS/initscripts-8.45.30-2.el5.centos.i386.rpm | cpio -idv ./etc/inittab
4、exit 退出 chroot 状态
exit 退出修复模式

二、源代码包安装

应用举例（proftpd）：
tar -xzvf proftpd-1.3.3d.tar.gz （解压解包）
cd proftpd-1.3.3d
./configure --prefix=/usr/local/proftpd（配置）
make（编译）
make install （安装）
软件下载地址：http://www.proftpd.org

1、源代码包适用性比较强，适用几乎所有 UNIX/Linux 系统和所有硬件平台
2、定制性强
3、软件分发新版本都是先发源代码包

安装步骤
1、解压解包
2、配置　./configure --prefix=指定安装目录　（放在/usr/local）
./configure --help　查看 configure 支持的选项
（搜集系统信息，主要生成 makefile 文件）　gcc、make 编译工具已经安装
3、编译 make　（源代码编译成可执行文件）
4、安装 make install　（把编译的可执行文件拷贝并设置权限）

源代码包卸载：1、关闭进程　2、删除安装目录　　rm -rf /usr/local/proftpd

三、脚本安装

应用举例（webmin）
tar -xzvf webmin-1.530.tar.gz
cd webmin-1.530
vi README
./setup.sh
软件下载地址：http://www.webmin.com

脚本安装：没有 configure 脚本，查看 README 或 INSTALL 文件

四、DEB 包管理

- 查看软件包内容 dpkg -L （类似 rpm -ql）
- 查看软件包详细信息 dpkg -l （类似 rpm -qi）
- 安装软件包 dpkg -i （类似 rpm -i）
- 卸载已安装软件包 dpkg -r （类似 rpm -e）
- 查看软件包信息 apt-cache show （yum info）
- 安装 apt-get install （yum install）
- 删除 apt-get remove （yum remove）
- 更新已安装包 apt-get upgrade （yum update）

知识点总结

- 软件包管理的思路与方法
- Redhat 系 Linux 软件包管理
  - 二进制包管理：RPM、YUM
  - 源代码包管理
  - 脚本安装
- Debian 系 Linux 软件包管理

练习

- 二进制软件包安装、升级、卸载
- 二进制软件包查询
- 二进制软件包校检
- 应用案例：
- /etc/inittab 文件恢复、vim 配置文件注释
- 源代码包安装练习

练习题

1、查询 /etc/rc.d/init.d/sendmail 文件隶属于哪个 RPM 包
2、卸载 squid 软件包，分别通过安装光盘重新安装 squid、通过 yum 源安装 squid
3、查看系统安装包 samba 软件的信息及安装了哪些文件系统中、并查询 samba 的配置文件有哪些
4、执行安装光盘为 yum 源，并安装 gcc
5、练习安装源代码包 proftpd
6、练习脚本安装 webmin
7、升级软件包 ntp
8、删除系统配置文件/etc/inittab ， 进入光盘修复模式， 通过 RPM 包提取源文件进行恢复
9、执行命令 “chmod 777 /usr/sbin/visudo”、“rm /etc/sudoers”后校检 sudo 软件包，查看结果后恢复 sudo 软件包原有权限设置及文件

3.22.1 Linux 用户管理

用户管理配置文件

- 用户信息文件：/etc/passwd
- 密码文件：/etc/shadow
- 用户组文件：/etc/group
- 用户缺省配置文件：
- /etc/login.defs
- /etc/default/useradd
- 新用户配置文件目录：/etc/skel
- 登录信息文件：/etc/motd

/etc/passwd 文件格式

/etc/passwd
root:x:0:0:root:/root:/bin/bash
用户名:密码位:UID:缺省组的 GID:注释:宿主目录:命令解释器

密码加密：md5、DES 单向不可逆
UID：UID 为 0 是管理员
组：如果添加用户没有指定所属组，自动创建一个和用户同名的组 * 不推荐
* 组按功能划分（webadmin）或者按组织结构划分（project01）
* 每个用户可以属于多个组

字段	含义
用户名	用户登录系统时使用的用户名
密码	密码位
UID	用户标识号
GID	缺省标识号
注释性描述	例如存放用户全名等信息
宿主目录	用户登录系统后的缺省目录
命令解释器	用户使用的 Shell,默认为 bash

shell：cat /etc/shells

用户类型

Linux 用户分为三种：

- 超级用户 （root,UID=0）
- 普通用户 （UID 500-60000）
- 伪用户 （UID 1-499)

伪用户

1、伪用户与系统和程序服务相关
bin、deamon、shutdown、halt 等，任何 Linux 系统默认都有这些伪用户
mail、news、games、apache、ftp、mysql 及 sshd 等，与 Linux 系统的进程相关
2、伪用户通常不需要或无法登陆系统
3、可以没有宿主目录

用户组

- 每个用户都至少属于一个用户组
- 每个用户组可以包括多个用户
- 同一个用户组的用户享有该组共有的权限

/etc/shadow 文件格式

字段	含义
用户名	用户登录系统时使用的用户名
密码	加密密码
最后一次修改时间	用户最后一次修改密码的天数
最小时间间隔	两次修改密码之间的最小天数
最大时间间隔	密码保持有效的最多天数
警告时间	从系统开始警告到密码失效的天数
帐号闲置时间	帐号闲置时间
失效时间	密码失效的绝对天数
标志	一般不使用

useradd---passwd
　　|
/etc/passwd
　　|　　　　pwconv 同步密码
/etc/shadow

pwunconv　回写密码 （UNIX 不提供）

passwd -d 与用户/etc/shadow 密码清空一样 ： 用户登录时不需提供密码验证

/etc/motd　今日消息，登录成功都会显示此信息

用户管理命令

chage 设定密码
　　-l　查看用户密码设置
　　-m　密码修改最小天数
　　-M　密码修改最大天数
　　-d　密码最后修改的日期
　　-I　密码过期后，锁定账户的天数
　　-E　设置密码的过期日期，如果为 0，代表密码立即过期；如果为-1，代表密码永不过期
　　-W　设置密码过期前，开始警告的天数

/etc/group 文件格式

字段	含义
组名	组的名称，建议按功能或组织来划分和命名
组密码	一般不使用
GID	组标识号
组内用户列表	属于该组的所有用户列表

/etc/group 用户组信息文件
组密码用途：当非组内成员切换到该组时使用

手工添加用户

- 分别在 /etc/passwd、/etc/group 和/etc/shadow 文件中添加一笔记录
- 创建用户宿主目录（注意宿主目录所有者）
- 向用户宿主目录中拷贝默认的配置文件（/etc/skel）

1、添加用户信息 /etc/passwd （等同于 useradd）
2、添加用户密码 /etc/shadow （等同于 passwd）
3、创建宿主目录（改变所有者为普通用户）

添加用户

useradd 设置选项 用户名 -D 查看缺省参数
u : UID
g : 缺省所属用户组 GID
G : 指定用户所属多个组
d : 宿主目录
s : 命令解释器 Shell
c : 描述信息
e : 指定用户失效时间
passwd cooper

useradd
-u UID
-g default GID *

-G others group　　#
-d home directory
-s shell
-c comment　　*
-e expire　　# 指定帐目失效时间　-e 20110404

useradd　userdel　usermod
groupadd　　groupdel　groupmod
del-delete
mod-modify

Solaris ：　useradd 设置 -m 自动创建

SetUID

思考：为什么普通用户可以更改密码？
SetUID 的定义：当一个可执行文件具有 SetUID 权限，用户执行这个文件（命令或脚本）时，将以这个文件所有者的身份执行。
范例：1、将 touch 命令授予 SetUID 权限
　　2、当 vi 命令被授予 SetUID 权限
　　3、查找 SetUID 程序：
　　find / -perm -4000 -o -perm -2000

SetUID
Linux 中默认所有的命令所有者都是 root

设置 SetUID(4)权限　chmod u+s　　chmod 4755
　　u-s　　755
SetGID(2)
当一个可执行文件具有 SetGID 权限，用户执行这个文件（命令或脚本）时，将以这个文件所属组的身份执行。
chmod g+s　　chmod 2755
　　g-s　　755

-perm 指定权限　　-perm -0777 -a -type f　* 查找权限为 777 的文件
-perm -4000 -o -perm -2000
0 代表不限制

用户组管理命令

- 添加用户组 groupadd
- groupadd -g 888 webadmin
- 创建用户组 webadmin ，其 GID 为 888

- 删除用户组: groupdel 组名
- 修改用户组信息 groupmod
- groupmod -n apache webadmin
- 修改 webadmin 组名为 apache
- gpasswd 设置组密码及管理组内成员
- -a 添加用户到用户组
- -d 从用户组中删除用户
- -A 设置用户组管理员
- -r 删除用户组密码

删除用户组：
1、哪些文件或目录所属组是这个组 find / -group 组名
2、组内成员

gpasswd(Only Linux) 设置组密码 gpasswd 组名 /etc/gshadow

gpasswd -a 用户名 组名
/etc/gshadow 用户名:密码:管理员:组内成员

修改用户信息

usermod
- usermod -G softgroup cooper
- 将用户 cooper 添加到 softgroup 用户组中
- usermod -l austin -d /home/austin -g lampbrother cooper
- 将用户 cooper 的登录名改为 austin，加入到 lampbrother 组中，用户目录改为/home/austin

usermod 选项同 useradd
1、把用户加到一个用户组中 usermod -G 组名 用户名 UNIX/Linux
2、usermod -G 组名（指定多个组，加多个组名逗号分隔） 用户名

删除用户

userdel -r 用户名
-r ：删除用户目录

手工删除：
使用 find 命令查找属于某个用户或用户组的文件 find 选项-user、-uid、-group、-gid
1、对需要保留的文件进行移动和备份
2、对不需要的文件进行删除
3、清除文件中的相关表项
4、清除用户的宿主目录

禁用和恢复用户

禁用

usermod -L username

passwd -l username

恢复

usermod -U username

passwd -u username

用户授权实例

授权用户 jack 和 mary 对目录 /software 有写权限

groupadd softadm

usermod -G softadm jack

gpasswd -a mary softadm

chgrp softadm /software

chmod g+w /software

ls -d /software

drwxrwxr-w 2 root softadm 512 jul 14 06:17 /software

grep softadm /etc/group

Softadm ::100:jack,mary

一个用户授权对目录 w　　chown 改变所有者

多个用户授权对目录 w

原理：改变目录所属组，把用户加入到所属组，对所属组授权 w

案例：限制用户 **su** 为 **root**

groupadd sugroup

chmod 4550 /bin/su

chgrp sugroup /bin/su

ls -l /bin/su

-r-sr-x--- 1 root sugroup 18360 Jan 15 2010 /bin/su

设定后，只有 sugroup 组中的用户可以使用 su

useradd helen

passwd helen

usermod -G sugroup helen

用户管理命令

- pwck 检测 /etc/passwd 文件
- vipw 编辑 /etc/passwd 文件 （锁定文件）
- id 查看用户 id 和组信息（缺省）
- finger 查看用户详细信息
- su 切换用户 （su - 环境变量切换）
- passwd -S 查看用户密码状态
- who 查看当前登录用户信息
- grpck 用户组配置文件检测
- vigr 编辑 /etc/group 文件 （锁定文件）
- groups 查看目录隶属于哪些用户组
- newgrp 切换用户组

pwck （password check）检测用户文件 /etc/passwd /etc/shadow
grpck （group check） 检测用户组文件 /etc/group /etc/gshadow * 检测组内成员是否有效

vipw = vi /etc/passwd + 锁定文件
vigr = vi /etc/group + 锁定文件

finger 查看用户信息（用户最后登录信息、邮件信息、计划任务信息）

su 环境变量不变
su - 环境变量改变

sudo 授权

-授权普通用户以 root 身份执行命令
-命令范围可限定至具体选项和参数

配置文件：/etc/sudoers，编辑配置文件命令 visudo ，普通用户使用命令 sudo。
格式：用户组 （组名） 主机地址=命令 （绝对路径）

sudo
设置：visudo （类似 vipw）
用户名（%组名） 主机地址（本机地址，IP）=授权命令（多个命令逗号分隔，写绝对路径）
用户执行：sudo 命令 -l

案例：授权 Apache 服务器，让普通用户可以管理
1、启动关闭 apache
2、更新网页

3、修改配置文件

1、/etc/rc.d/init.d/httpd start/reload/status/fullstatus/configtest （检测语法错误）
2、/var/www/html 改变所有者 chown test /var/www/html
3、vi /etc/httpd/conf/httpd.conf

知识点总结

- Linux 用户管理概念及原理
- Linux 用户管理相关配置文件
- Linux 用户管理常用命令
- 特殊权限 SetUID
- 管理员授权

练习

- 手工添加删除用户及用户组
- 管理员授权练习
- 用户及用户组授权练习
- SetUID 权限操作练习
- 常用用户及用户组操作命令练习

练习题

1、添加用户组 webadmin，然后添加用户 neil，要求设置用户 neil 的缺省组为 webadmin、指定其描述信息为“Marker xiaoshenyang”，设置其密码为 lampbrother，在成功添加用户 neil 后再将其加入用户组 root
2、查看 neil 用户的基本信息（包括宿主目录、使用 Shell 等），并查看 neil 用户都隶属于哪些用户组
3、把 neil 的用户名改为 admin ， 并设置 admin 具有和管理员一样的权限
4、锁定 admin 不允许其登录 linux
5、把用户 admin 从用户组 root 中删除，在 linux 中删除用户 admin（不保留其宿主目录）
6、添加用户 david，并在 linux 中设置只有 david 可以执行 su 命令切换为 root
7、手动添加用户 helen
8、设置命令 vi（/usr/bin/vim）具有 SetUID 权限，切换到用户 helen 使用 vi 编辑/etc/shadow 把 david 用户密码删除，尝试用 david 登录查看效果
9、授权用户可以管理 apache 服务器
10、修改用户登陆后提示信息为“Welcome to www.lampbrother.net”
11、创建文件共享目录 /share，授权用户 david 和 helen 对 /share 目录有读取写权限

3.23.1 Linux 进程管理

进程和程序的区别

1、程序是静态概念，本身作为一种软件资源长期保存；而进程是程序的执行过程，它是动态概念，有一定的生命期，是动态产生和消亡的。
2、程序和进程无一一对应关系。一个程序可以由多个进程公用；另一方面，一个进程在活动中可顺序地执行若干个程序。

父进程与子进程

1、子进程是由一个进程所产生的进程，产生这个子进程的进程称为父进程。
2、在 Linux 系统中，使用系统调用 fork 创建进程。Fork 复制的内容包括父进程的数据和堆栈段以及父进程的进程环境。
3、父进程终止子进程自然终止。

父进程终止子进程自然终止
1、父进程终止，子进程没有正常终止，子进程会变成孤儿进程，将父进程指向 init 进程
2、子进程终止，父进程没有切断进程通信，子进程变成僵尸进程，关闭僵尸进程

前台进程和后台进程

前台进程：

在 Shell 提示处打入命令后，创建一个子进程，运行命令，Shell 等待命令退出，然后返回到对用户给出提示符。这条命令与 Shell 异步运行，即在前台运行，用户在它完成之前不能执行另一个命令。

后台进程：

在 Shell 提示处打入命令，若后随一个&,Shell 创建的子进程运行此命令，但不等待命令退出，而直接返回对用户给出提示。这条命令与 Shell 同步运行，即在后台运行。后台进程必须是非交互式的。

查看用户信息 w

w 显示信息的含义

- load average： 分别显示系统过去 1、5、15 分钟内的平均负载程度。
- FROM：显示用户从何处登录系统，":0.0"的显示代表该用户使用 X Window 下登录，本地终端登录 tty，远程终端登录 pts
- IDLE：用户闲置的时间。这是一个计时器，一旦用户执行任何操作，该计时器便会被重置

- JCPU：以终端代号来区分，该终端所有相关的进程执行时，所消耗的 CPU 时间会显示在这里
- PCPU：CPU 执行当前程序耗费的时间
- WHAT：用户正在执行的操作

查看个别用户信息：w 用户名

up－系统运行时间 uptime
load average 负载程度值，平均值小于 0.8 的负载较轻

查看进程信息 ps

-a all -u user -x xterm -e every -l long

STAT
S-Sleeping
D-DropSleeping
R-Running
Z-Zombie
T-sTopped

常用选项
- a ： 显示所有用户进程
- u ： 显示用户名和启动时间
- x ： 显示没有控制终端的进程
- e ： 显示所有进程，包括没有控制终端的进程
- l ： 显示详细信息

ps 应用实例
ps 查看隶属于自己的进程
ps -u or -l 查看隶属于自己进程详细信息
ps -le or -aux 查看所有用户进程的详细信息
ps -uU sam 查看系统中指定用户执行的进程
ps -le | grep init 查看指定进程信息

ps 常用输出信息的含义
- PID ： 进程号
- PPID ： 父进程的进程号
- TTY ： 进程启动的终端
- STAT ： 进程当前状态
- S 休眠状态，D 不可中断的休眠状态，R 运行状态，Z 僵死状态，T 停止
- NI ： 进程优先级
- TIME ： 进程自从启动以来占用总的 CPU 时间
- COMMAND/CMD ： 进程的命令名

- USER/UID ：用户
- %CPU ：占用 CPU 时间和总时间的百分比
- %MEM ：占用内存与系统内存总量的百分比

kill - 杀死进程

为什么要杀死进程

- 该进程占用了过多的 CPU 时间
- 该进程锁住了一个终端，使其他前台进程无法运行
- 运行时间过长，但没有预期效果
- 产生了过多到屏幕或磁盘文件的输出
- 无法正常退出

kill - 关闭进程

- 关闭进程：kill 进程号
- kill -9 进程号 （强行关闭）
- kill -1 进程号 （重启进程）
- 关闭图形程序：xkill
- 结束所有进程：killall
- 查找服务进程号：pgrep 服务名称
- 关闭进程：pkill 进程名称

kill -l 查看 kill 信号
killall （Linux Only）

/proc 存放在内存镜像中 -process
1、系统当前运行的进程信息
2、系统信息（硬件信息）
cpuinfo、meminfo、partitions

nice 和 renice

nice NI

- 指定程序的运行优先级
- 格式：nice -n command
- 例如：nice --5 myprogram

renice

- 改变一个正在运行的进程的优先级

- 格式：renice n pid
- 例如：renice -5 777

优先级取值范围为（-20，19）

nohup

使进程在用户退出登陆后仍旧继续执行，nohup 命令将执行后的数据信息和错误信息默认储存到文件 nohap.out 中
格式：nohup program &

进程的挂起和恢复

进程的挂起（暂停）和终止
挂起（Ctrl + Z）（暂停）
终止（Ctrl + C）
进程的恢复
恢复到前台继续运行（fg）
恢复到后台继续运行（bg）
查看挂起及后台执行进程（jobs）

top

作用：进程状态显示和进程控制，默认每秒自动刷新一次（动态显示）
u ： 查看指定用户的进程
k ： 终止执行中的进程
r ：重新设置进程优先级
d ：改变刷新时间间隔
h or ？ ：获得帮助

top （Linux） ＝ ps 、kill 、renice 、w * 占用系统资源较多
子命令 u－user ＝ ps -uU 用户名
k－kill ＝ kill -15 -9 -1
r－renice ＝ renice
d－delay 改变刷新周期
W－write 保存设置
q－quit 退出

计划任务

为什么要设置计划任务
计划任务的命令

- at　　安排作业在某一时刻执行一次
- batch　安排作业在系统负载不重时执行一次
- cron　安排周期性运行的作业

一次性计划任务 at batch
周期性计划任务 crontab

at 命令的功能和格式

功能：安排一个或多个命令在指定的时间运行一次
at 的命令格式及参数

- at [-f 文件名] 时间
- at -d or atrm 删除队列中的任务
- at -l or atq 查看队列中的任务

at 命令指定时间的方式

绝对计时方法

- midnight noon teatime
- hh:mm [today]
- hh:mm tomorrow
- hh:mm 星期
- hh:mm MMDDYY

相对计时方法

- now + n minutes
- now + n hours
- now + n days

指定在今天下午 17：30 执行某命令（假设在时间是下午 14：30，2011 年 3 月 23 日）。
命令格式如下：

- at 5:30pm
- at 17:30
- at 17:30 today
- at now + 3 hours
- at now + 180 minutes
- at 17:30 032311

at 命令使用范例

交互方式
    at 9:00
使用命令文件方式
    1.生成文件 at.script
    2.使用 at 命令
    at -f at.script 9:00 032311
    or
    at < at.script 9:00 032311

at(batch) 时间
    17:30
    9:00 Sun
    9:00 040111
    now +3 minutes
at> 命令（绝对路径）
at> ctrl+d 保存退出

查看计划任务 atq at -l
删除计划任务 atrm at -d
存放目录：/var/spool/at
确保 atd 进程启动 /etc/rc.d/init.d/atd start

at 配置文件

作用：限制哪些用户可以使用 at 命令
    /etc/at.allow
    /etc/at.deny

at 的执行结果和配置文件

如果/etc/at.allow 文件存在，那么只有列在此文件中的用户才可以使用 at 命令；若/etc/at.allow 文件不存在，则检查/etc/at.deny 文件是否存在。若/etc/at.deny 存在，则在此文件中列出的用户都不能使用 at 命令；如果两个文件都不存在，则只有超级用户可以使用 at 命令；如果两个文件都存在而且均为空，则所有用户都可以使用 at 命令。

batch 命令

作用：安排一个或多个命令在系统负载较轻时运行一次（一般情况下负载较轻平均负载降到 0.8 以下）。

使用方法同 at

crontab 命令格式

作用：用于生成 cron 进程所需要的 crontab 文件
crontab 的命令格式
crontab { -l | -r | -e }

- -l 显示当前的 crontab
- -r 删除当前的 crontab
- -e 使用编辑器编辑当前的 crontab 文件

crontab 文件格式

minute hour day-of-month month-of-year day-of-week commands
其中
minute 一小时中的哪一分钟 [0~59]
hour 一天中的哪小时 [0~23]
day-of-mouth 一月中的那一天 [1-31]
mouth-of-year 一年中的哪一天 [1-12]
day-of-week 一周中的哪一天 [0~6]
commands 执行命令

书写注意事项

选项都不能为空，必须填入，不知道的值使用通配符*表示任何时间
每个时间字段都可以指定多个值，不连续的值使用,间隔，连续的值用-间隔
间隔固定时间执行书写为*/n 格式
命令应该给出绝对路径
用户必须具有运行对应的命令或程序权限

crontab 应用实例
应用范例：
crontab -e
55 1 7 * * 1-5 /usr/bin/wall < /etc/motd
0 18 ** 1-5 /sbin/init 5
0 0 1,15 * 1 /bin/cp -Ru /etc /backup
*/2 12-14 * 3-6,9-12 1-5 /root/test.apache.sh
*crontab 文件建立好以后可以到/var/spool/cron 目录确认

crontab -e edit 窍门：知道的时间写上，不知道的时间写成*

1、每天凌晨两点自动备份/website 目录到/backup
2、备份/website 到/backup，每周日凌晨两点做完全备份，每周二周五凌晨两点做增量备份（cp、tar、scp）
3、吉利大学 web 服务器，定期检测 apache 是否启动没有进程就启动服务，中午 12 点到下午 2 点，每隔 2 分钟
分钟 小时 天 月 星期 命令
0 2 * * * /bin/cp -R /website /backup
0 2 * * 0 /bin/cp -R /website /backup/website_$(date +%Y%m%d) 2>> /backup/website.errlog.total
0 2 * * 2,5 /bin/cp -Ru /website /backup/website 2>> /backup/website.errlog.add ; /bin/date >> /backup/website.errlog.add
*/2 12-14 * 3-6,9-12 1-5 /root/test.apache.sh

cron 配置文件

作用：显示哪些用户可以使用 crontab 命令
- /etc/cron.allow
- /etc/cron.deny

crontab 计划任务存放位置 /var/spool/cron

配置文件 /etc/crontab /etc/rc.d/init.d/crond start

获得命令或配置文件的简短介绍
whatis 命令
apropos 配置文件
makewhatis 定期搜集更新帮助信息

进程处理方式

- standalone 独立运行
- xinetd 进程托管
- atd、crond 计划任务

1、standalone
优点：响应速度快 缺点：占用系统资源多
2、xinetd (unix:inetd)
优点：占用系统资源少 缺点：响应速度慢
/etc/xinetd.d 存放所有托管服务
disable 服务是否启用
server 服务启动脚本绝对路径
socket_type stream(tcp) dgram(udp)

tcp 传输控制协议：三次握手（定时器、断点重发） * 数据传输可靠

A B
-----> SYN 我们可以建立连接 DOS 拒绝服务攻击
<----- ACK/SYN 同意建立连接/等待和你建立连接
-----> ACK 建立连接

udp 用户数据报协议 * 传输速度快
wait no(tcp) yes(udp)
user 指定管理用户
重启 xinetd 进程生效
3、计划任务进程
每隔一分钟醒来一次检测计划任务

知识点总结

- Linux 进程管理基本概念
- Linux 进程管理命令
- 查看进程信息
- 关闭重启进程
- 改变优先级
- 前台后台进程
- 计划任务
- 一次性计划任务
- 周期性计划任务
- 进程处理方式

练习

- 练习服务与进程管理命令
- 尝试模拟企业需求书写计划任务

练习题

1、创建用户 tony 并通过 Windows 客户端工具登录 linux 系统，在 linux 系统中用 root 本地登录后将 tony 用户踢出系统；并在踢出系统后禁用 tony 用户
2、启动 apache 服务（设置优先级为 - 5），查看其进程信息后，关闭 httpd 服务（可用五种方法）
3、用户 tony 在退出登录前，执行命令 find 查找 /tmp 中以 “ssh-”开头的文件，查找结果存放在 tony 宿主目录的 ssh.find 文件中
4、使用 find 查找/usr 目录下以 mkfont 开头的文件，并查找结果保存到/root/find.mkfont 文

件中；执行后挂起（暂停）此命令；然后再调到后台执行，执行完成后查看结果
5、写一条计划任务在 4 月 1 日愚人节当天九点发广播祝福所有人节日快乐
6、写一条计划任务每周二、周五凌晨亮点自动备份系统目录 /etc、/boot 到/backup 目录中，并设置每次计划任务执行出错保存日志记录；另写一条计划任务每周一凌晨两点同步一次软硬件时钟（以软件时钟为参照同步硬件时钟）
7、写一条计划任务周一到周五下午五点半自动关机，关机前半小时自动发广播警告
8、修改 xinetd 托管的 telnet 服务配置文件，启动 telnet 服务

3.24.1 Linux 文件系统管理 （3.25 续）

文件系统构成

- /usr/bin、/bin：存放所有用户可以执行的命令
- /usr/sbin、/sbin：存放只有 root 可以执行的命令
- /home：用户缺省宿主目录
- /proc：虚拟文件系统，存放当前进程信息
- /dev：存放设备文件 device /dev/sda /dev/hdc-/dev/cdrom b-block 块设备 硬盘 c-charset 字符设备 打印机
- /lib：存放系统程序运行所需的共享库
- /lost+found：存放一些系统出错的检查结果
- /tmp：存放临时文件
- /etc：系统配置文件
- /var：包含经常发生变动的文件，如邮件、日志文件、计划任务等 /var/spool/mail /var/log /var/spool/at /var/spool/cron
- /usr：存放所有命令、库、手册页等 ＝C:\windows
- /boot：内核文件及自举程序文件保存位置
- /mnt：临时文件系统的安装点

特殊权限：粘着位 t
粘着位的定义：当权限为 777 的目录被授予粘着位，用户只能在此目录下删除自己是所有者的文件。

常用命令

- 查看分区使用情况：df
- 查看文件、目录大小：du
- 查看文件详细时间参数：stat
- 校检文件 md5 值：md5sum
- 检测修复文件系统：fsck、e2fsck
  （单用户模式卸载文件系统后执行）

df -h 查看分区信息

du -h 查看文件大小 du -sh 统计目录大小
stat modify 文件内容 change 文件属性 （Linux）
md5sum 校验 WinMD5
-c check 检测
md5sum -c /test/tab.md5.20110324 | grep FAILED

init 1
umount /website
fsck -y /website or /dev/sda5 file system check e2fsck -- ext2
-y 自动确认

添加硬盘分区

- 划分分区（fdisk）
- 创建文件系统（mkfs）
- 尝试挂载（mount）
- 写入配置文件（/etc/fstab）

划分分区 （fdisk）

fdisk -l 查看硬盘信息

分区分为三种：主分区、扩展分区、逻辑分区
PC Server 主分区和扩展分区只能有四个
扩展分区并不实际存储数据，是个容器的概念，存储逻辑分区

1、BIOS 查看硬盘信息
2、内核是否驱动硬件 dmesg
fdisk -l
3、划分分区 fdisk 硬盘名称
m 帮助
p 显示分区信息
n 添加分区 +sizeM
t 设置文件系统类型 83 ext3 82 swap
d 删除分区
w 保存退出
q 不保存退出
4、创建文件系统 mkfs -t 指定文件系统类型(ext3/swap) 分区名 = mkfs.ext3
5、创建挂载点尝试挂载
6、写入分区表配置文件 /etc/fstab file system table
LABEL=/ / ext3 defaults 1 1
分区设备名或卷标 挂载点 文件系统类型 设置 是否检测 检测顺序
0/1 0/1/2 （/分区

都是优先检测）

```
[root@localhost ~]# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklab
el
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): m
Command action
   a   toggle a bootable flag
   b   edit bsd disklabel
   c   toggle the dos compatibility flag
   d   delete a partition
   l   list known partition types
   m   print this menu
   n   add a new partition
   o   create a new empty DOS partition table
   p   print the partition table
   q   quit without saving changes
   s   create a new empty Sun disklabel
   t   change a partition's system id
   u   change display/entry units
   v   verify the partition table
   w   write table to disk and exit
   x   extra functionality (experts only)

Command (m for help): _
```

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
SCSI device sdb: 10485760 512-byte hdwr sectors (5369 MB)
sdb: Write Protect is off
sdb: cache data unavailable
sdb: assuming drive cache: write through
SCSI device sdb: 10485760 512-byte hdwr sectors (5369 MB)
sdb: Write Protect is off
sdb: cache data unavailable
sdb: assuming drive cache: write through
Syncing disks.
[root@localhost ~]# fdisk /dev/sdb

Command (m for help): q

[root@localhost ~]# _
```

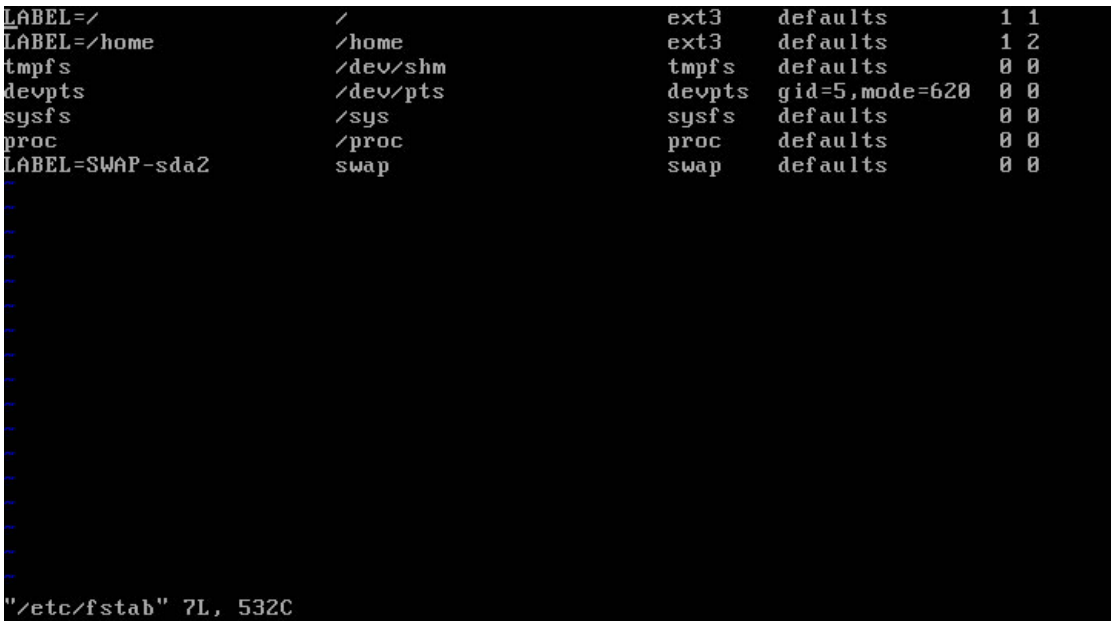
创建文件系统（**mkfs**）

```
[root@localhost ~]# mkfs -t ext3 /dev/sdb1
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
251392 inodes, 502023 blocks
25101 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=515899392
16 block groups
32768 blocks per group, 32768 fragments per group
15712 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 35 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
[root@localhost ~]# _
```

尝试挂载（**mount**）

```
[root@localhost ~]# mkdir /mnt/test
[root@localhost ~]# mount /dev/sdb1 /mnt/test
[root@localhost ~]# mkdir /mnt/test/newdir
[root@localhost ~]# ls -l /mnt/test
total 20
drwx------ 2 root root 16384 Mar 20 03:10 lost+found
drwxr-xr-x 2 root root  4096 Mar 20 03:13 newdir
[root@localhost ~]# _
```

写入配置文件（/etc/fstab）



/etc/fstab 挂载设置

ro 只读
noexec 可执行文件不可运行
nosuid SetUID 命令不可执行
noauto 不自动挂载
acl 启动 ACL 权限控制
　　命令：getfacl 查看、setfacl 设置

权限管理

1、ugo　rwx　基本权限设置　ls -l、chmod
一个用户授权：改变所有者 chown
多个用户授权：改变文件或目录所属组，对组授权，把用户加入组中
rwx 对文件和目录权限的含义
默认创建文件不能有 x 执行权限
2、SetUID、SetGID、粘着位
SetUID 可执行文件　用户会以文件所有者身份执行　　touch/mkdir/vi/kill
粘着位 权限为 777 目录　用户只能删除自己是所有者的文件
3、管理员授权 sudo

授权命令普通用户可以以 root 身份执行、命令具体到选项和参数
apache 授权普通用户管理
4、基于文件系统的权限
defaults：rw, suid, dev, exec, auto, nouser,and async　　man mount　-o
rw 可读写　ro 只读　（备份分区、文件共享分区）
重新挂载分区：mount -o remount 分区名
1)、永久有效
vi /etc/fstab　　defaults,ro
2)、临时有效
mount -o remount,ro/rw 分区名
suid SetUID 命令可执行　nosuid　SetUID 命令不生效　（webserver 网页分区）
exec 可执行文件　noexec　可执行文件不可执行

dev　nodev　是否支持设备文件
auto　noauto　是否自动挂载
nouser user=用户名　挂载的用户身份
async　sync　异步或同步数据
5、访问控制列表 ACL　ls -l 查看权限位 +
getfacl 查看文件或目录的 ACL 权限
setfacl 设置文件或目录的 ACL 权限　setfacl -m user:用户名:权限 文件或目录
group:组名

SWAPFILE 实现

1. # mkdir /var/swap
   # chmod 700 /var/swap
2. # dd if=/dev/zero of=/var/swap/file.swp bs=1024 count=65536
3. # mkswap /var/swap/file.swp
4. # vi /etc/fstab
   /var/swap/file.swp swap swap defaults 0 0
5. free
6. Swapon /var/swap/file.swp
7. Free

swapfile
1、创建 swapfile 存放目录并设置只有 root 可以访问
2、创建指定大小的文件
dd if=/dev/zero　　of=/var/swap/file.swp　bs=1024　　count=65536
导入的文件或设备　生成文件　　数据块大小(1k)　　指定生成文件大小
inputfile　　outputfile　　block size（byte）
伪设备/dev/zero 向输出中写零
伪设备/dev/null 输出消失（黑洞）
3、启用文件 swapfile 功能
mkswap /var/swap/file.swp

4、写入配置文件/etc/fstab
/var/swap/file.swp swap swap defaults 0 0
手动启动 swapfile ： swapon /var/swap/file.swp swaponff
free 查看内存和 swap 空间情况
硬盘对拷 dd if=/dev/sda of=/dev/sdb

添加硬盘 加 swap 分区：
mkswap /dev/sdb2
swapon /dev/sdb2

磁盘配额

1、开启分区配额功能
编辑 /etc/fstab 文件，在挂载属性上加上表示 usrquota 或 grpquota
/dev/sda3 /home ext3 defaults,usrquota 1 2
mount -o remount /home
临时设置：mount -o remount,usrquota /home
2、建立配额数据库（进入单用户模式）
quotacheck -c 创建 -v 详细信息 -u 用户配额 -g 用户组配额 -a 所有分区 -m 强行检测
quotacheck -cvu 分区名
quotacheck -cvuga
会生成 aquota.user 、 aquota.group 两个文件
3、启动配额功能
quotaon 分区名称 quotaon /home quotaoff 关闭配额
4、编辑用户配额
edquota 用户名 edquota -g 用户组名
edquota -t 设置宽限期
复制用户配额
edquota -p 模板用户 复制用户 1 复制用户 2

blocks 空间大小(kb) soft 软限制 hard 硬限制
inodes 文件多少 （i 节点）

quota 命令查看用户的配额情况
管理员查看配额信息： requota -a

创建配额的选项

■ 软限制（Soft limit）：定义用户可以占用的磁盘空间数。当用户超过该限制后会收到超

过配额的警告

- 硬限制（Hard limit）：当用户试图将文件存放在其已经超过该限制值目录时，报告文件系统错误。
- 宽限期（Grace period）：定义用户在软限制下可以使用其文件系统的期限。

系统的潜在威胁

- 系统硬件故障
- 软件故障
- 电源故障
- 用户的误操作
- 人为破坏
- 缓存中的内容没有及时的写入磁盘
- 自然灾害

备份介质的选择

备份介质

硬盘

光盘

磁带

可移动存储设备

一般在选择备份介质时，要从可靠性、速度和介质之间进行权衡

备份策略

完全备份

每隔一段时间对系统进行一次完全的备份，这样在备份时间间隔内，一旦系统发生故障使得数据丢失时，就可以用上一次备份数据到上一次备份时的情况。

增量备份

首先进行一次完全备份，然后每隔一段较短的时间进行一次备份，但是仅仅备份每个短时期内更改的内容。

1、备份

2、备份分区 ro 、umount

3、压缩 bzip2

4、校验 md5sum -c

5、加密 GnuPG PGP 非对称密钥加密

备份的分类

系统备份

实现对操作系统和应用程序的备份

尽量在系统崩溃以后能快速简单完全地恢复系统的运行

主要备份 /etc 、 /boot 、 /var/log 、/usr/local 等

一般只有当系统内容发生变化时才进行

用户备份

实现对用户文件的备份 /home

用户的数据变动频繁

通常采用增量备份策略进行

记录更改建立备份日志

记录系统的更改

记录对系统进行了哪些修改的详细描述及为什么要进行修改

建立备份日志

使用备份日志表格

妥善保管

备份日志表格范例

- 机器名称、IP 地址、存放位置
- 备份时间
- 备份介质及其编号
- 备份的文件系统
- 备份的目录和文件
- 使用的备份命令
- 备份人员及其他

cp 命令备份举例

备份目录

cp -Rpu 备份目录 目标目录

-p 保持备份目录及文件属性

-u 增量备份

远程备份可用 scp

tar 命令使用举例

tar -zcf /backup/etc_20110303.tar.gz /etc
备份 /etc 目录，可同时打包多个目录
tar -zcf backup_user_20110303.tar.gz /etc/passwd /etc/shadow /etc/group /etc/gshadow
对 /etc 目录下指定文件进行备份
tar -ztf backup_user_20110303.tar.gz
查看备份包中文件（不解压）
tar -zxf /backup/etc_20110303.tar.gz
还原 /etc 目录，默认还原到打包文件源目录，-C 可以指定还原目录
tar -zxf backup_user_20110303.tar.gz etc/group
只恢复备份中的指定文件
tar -rf backup_user_20110303.tar /etc/default/useradd /etc/login.defs
将 /etc/default/useradd 、 /etc/login.defs 的内容追加到 backup_user_20110303.tar
tar -uf backup_user_20110303.tar /etc/passwd
将 /etc/passwd 目录中修改过的内容追加到备份文件
* -r 与 -u 选项只能针对 tar 包使用

tar 命令备份举例

为备份文件名添加时间（年月日）
tar -zcf /backup/etc_$(date +%F).tar.gz /etc
添加年月日小时分钟
tar -zcf /backup/etc_$(date +%Y%m%d-%H%M).tar.gz /etc

知识点总结

- Linux 文件系统构成
- 文件系统管理命令
- 粘着位权限
- 如果添加新硬盘
- /etc/fstab 配置文件设置
- swapfile 功能设置
- 磁盘配额设置
- 备份理念及 cp、tar 命令备份应用

练习

- 粘着位试验
- ACL 权限设置
- 添加新硬盘划分分区
- 通过 swapfile 功能增加 swap 空间
- 设置用户磁盘配额
- 备份系统数据结合计划任务应用
- 文件系统操作命令

练习题

1、用 root 用户登录 linux，创建目录/test ，在/test 目录下创建文件 newfile，授予/test 目录所有用户都有 rwx 权限，创建普通用户 testuser，切换到 testuser 执行“rm /test/newfile”是否可以执行；切换回 root 授予 /test 目录粘着位 t 权限，再切换到 testuser 执行“rm /test/newfile”是否可以执行

2、查看根分区使用情况；统计 /usr/local 目录的大小；查看文件 /etc/fstab 的时间属性；生成 /etc 目录下所有以 .conf 结尾的文件的 md5 效验值并尝试修改某一 .conf 结尾文件后效验

3、在 VMware 中添加一块新的 10G 的 SCSI 硬盘，并划分一个 8GB 的分区，设置挂载点为 /apache ，并设置为 linux 启动时自动挂载分区，且设置 /apache 分区可执行文件不可运行及具有 SetUID 命令不可运行；设置用户 zhangsan 和 lisi 对/apache 目录有 rwx 权限，用户 wangwu 和 zhaoliu 有 rw 权限（通过 ACL 权限控制）

4、通过 swapfile 功能增加 256Mswap 空间

5、设置用户 zhangsan 和 lisi 在/home 分区（如在安装系统时无划分/home 分区则在 /分区）中，只能使用 50MB 的磁盘空间

6、编写计划任务，使用 tar 命令每周日凌晨三点做一次完整备份，要求对备份文件标注日期年月日，备份文件存放在/backup 目录下；每周二五凌晨两点对/etc 目录做一次增量备份，备份文件也存放在/backup 目录下

3.28.1 Shell 编程

一个简单的 shell 程序

```
$ cat example
#! /bin/sh
#This is to show what a example looks like.
echo "Our first example"
echo # This insrts an empty line in ouput.
echo "We are currenty in the following directory."
/bin/pwd
```

```
echo
echo "This directory contains the following files"
/bin/ls
```

Shell 结构：

1.#! 指定执行脚本的 shell

2.# 注释行

3.命令和控制结构

创建 shell 程序的步骤：

第一步：创建一个包含命令和控制结构的文件。

第二步：修改这个文件的权限使它可以执行。

使用 chmod u+x

第三步：执行 ./example （也可以使用"sh example" 执行）

Shell 变量

变量：是 shell 传递数据的一种方法，用来代表每个取值的符号名。

Shell 有两类变量：临时变量和永久变量。

临时变量是 shell 程序内部定义的，其使用范围仅限于定义它的程序，对其它程序不可见。包括：用户自定义变量，位置变量。永久变量是环境变量，其值不随 shell 脚本的执行结束而消失。

自定义用户变量

用户定义的变量由字母或下划线开头，由字母、数字或下划线序列组成，并且大小写字母意义不同。变量名长度没有限制。

在使用变量值时，要在变量名前加上前缀“$”。

设置和使用变量

设置变量：习惯上用大写字母来命名变量。变量名只能以字母表中的字符开头，不能用数字。

变量赋值：赋值号“=”两边应没有空格。

定义时赋值，如 NUM=1

将一个命令的执行结果赋给变量，如：TIME=`date`

将一个变量赋给另一个变量，如：A=$B

使用 echo 命令查看变量值。例如：echo $A

列出所有的变量：

```
# set
```

包含多个字的变量：

```
$NAME=Mike Ron
```

运行时出错，应改为：

$NAME="Mike Ron" 或 $NAME='Mike Ron'
单引号和双引号的区别：
$ABC='$NAME Junior'
#echo $ABC
$NAME Junior
单引号之间的内容原封不动地指定给变量。
删除变量：
unset NAME

位置变量和特殊变量

Shell 解释执行用户命令时，将命令行的一个部分作为命令名，其它部分作为参数。由出现在命令行上的位置确定的参数称为位置参数。
例如：
ls -l file1 file2 file3
$0 这个程序的文件名 ls -l
$n 这个程序的第 n 个参数值,n=1-9

特殊变量
$* 这个程序的所有参数
$# 这个程序的参数个数
$$ 这个程序的 PID
$! 执行上一个后台命令的 PID
$? 执行上一个命令的返回值

Shell 命令

read 命令

read 命令：从键盘读入数据，赋给变量
如：read USERNAME
read 的例子：
#! /bin/sh
read first second third
echo "the first parameter $first"
echo "the second parameter is $second"
echo "the third parameter is $third"

expr 命令

Shell 变量的算术运算：
expr 命令：对整数型变量进行算术运算
例如：expr 3 + 5
　　expr $var1 - 5
　　expr $var1 / $var2
　　expr $var * 10
复杂的运算：
　　expr `expr 5 + 7` /$var4
将运算结果赋予变量：
　　var4=` expr $var1 / $var2 `

```
#!/bin/sh
a=10
b=20
c=30
value1=`expr $a + $b +$c`
echo "The value of value1 is $value1"
value2=`expr $c / $b`
echo "The value of value2 is $value2"
value3=`expr $c \* $b`
echo "The value of value3 is $value3"
value4=`expr $a + $c / $b`
echo "The value of value4 is $value4"
```

变量测试语句

变量测试语句：用于测试变量是否相等、是否为空、文件类型等。
格式：
　　test 测试条件
测试范围：整数、字符串、文件
字符串测试：
test str1=str2 测试字符串是否相等
test str1!=str2 测试字符串时候不相等
test str1　　测试字符串是否不为空
test -n str1　　测试字符串是否不为空
test -z str1　　测试字符串是否为空
整数测试：
test int1 -eq int2 测试整数时候相等
test int1 -ge int2 测试 int1 是否>=int2
test int1 -gt int2 测试 int1 是否>int2
test int1 -le int2 测试 int1 是否<=int2

test int1 -lt int2 测试 int1 是否<int2
test int1 -ne int2 测试整数是否不相等
文件测试
test -d file 指定文件是否目录
test -f file 指定文件是否是常规文件
test -x file 指定文件是否可执行
test -r file 指定文件是否可读
test -w file 指定文件是否可写
test -a file 指定文件是否存在
test -s file 文件大小是否非 0

变量测试语句一般不单独使用，一般作为 if 语句的测试条件，如：

```
    if test -d $1 then
        ...
    fi
```

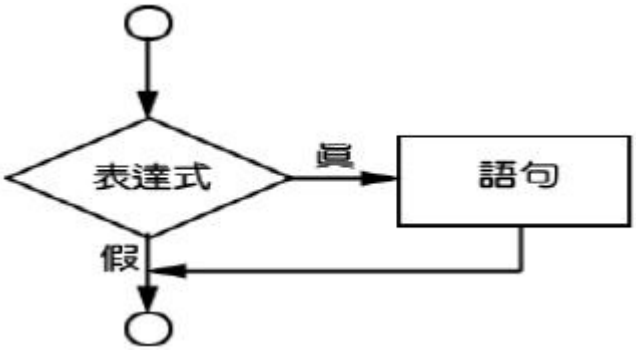
变量测试语句进行简化，如

```
    test -d $1  等价于[ -d $1 ]
#!/bin/sh
if [ $# -ne 2 ];then
    echo "Not enough parameters"
    exit 0
fi
if [ $1 -eq $2 ]; then
    echo "$1 equals $2"
elif [ $1 -lt $2 ];then
    echo "$1 littler than $2"
elif [ $1 -gt $2 ]; then
    echo "$1 greather than $2"
fi
```

流控制语句

流控制语句：用于控制 shell 程序的流程
exit 语句：退出程序执行，并返回一个返回码，返回码为 0 表示正常退出，非 0 表示非正常退出。
例如：exit 0

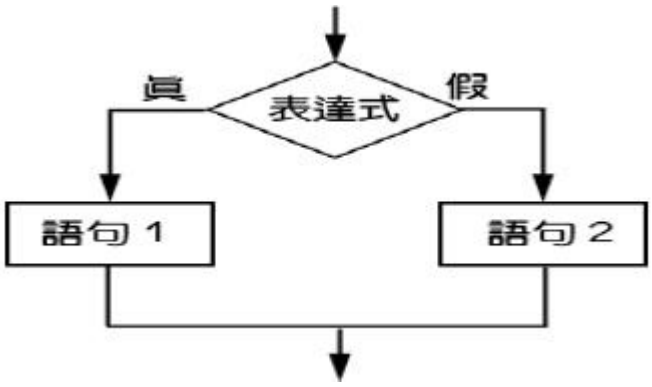
if 语句的流程图



if ... then ... fi 语句，例如：

```
#!/bin/sh
if [ -x /etc/rc.d/init.d/httpd ]
   then
   /etc/rc.d/init.d/httpd restart
fi
```

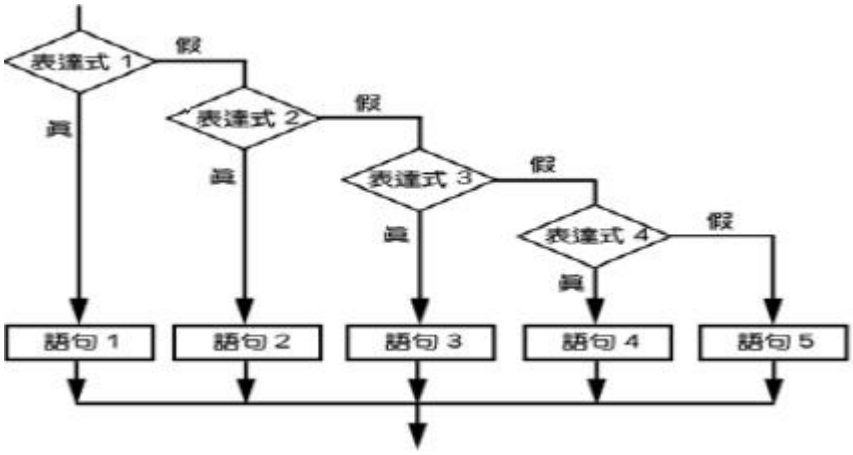
if/else 语句的流程图



更复杂的 if 语句：

```
      if 条件 1 then
           命令 1
      elif 条件 2 then
           命令 2
      else
           命令 3
      fi
```

if/else 嵌套的流程图



多格条件的联合：

-a ： 逻辑与，仅当两个条件都成立时，结构为真。

-o ： 逻辑或，两个条件只要有一个成立，结果为真。

```
echo "please input a file name;"
read file_name
if [ -d $file_name ]
   then
   echo "$file_name is directory"
elif [ -f $file_name ]
   then
   echo "$file _name is a common file"
elif [ -c $file_name -o -b $file_name ]
   then
   echo "$file_name is a device file "
else
   echo "$file_name is unknown file"
fi
```

for ... done 语句

格式 ： for 变量 in 名字表

do

命令列表

done

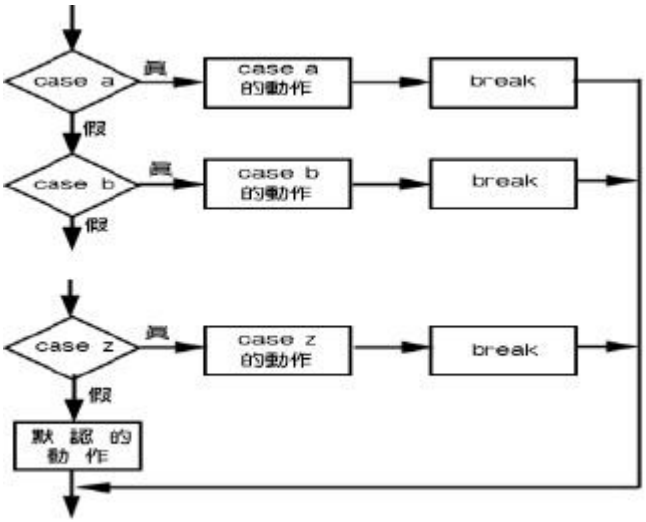
例子：

```
#!/bin/sh
for DAY in Sunday Monday Tuesday Wednesday Thurday Friday Saturday
do
  echo "The day is : $DAY"
```

done
select　变量 in 关键字
do
　　command 1
　　.....
　　command n
done
select 把关键字中的每一项做成类似表单，以交互的方式执行 do 和 done 之间的命令。

case ... esac 语句，格式：

　　case 变量 in
　　　　字符串 1） 命令列表 1
　　　　　　..
　　　　　　;;
　　　　...
　　　　字符串 n）命令列表 n
　　　　　　..
　　　　　　;;
　　esac



while 语句，格式：

　　while 条件
　　do
　　　命令
　　done
#!/bin/sh
num=1

```
while [ $num -le 10 ]
do
   SUM=`expr $sum \* $num`
   echo $SUM
   num=`expr $num + 1`
done
```

until 语句，格式：

```
until 条件
do
   命令
done
```

until 类似 while 循环，不同的是 until 是条件返回值为假才继续执行。

跳出循环：break 和 continue

break：跳出整个循环
continue：跳出本次循环，进行下次循环

shift 指令：参数左移，每执行一次，参数序列顺序左移一个位置，$#值减 1，用于分别处理每个参数，移出去的参数不再可用

```
#!/bin/sh
if [ $# -le 0 ]
then
   echo "Not enough parameters"
   exit 0
fi
sum=0
while [ $# -gt 0 ]
do
   sum=`expr $sum +$1`
   shit
done
echo $sum
```

函数应用

函数的定义：

函数名 ()
{
命令序列
}
函数的调用：不带()
函数名 参数 1 参数 2 ...
函数中的变量：
变量均为全局变量，没有局部变量
函数中的参数：调用函数时，可以传递参数，在函数中用$1、$2....来引用

Shell 脚本调试

sh -x script
这将执行该脚本并显示所有变量的值
sh -n script
不执行脚本只是检查语法的模式，将返回所有的语法错误。

awk 命令应用

awk -F 域分隔符'命令'
示例：
1、检测系统中 UID 为 0 的用户
awk -F: '$3==0 {print $1}' /etc/passwd
2、检测系统中密码为空的用户
awk -F: 'length($2)==0 {print $1}' /etc/shadow

echo 密码 | passwd --stdin 用户名
sh 脚本
1、对脚本有 r 权限
2、 对脚本目录由 rx 权限
脚本：对脚本有 x 权限

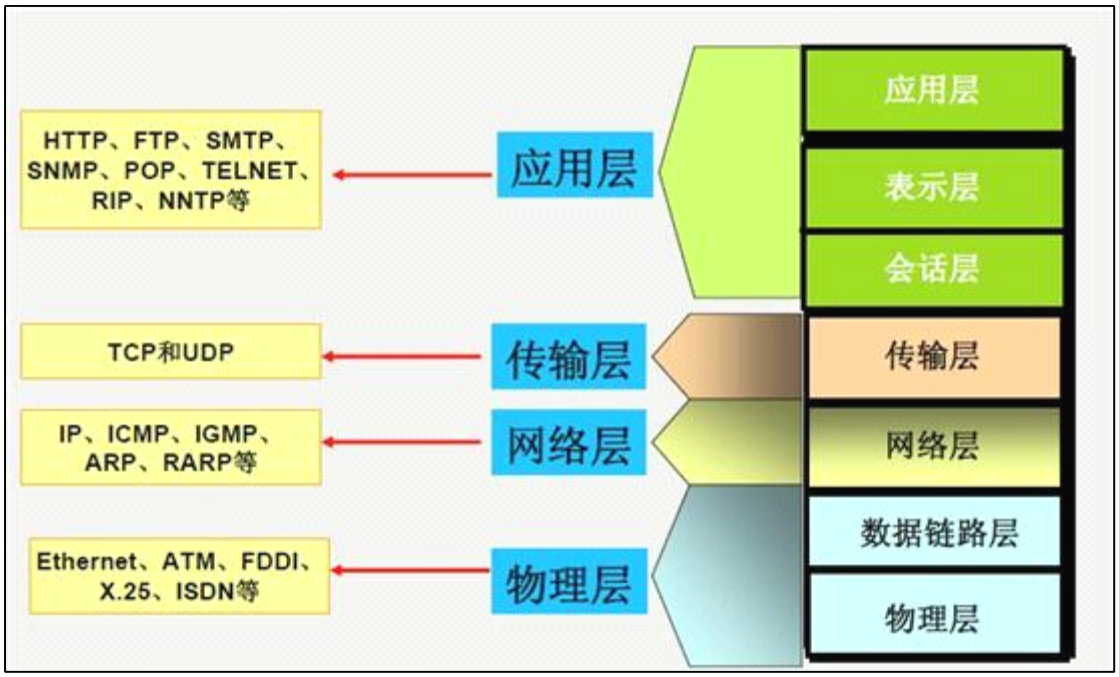
知识点总结

- 掌握 Shell 编程的基本语法
- 掌握结合系统命令编写应用脚本
- 掌握 Shell 编程调试命令

3.29.1 Linux 网络设置

互联网寻址过程

TCP/IP 与 OSI 的比较



网卡
TCP/IP－物理层
OSI－物理层（网卡）/数据链路层（MAC 地址）

网络层：IP 地址、ICMP、ARP/RARP……

传输层：TCP、UDP

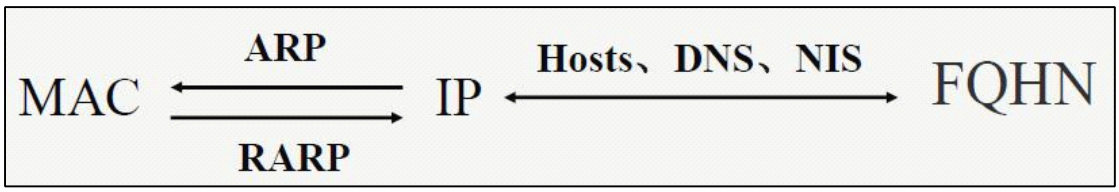
会话层：建立连接会话
表示层：加密、压缩
应用层：服务应用

数据链路层　　网络层　　应用层
MAC　　IP　　FQHN（完整的计算机名称）

00:0C:29:D0:32:37
前 24 位代表厂商标识，后 24 位代表网卡标识

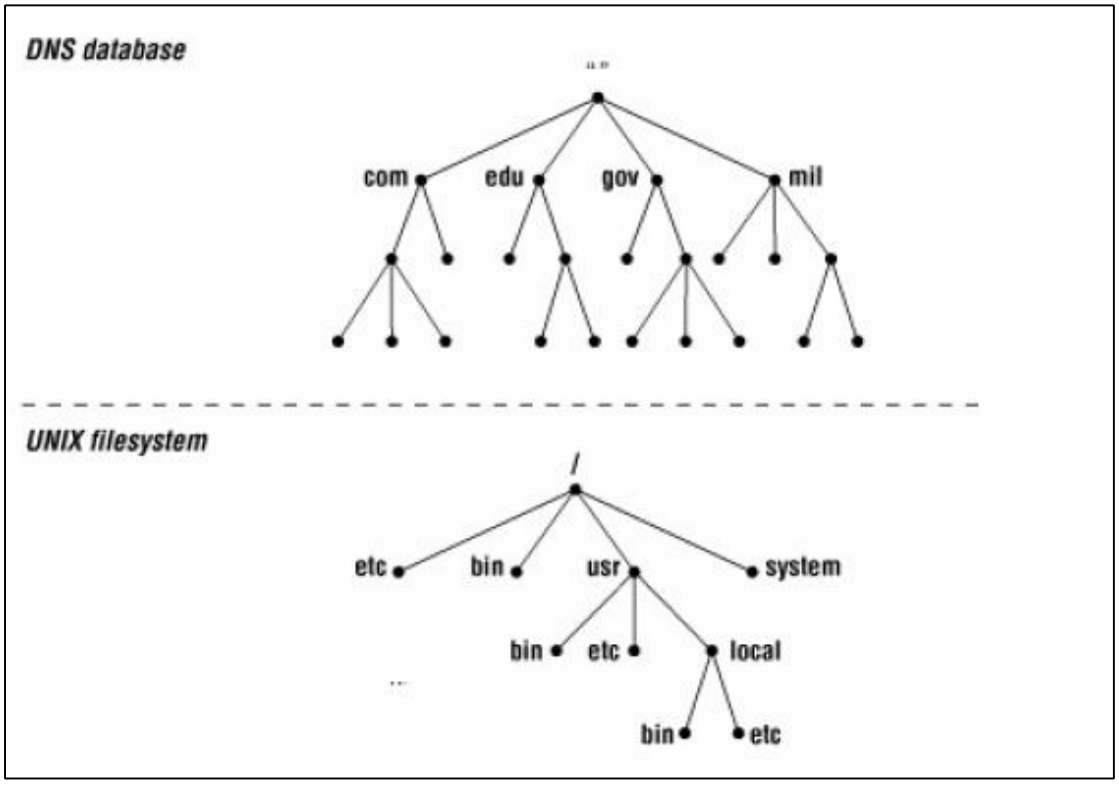
192.168.16.155
IPv4　IPv6　128

互联网的计算机寻址

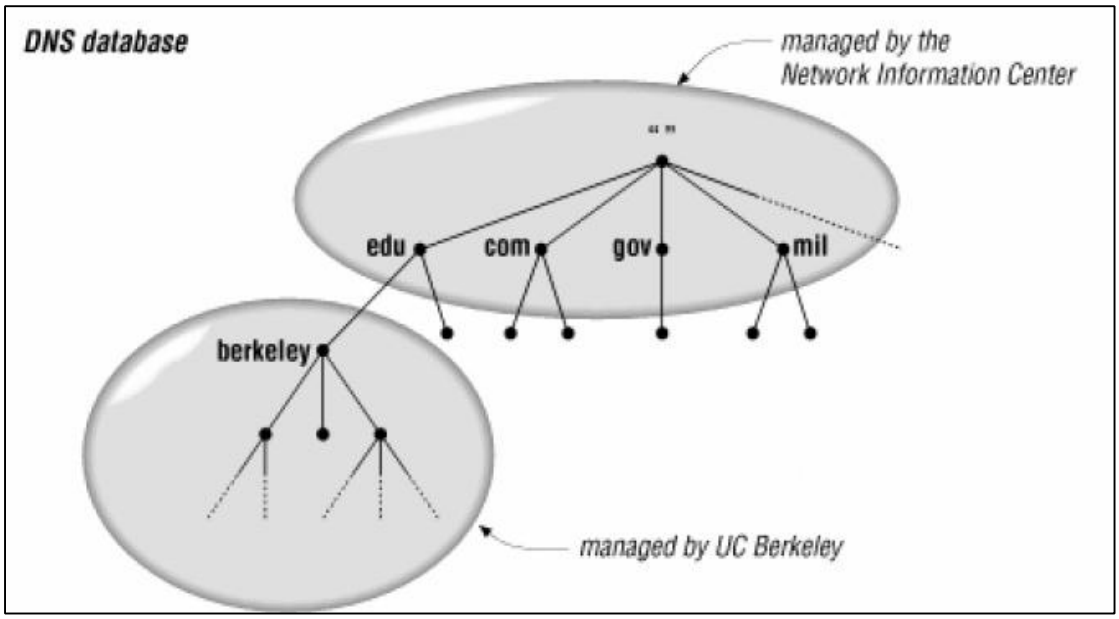


www.lampbrother.net.
主机名 二级域 顶级域 根域

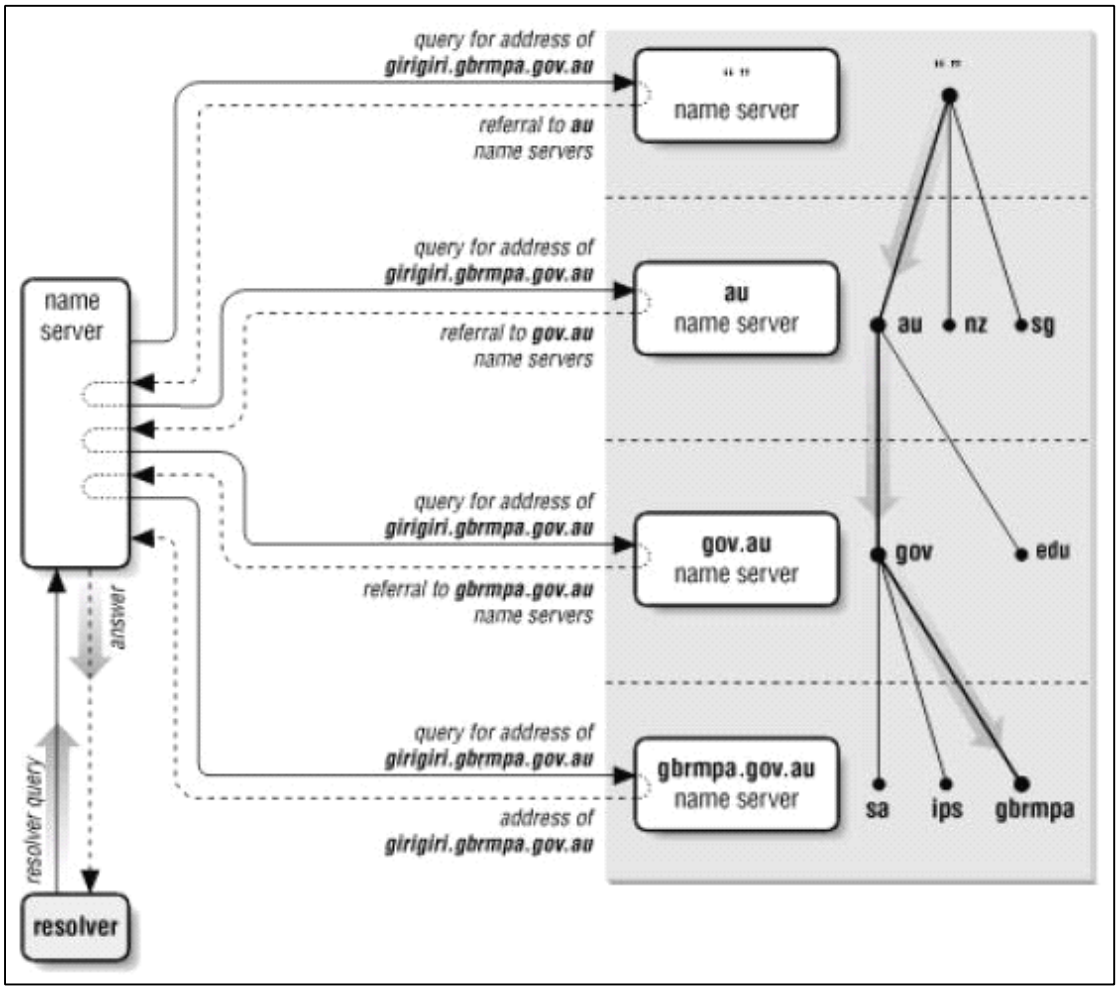
域名服务的层次结构



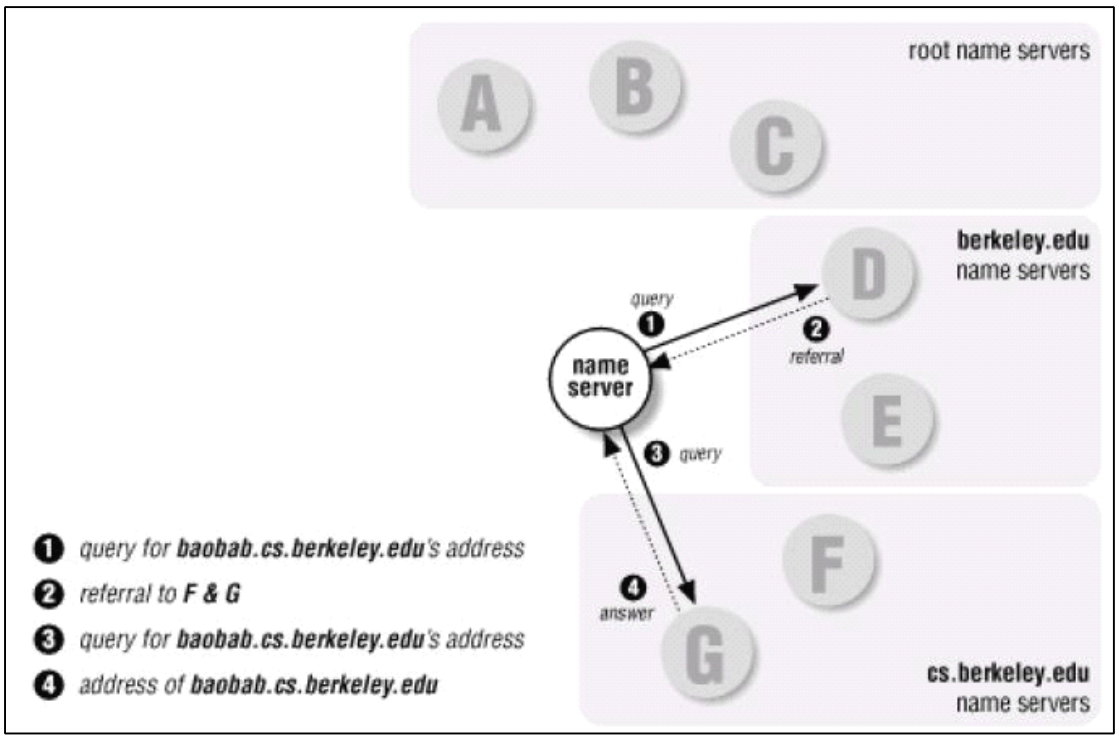
域的委托管理


DNS database
managed by the
Network Information Center
" "
edu
com
gov
mil
berkeley
managed by UC Berkeley

递归查询


query for address of
girigiri.gbrmpa.gov.au
" "
name server
referral to au
name servers
query for address of
girigiri.gbrmpa.gov.au
au
name server
referral to gov.au
name servers
name
server
query for address of
girigiri.gbrmpa.gov.au
gov.au
name server
referral to gbrmpa.gov.au
name servers
answer
query for address of
girigiri.gbrmpa.gov.au
gbrmpa.gov.au
name server
address of
girigiri.gbrmpa.gov.au
resolver query
resolver
" "
au
nz
sg
gov
edu
sa
ips
gbrmpa

缓存机制



ARP 解析过程

MAC报头		IP报头		ARP请求报文
信宿	信源	信宿	信源	
FF-FF-FF-FF-FF-FF	02:60:8C:01:02:03	166.111.1.1	166.111.1.2	你的MAC地址是什么

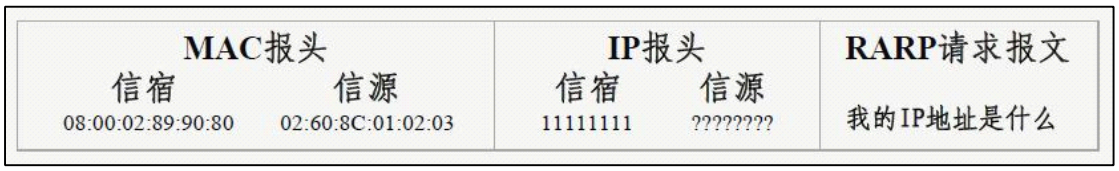
MAC报头		IP报头		ARP应答报文
信宿	信源	信宿	信源	
02:60:8C:01:02:03	08:00:02:89:90:80	166.111.1.2	166.111.1.1	这是我的MAC地址

MAC报头		IP报头		数据
信宿	信源	信宿	信源	
08:00:02:89:90:80	02:60:8C:01:02:03	166.111.1.1	166.111.1.2	

ARP 地址解析协议

arp -a 查看 arp 缓存表
-d 删除 arp 记录
-s 添加静态 arp 记录

RARP 解析过程



思考，什么时候使用 RARP？
知道自己的 MAC 地址，不知道自己的 IP 地址

/etc/hosts 文件解析
IP 地址　主机名或域名　别名

DNS Domain Name System －－ BIND

--> www.lampbrother.net
1、hosts
2、DNS 服务器
3、DNS 查询

hosts--DNSServer--根域--.net--lampbrother.net

www.lampbrother.net　再次访问调用缓存记录
www.iso.net
hosts--DNSServer--.net--iso.net
www.php.lampbrother.net
hosts--DNSServer--lampbrother.net

指定 DNS 服务器：/etc/resolv.conf
nameserver DNS 服务器地址 1 DNS 服务器地址 2 （最多写三个）

NIS　Network Information System　网络信息系统
信息集中管理（用户信息文件、域名解析文件 hosts）

MAC－－> 多 IP
IP－－> 多 MAC　cluster、bonding
IP－－> 多域名
域名－－> 多 IP

网络配置文件

- /etc/sysconfig/network-scripts/ifcfg-eth0 文件，保存 IP 地址和网关
- /etc/sysconfig/network 文件，保存本机主机名
- /etc/rc.d/init.d/network 网络启动脚本

- /etc/hosts 主机名数据库
- /etc/services 文件，网络服务信息
- /etc/resolv.conf 指定 DNS 服务器地址

/etc/sysconfig/netwrok-scripts/ifcfg-eth0
IPADDR=
GATEWAY= 设置网关
/etc/sysconfig/network
HOSTNAME=

网络管理命令

- 查看网络端口设置：ifconfig
- 查看 ARP 表信息：arp
- 探测远程主机：ping
- 操作路由表：route
- 查看路由路径：traceroute
- 监控网络状态：netstat

ifconfig eth0 192.168.16.155 netmask 255.255.252.0 broadcast
ifconfig eth0 down/up

route add default gw 192.168.16.111 指定网关

Zebra 路由软件【小日本的】
traceroute 网络地址

练习

1、练习 Linux 网络管理命令
2、熟悉 Linux 网络配置文件

3.29.2 文件共享服务（3.30 续）

Windows 与 Linux 文件共享-Samba

Samba 简介

在 UNIX 系统中，Samba 是通过服务器消息块协议（SMB）在网络上的计算机之间共享文件和打印服务的软件包。

SMB（Server Message Block）协议是一种客户端服务器协议，位于应用层，是 windows 共享文件和打印服务的标准化协议。可以在两台计算机之间共享文件、打印机。

Samba 是一个开发源代码的软件，任何拥有 GNU 组织通用公共许可证 GPL 的用户都可以使用，并免费获得它的源代码和可执行文件。
Samba 的主页：http://www.samba.org

启动服务：/etc/rc.d/init.d/smb start
Samba 有两个守护进程：smbd 和 nmbd
--------- smbd 监听 139 TCP 端口 用户验证、文件共享
--------- nmbd 监听 137 和 138UDP 端口 浏览共享、名称通信
smbd 进程的作用是处理 SMB 请求包，负责用户验证和文件共享；nmbd 进程的作用是处理浏览共享和计算机名称解析。

/etc/samba/smb.conf .ini
rpm -Vf 文件

Samba 配置
Samba 配置文件应存放在：
/etc/samba/smb.conf
包括四个设置段：
[global] 设置全局环境选项
[homes] 设置用户宿主目录共享
[printers] 设置打印机共享
[sharefiles] 设置文件共享
注：行前有分号“;”或英镑符“#”表示注释

[global]段主要选项设置：
workgroup = 指定工程组或域
server string = 描述
security = 指定安全模式
hosts allow = 限定主机访问
log file = 指定日志文件存放位置
max log size = 指定日志文件大小

security = user 默认，Samba 服务器验证用户
share 无验证机制
server 第三方主机验证用户
domain 第三方主机（Windows 域控制器）验证用户

hosts allow 只允许几台主机可以访问
hosts deny 只禁止几台主机可以访问

服务限制：1、哪些主机可以访问 2、哪些用户可以访问

[homes]段主要选项设置：
comment = Home Directories

browseable = no
writable = yes

homes 段：用户默认可以通过 Samba 服务器访问自己的宿主目录
comment 描述　browseable 不具有访问权限的目录不显示　writable 权限

Samba 应用实例一：

允许用户通过 Windows 客户端访问自己的宿主目录。
1、安装 Samba，不需对配置文件做修改，即可实现此功能
如果安装启动了 SELinux，需要先执行：
　setsebool -P samba_enable_home_dir on
2、设置用户 Samba 验证密码
　smbpasswd -a 用户名
3、启动 Samba 服务
　/etc/rc.d/init.d/smb start
Windows 客户端访问 Samba 服务器共享资源：
“开始”-“运行”，输入 \\Samba 服务器地址
Samba 服务器端查看访问客户端信息：
smbstatus

iptables -F 关闭防火墙
/etc/sysconfig/selinux　SELINUX=disabled

Samba 应用实例二：

设置 Samba 共享目录/software ，允许用户 jack 与 mary 可以通过 Windows 客户端访问，并具有读写权限。
1、创建目录/software ，添加用户 jack 和 mary
2、如果安装了 SELinux，执行命令启动用户可以访问系统目录：
chcon -t samba_share_t 共享目录
3、在 Samba 配置文件/etc/samba/smb.conf 末尾添加：
[software]
path = /software
vaild users = jack mary
writable = yes
4、设置用户 jack、mary 的 Samba 验证密码，启动 Samba 服务（服务若已启动，不需要重启）。

[共享名]
path = 共享目录
valid users = 有效访问用户

writable = 可写 yes/只读 no

小技巧：
Windows 在关掉所有共享文件和目录时在 DOS 命令方式下输入命令：
net use * /delete /y
就可以不必注销或重启而以另一个用户的身份登录 samba 服务器。
问题：用户 jack 与 mary 是否可以对/software 进行写操作？、
groupadd softadm
usermod -G softadm jack
usermod -G softadm mary
chgrp softadm /software
chmod g+w /software
ls -ld /software
drwxrwxr-x 2 root softadm 512 Jul 14 06:17 /software
grep softadm /etc/group
softadm : : 100 :jack,mary

Samba 服务器检测命令
测试语法错误：
testparm

Samba 应用案例三：

建立一个公共的只读目录 /public,所有人可以浏览目录的内容。
[public]
comment = Read Only Public
path = /pubilc
writable = no

Samba 应用案例四：

建立部门资源共享目录 /hr ，部门每个人都能读写，但不能删除别人的文件。
[hr]
path = /hr
valid users = zhangsan lisi wangwu
writable = yes
#chmod 1777 /hr

FTP 服务器配置

Wu-FTP：古老、配置复杂。
Proftp：功能强大
vsftp：安全、高速、稳定。
系统默认 ftp 软件
启动：/etc/rc.d/init.d/vsftpd start
（默认启动后即支持用户登录访问其宿主目录及匿名访问）
配置文件：/etc/vsftpd/vsftpd.conf

选项设置

anonymous_enable = YES
允许匿名登录
local_enable = YES
允许本地用户登录
xferlog_enable = YES
xferlog_file = /var/log/xferlog
激活上传和下载日志
ftpd_banner=Welcome to Sam's FTP service,enjoy it
设置欢迎信息
listen_port = 10011
指定非标准端口

允许用户可以上传到宿主目录
1、若启动 SELinux 开始上传设置
setsebool -P ftp_home_dir 1
*可关闭 SELinux，编辑/etc/selinux/config
setsebool -P allow_ftpd_full_access 1
2、确保 vsftpd 选项开启
write_enable = yes

idle_session_timeout = 600
用户会话空闲 10 分钟后被挂断
max_clients = 200
服务器的总的并发连接数为 200
max_per_ip = 3
每个客户机的最大连接数为 3
Local_max_rate = 50000
Anon_max_rate = 30000
本地用户的最大传输速率为 50KB/S，匿名用户为 30KB/S。

本地用户的访问控制

闲置指定的本地用户不能访问，而其他本地用户可以访问：
userlist_enable = YES
userlist_deny = YES
userlist_file = /etc/vsftpd.user_list
限制指定的本地用户可以访问，而其他本地用户不可以访问：
userlist_enable = YES
userlist_deny = NO
userlist_file = /etc/vsftpd.user_list

思考：
如何设定网页存放目录，某个用户可以更新网页，另外两个用户只可以查看。

匿名 FTP 用户名：ftp　anonymous
密码：邮箱
ftp 伪用户宿主目录 /var/ftp

/etc/vsftpd/vsftpd.conf

ftp FTP 地址
ftp>
ls - 查看目录下文件
cd - 切换目录（FTP）
bin - 二进制传输
lcd - 切换下载目录（本地）
get - 下载单个文件
mget - 下载多个文件
put - 上传单个文件
mput - 上传多个文件
prompt 一 关闭交互模式
bye - 退出
open - 连接 FTP 服务器
user - 输入 FTP 服务器用户名和密码

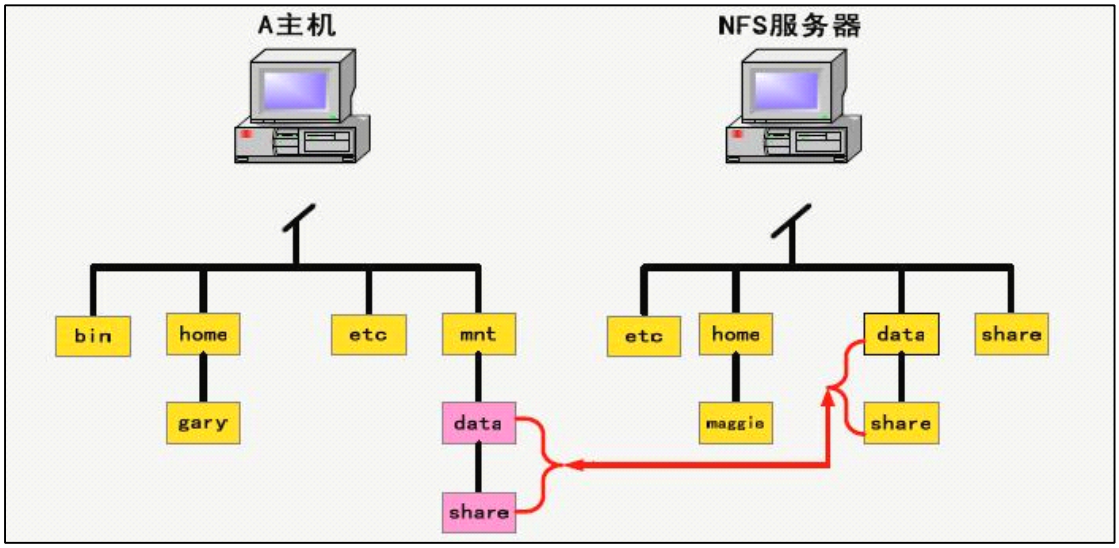
ftp -n < auto.ftp

chroot　在 ftp 用户把宿主目录当作/目录

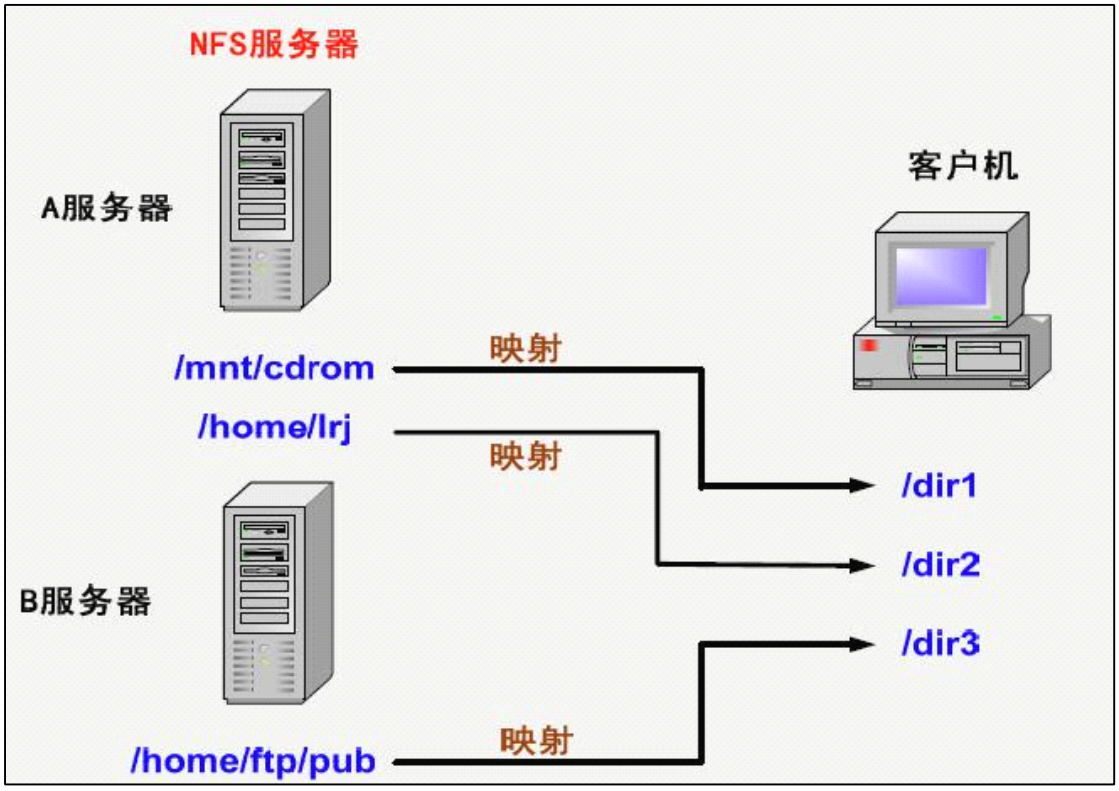
nmap 端口扫描工具

NFS 网络文件服务

NFS 即网络文件系统，用户在 Linux 系统间通过网络进行文件共享，用户可以把网络中 NFS 服务器提供的共享目录挂载到本地目录中，可以像对本地文件系统一样操作 NFS 文件系统中的内容。



NFS 共享示意图



安装和启动 NFS 服务

NFS 服务是 Linux 系统内置的功能，一般安装时并不会自动指定开机时启用。因此，可以执行 ntsysv 命令，设置 nfs 开机时自动启动 NFS 服务。

rpm -q portmap

rpm -qa | grep nfs

查看时候安装 NFS 相关组件

service portmap start

service nfs start

注意：portmap 一定要优先于 nfs 启动

检查 RPC 程序是否启动

由于 NFS 是通过 RPC（Remote Procedure Call，远程过程调用）协议来使用远程计算机的文件，因此系统中的 RPC 服务必须启动。

```
[root@www root]# rpm -qa nfs-utils portmap
portmap-4.0-54
nfs-utils-1.0.1-2.9
[root@www root]# rpcinfo -p
   program vers proto   port
    100000    2   tcp    111  portmapper
    100000    2   udp    111  portmapper
    100024    1   udp  32768  status
    100024    1   tcp  32768  status
    391002    2   tcp  32769  sgi_fam
[root@www root]# _
```

设置共享目录

设置 NFS 服务器共享的目录，则需配置

/etc/exports 文件来控制

exports 格式编写

/sharedirectory　　host　　（rights）

共享目录的绝对路径　　客户主机地址　　权限

exports 配置举例

/home/ftp/pub *（ro）

表示所有主机皆可访问此目录，权限为可读（ro）
/home/ftp/pub 192.168.9.100 (ro)
只允许 192.168.9.100 主机访问此目录，权限为可读 （ro）
/home/ftp/pub 192.168.9.0 (rw)
只允许 192.168.9.0 该网段上的主机访问此目录，权限为（rw）

exportfs 重新输出共享目录格式：exportfs -rv

配置 NFS 客户端

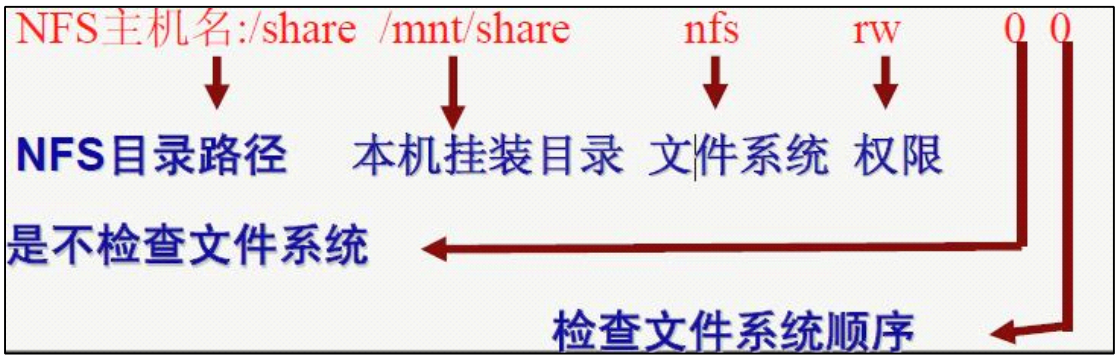
mkdir /mnt/share 创建目录挂载点 share
showmount -e NFS 主机名或 IP 地址
查看 NFS 服务器共享了哪些目录
mount NFS 主机名或 IP 地址： /share/ /mnt/share/
将 NFS 主机上的共享目录，映射到指定位置
ls /mnt/share 查看新映射的目录
umount /mnt/share/ 断开映射目录

exportfs 命令

exportfs 命令用户维护当前主机中 NFS 服务器的输出目录列表。
重新输出共享目录
命令格式： exportfs -rv
功能描述
该命令是 NFS 服务器重新读取 exportfs 文件中的设置，使用该命令可以在改变 exports 文件设置后，使当前 NFS 服务器生效，而不需要重新启动 NFS 服务器。

开机时自动映射 NFS 目录

```
LABEL=/                 /                       ext3    defaults        1 1
LABEL=/32               /32                     ext3    defaults        1 2
none                    /dev/pts                devpts  gid=5,mode=620  0 0
LABEL=/home             /home   ext3  defaults,usrquota,grpquota         1 2
none                    /proc                   proc    defaults        0 0
none                    /dev/shm                tmpfs   defaults        0 0
LABEL=/usr              /usr                    ext3    defaults        1 2
LABEL=/var              /var                    ext3    defaults        1 2
/dev/sda5               swap                    swap    defaults        0 0
/dev/cdrom              /mnt/cdrom              iso9660 noauto,owner,kudzu,ro 0
0
/dev/fd0                /mnt/floppy             auto    noauto,owner,kudzu 0 0
```



远程管理控制 SSH

ssh 远程登录：ssh 用户名@远程主机 IP 地址
常用选项：-2 ：强制使用第二代 SSH 协议
-p ： 端口号
示例：
ssh -2 sam@192.168.9.25
配置文件：/etc/ssh/sshd_config
Windows 平台 SSH 工具：SSH Workstation

/etc/exports
SSH
1、ssh 远程登录 SecureCRT
2、sftp 文件共享（类 FTP） SSH Secure File Transfer Client
3、scp 文件共享（类 cp）

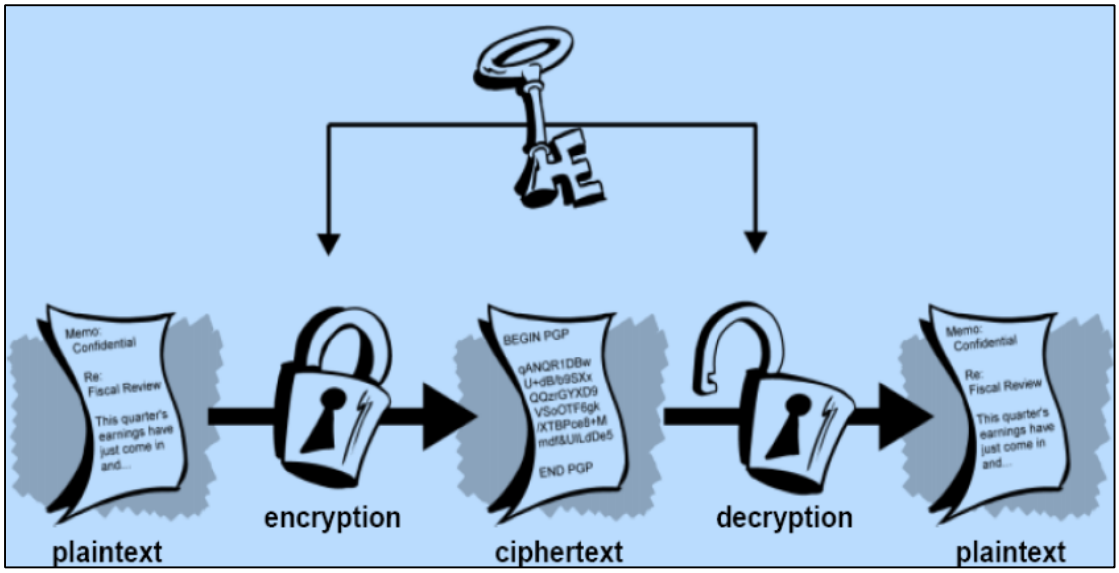
本机拷贝文件到远程主机：
scp 本地文件 用户名@远程主机地址：远程主机目标目录
scp -r 本地目录 用户名@远程主机地址：远程主机目标目录
从远程主机拷贝文件到本地：
scp 用户名@远程主机地址：远程文件 本地目录
scp -r 用户名@远程主机地址：远程目录 本地目录

常用选项： -p ： 保持原有文件属性
-r ： 复制目录
-P ： 指定端口号

对称密钥加密

加密与解密使用同一密钥
优势：速度快

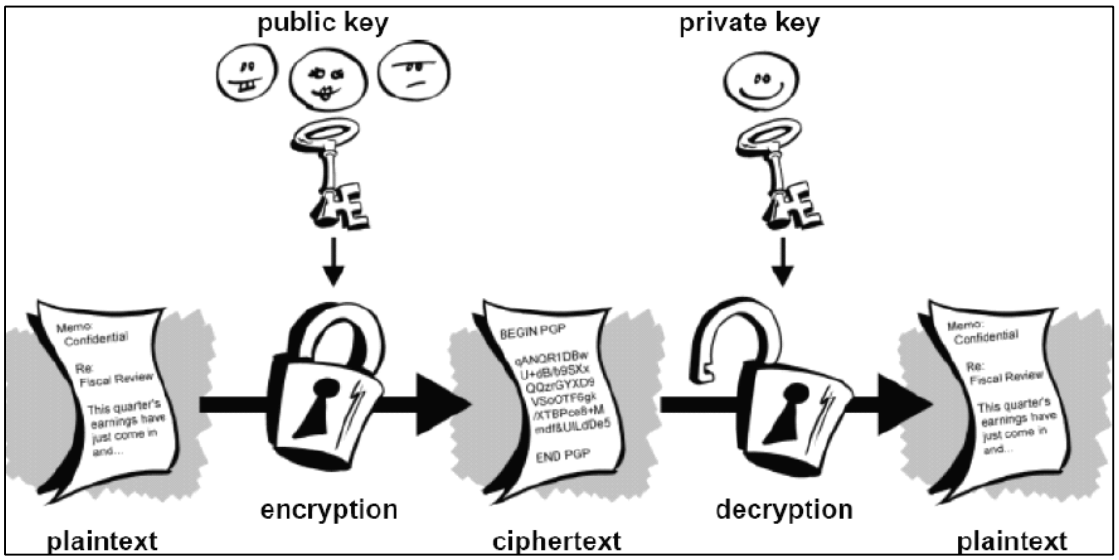
缺点：密钥本身需要交换

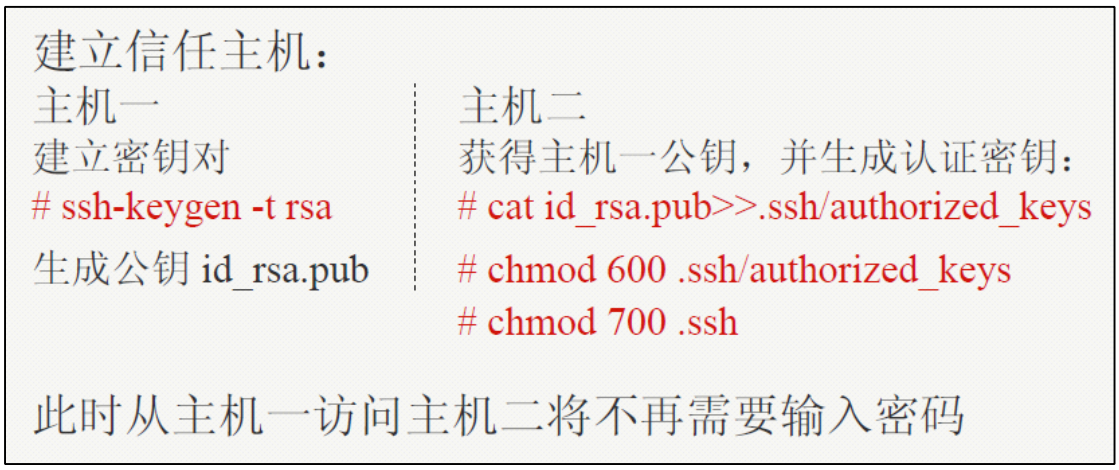


非对称密钥加密

也称公开密钥加密，使用时生成两个密钥，一个公开存放，为公钥；一个私人持有为私钥。用户用其中一个密钥加密的数据只有用另一个密钥才能解密。

优势：安全性好
缺点：速度慢
所以加密信息时，通常是对称密钥与非对称密钥机密结合使用。





SSH 信任主机

原理：192.168.16.132(root) ---> 192.168.16.131(backup)
公钥---------->信任密钥

数字签名认证的实现：本地主机发送私钥数据，远程主机公钥印证，建立信任关系

主机一 主机二
192.168.16.132 192.168.16.131
用户: root 用户: backup
|
root: |
生成密钥对: ssh-keygen -t rsa |
root 宿主目录下.ssh/id_rsa.pub |
----------> /home/backup (拷贝至登录用户宿主目录下)

backup:
mkdir .ssh
cat ~/id_rsa.pub >> ~/.ssh/authorized_keys
chmod 700 .ssh
chmod 600 .ssh/authorized_keys

root:
ssh backup@192.168.16.131
scp -rp backup@192.168.16.131:/website /backup/website_$(date +%F)
rsync -arHz --progress --delete -e ssh backup@192.168.16.131:/website /backup

注意：与用户身份相关

ssh 1、空密码用户登录禁止 2、root 登录禁止

-2 SSH2
/etc/ssh/sshd_config
PermitRootLogin yes
Port

服务器/website 192.168.16.155　penny －－> 备份服务器（localhost） root
每周日 做一次完全备份
1、crontab -e
0 2 * * 0 /usr/bin/scp -rp webadm@192.168.16.155:/website /backup/website_$(date +%F)
* 建立信任主机
1）备份服务器生成密钥，拷贝公钥给服务器
2）服务器生成认证密码

周一到周六 做一次增量备份
2、crontab -e
*/10 * * * 1-6 /urs/bin/rsync -arHz --delete -e ssh penny@192.168.16.155:/website /backup

用公钥加密，私钥解密 －－加密文件
用私钥加密，用公钥解密 －－数字签名

rsyns 应用

- 方便的增量备份实现
- 可镜像保存整个目录数和文件系统
- 保持文件的权限、时间、软硬链接等
- 文件传输效率高
- 可以使用 SSH 加密通道

启用 rsync：
编辑配置文件/etc/xinetd.d/rsync
设置 disable = no
重启 xinetd 进程 service xinetd restart

示例：
rsync -arHz --progress --delete webadm@192.168.16.155:/website　/backup
备份 web 服务器目录到本机/backup
rsync -arHz --progress --delete /script samlee@192.168.16.155:/home/samlee
上传/script 目录到用户 samlee 宿主目录下

-a 保持文件属性
-r 子目录递归处理
-H 保持文件硬链接
-z 备份文件传输时压缩处理
--progress 在传输时显示传输过程

--delete 删除目录备份没有的文件
-e ssh 使用 SSH 加密隧道传输

3.31.1 LAMP 环境搭建

一、准备工作

1、安装编译工具gcc、gcc-c++、make

注意解决依赖关系，推荐使用yum安装，若不能联网可使用安装光盘做为yum源——

1）编辑 yum 配置文件：

```
vi /etc/yum.repos.d/CentOS-Media.repo
    [c5-media]
    name=CentOS-$releasever - Media
    baseurl=file:///mnt/cdrom * 修改为光盘挂载点
file:///media/cdrom/
file:///media/cdrecorder/
    gpgcheck=1
    enabled=1 * 改为1意为启用
    gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-5
```

2）将网络 yum 源配置文件移至其他位置：

```
mv /etc/yum.repos.d/CentOS-Base.repo /backup/CentOS-Base.repo_$(date +%F)
```

3）依次安装 gcc、gcc-c++

```
yum install gcc
yum install gcc-c++
```

2、卸载系统 Apache、MySQL 和 PHP 的 RPM 安装包

下载前需关闭启动的服务，如httpd、mysqld

```
service httpd stop
rpm -e `rpm -qa | grep httpd`
service mysqld stop
rpm -e `rpm -qa | grep mysql`
```

rpm -e `rpm -qa | grep php`
* 有依赖关系的软件包可使用yum remove卸载

3、关闭 SELinux，允许防火墙 80 端口访问

1）关闭 SELinux

vi /etc/selinux/config
SELINUX=disabled * 若安装时没有禁用SELinux ，将enforcing改为disabled

2）关闭防火墙 Netfilter/iptables

因尚未做防火墙讲解，直接简单的关闭所有防火墙设置：
iptables -F

4、拷贝源码包，解包解压缩

拷贝可使用SSH Secure File Transfer Client、vsftpd、Samba等应用皆可
建议将LAMP环境安装源码包统一存放在一个目录下，如/lamp
可编写个批量处理脚本，一次性把所有.tar.gz的安装包解包解压缩

```
vi tar.sh
cd /lamp
    /bin/ls *.tar.gz > ls.list
for TAR in `cat ls.list`
    do
/bin/tar -zxf $TAR
    done
    /bin/rm ls.list
```

二、编译安装

* 每个源码包配置编译安装完成后，确认安装目录下是否生成安装文件

安装 libxml2

```
cd /lamp/libxml2-2.6.30
 ./configure --prefix=/usr/local/libxml2/
make install
```

安装 libmcrypt

cd /lamp/libmcrypt-2.5.8
./configure --prefix=/usr/local/libmcrypt/
make
make install
* 需调用gcc-c++编译器，未安装会报错

安装 libltdl，也在 libmcrypt 源码目录中，非新软件

cd /lamp/libmcrypt-2.5.8/libltdl
./configure --enable-ltdl-install
make
make install

安装 zlib

cd /lamp/zlib-1.2.3
./configure
make
make install
* zlib 指定安装目录可能造成 libpng 安装失败，故不指定，为卸载方便，建议 make install 执行结果输出到安装日志文件，便于日后卸载

安装 libpng

cd /lamp/libpng-1.2.31
./configure --prefix=/usr/local/libpng/
make
make install

安装 jpeg6

mkdir /usr/local/jpeg6
mkdir /usr/local/jpeg6/bin
mkdir /usr/local/jpeg6/lib
mkdir /usr/local/jpeg6/include
mkdir -p /usr/local/jpeg6/man/man1
cd /lamp/jpeg-6b
./configure --prefix=/usr/local/jpeg6/ --enable-shared --enable-static
make

make install
* --enable-shared与--enable-static参数分别为建立共享库和静态库使用的libtool

安装 freetype

cd /lamp/freetype-2.3.5
./configure --prefix=/usr/local/freetype/
make
make install

安装 autoconf

cd /lamp/autoconf-2.61
./configure
make
make install

安装 GD 库

cd /lamp/gd-2.0.35
./configure --prefix=/usr/local/gd2/ --with-jpeg=/usr/local/jpeg6/ --with-freetype=/usr/local/freetype/
make
make install
* 若前面配置zlib时没有指定安装目录，gd配置时不要添加--with-zlib=/usr/local/zlib/参数

安装 Apache

cd /lamp/httpd-2.2.9
./configure --prefix=/usr/local/apache2/ --sysconfdir=/etc/httpd/ --with-included-apr --disable-userdir --enable-so --enable-deflate=shared --enable-expires=shared --enable-rewrite=shared --enable-static-support
make
make install
* 若前面配置zlib时没有指定安装目录，Apache配置时不要添加--with-z=/usr/local/zlib/参数
启动Apache测试：
/usr/local/apache2/bin/apachectl start
ps -le | grep httpd
通过浏览器输入地址访问：http://Apache服务器地址，若显示“It works”即表明Apache正常工作
设置Apache系统引导时启动：
echo "/usr/local/apache2/bin/apachectl start" >> /etc/rc.d/rc.sysinit

安装 ncurses

cd /lamp/ncurses-5.6
./configure --with-shared --without-debug --without-ada --enable-overwrite
make
make install
* 若不安装 ncurses 编译 MySQL 时会报错
* --without-ada 参数为设定不编译为 ada 绑定，因进入 chroot 环境不能使用 ada ；
--enable-overwrite 参数为定义把头文件安装到/tools/include 下而不是/tools/include/ncurses 目录

安装 MySQL

groupadd mysql
useradd -g mysql mysql
* 添加用户组mysql ，将mysql用户默认组设置为mysql用户组
cd /lamp/mysql-5.0.41
./configure --prefix=/usr/local/mysql/ --with-extra-charsets=all
make
make install
* --with-extra-charsets=all参数为安装所有字符集
cp support-files/my-medium.cnf /etc/my.cnf
* 生成MySQL配置文件
/usr/local/mysql/bin/mysql_install_db --user=mysql
* 创建数据库授权表
chown -R root /usr/local/mysql
chown -R mysql /usr/local/mysql/var
chgrp -R mysql /usr/local/mysql
* 更改安装目录和数据目录的所有者、所属组
/usr/local/mysql/bin/mysqld_safe --user=mysql &
* 启动MySQL服务
ps -le | grep mysqld
/usr/local/mysql/bin/mysql -u root
mysql> SET PASSWORD FOR 'root'@'localhost'=PASSWORD('Am@ri31n');
mysql> exit
* 登录MySQL客户端控制台设置指定root密码
cp /lamp/mysql-5.0.41/support-files/mysql.server /etc/rc.d/init.d/mysqld
chown root.root /etc/rc.d/init.d/mysqld
chmod 755 /etc/rc.d/init.d/mysqld
chkconfig --add mysqld
chkconfig --list mysqld
chkconfig --levels 245 mysqld off
* 添加MySQL启动脚本，设置为只有运行级别3自启动

安装 PHP

cd /lamp/php-5.2.6
./configure --prefix=/usr/local/php/ --with-config-file-path=/usr/local/php/etc
--with-apxs2=/usr/local/apache2/bin/apxs --with-mysql=/usr/local/mysql/
--with-libxml-dir=/usr/local/libxml2/ --with-jpeg-dir=/usr/local/jpeg6/
--with-freetype-dir=/usr/local/freetype/ --with-gd=/usr/local/gd2/
--with-mcrypt=/usr/local/libmcrypt/ --with-mysqli=/usr/local/mysql/bin/mysql_config
--enable-soap --enable-mbstring=all --enable-sockets
* 若前面配置zlib时没有指定安装目录，PHP配置时不要添加--with-zlib-dir=/usr/local/zlib/参数
make
make install

* 若编译提示libtool没有发现等错误，需要安装libtool工具
rpm -ivh /mnt/cdrom/CentOS/libtool-*
libtool-1.5.22-7.el5_4.i386.rpm
libtool-ltdl-1.5.22-7.el5_4.i386.rpm
libtool-ltdl-devel-1.5.22-7.el5_4.i386.rpm

cp php.ini-dist /usr/local/php/etc/php.ini
* 生成PHP配置文件
vi /etc/httpd/httpd.conf
Addtype application/x-httpd-php .php .phtml
* 为将.phps做为PHP源文件进行语法高亮显示
重启Apache服务：/usr/local/apache2/bin/apachectl restart
* Apache无法启动，提示cannot restore segment prot after reloc: Permission denied错误，为SELinux 问 题 ， 可 关 闭 SELinux 或 者 尝 试 命 令 chcon -t texrel_shlib_t /usr/local/apache2/modules/libphp5.so
测试：vi /usr/local/apache2/htdocs/test.php
<?php
phpinfo();
?>
通过浏览器输入地址访问：http://Apache服务器地址/test.php

安装 Zend 加速器

cd ZendOptimizer-3.3.0a-linux-glibc21-i386
./install.sh
* 注意PHP配置文件和Apache启动脚本所在目录指定正确即可

安装 phpMyAdmin

cp -a /lamp/phpMyAdmin-3.0.0-rc1-all-languages /usr/local/apache2/htdocs/phpmyadmin
cd /usr/local/apache2/htdocs/phpmyadmin
cp config.sample.inc.php config.inc.php
vi config.inc.php
['auth_type']='http'
* 设置auth_type为http ，即设置为HTTP身份认证模式
通过浏览器输入地址访问：http://Apache 服务器地址/phpmyadmin/index.php
用户名为 root ，密码为 MySQL 设置时指定的 root 密码

4.01.1 Apache 服务器配置

Apache 基本配置

- 改变页面目录 DocumentRoot
- 设置索引文件 DirectoryIndex
- 设置服务器域名 ServerName
- 设置管理员邮箱 ServerAdmin
- 定义 Apache 缺省目录 ServerRoot
- 改变监听端口 Listen
- 日志文件 ErrorLog
- 定义进程用户及组 User Group

MPM 核心多处理模块设置
StartServers 50
MinSpareServers 15
MaxSpareServers 30
MaxClients 225
MaxRequestsPerChind 4 000

日志格式：时间 主机 进程 错误信息
ErrorLog 错误日志
CustomLog 访问日志

站点登录控制

第一步：在 Apache 配置文件/etc/httpd/httpd.conf 中添加：
Alias /doc "/share/doc"
<Directory "/share/doc">

Options Indexes FollowSymLinks
AllowOverride AuthConfig
Order allow,deny
Allow from all
<Directory>

第二步：在/share/doc 目录下建立.htaccess 文件：
AuthName "The LAMPBrother's Docs"
AuthType Basic
AuthUserFile /share/doc/.htpasswd
require valid-user

第三步：执行命令生成认证文件：
/usr/local/apache2/bin/htpasswd -c /share/doc/.htpasswd cooper
更改已设定的密码
/usr/local/apache2/bin/htpasswd -m /share/doc/.htpasswd cooper

第四步：重启服务，测试
通过浏览器输入 http://服务器 ip 或域名 /doc

/share/doc ---> http://www.lampbrother.net/doc
1、Apache 发布目录
vi /etc/httpd/httpd.conf
Alias /doc "/share/doc" 定义别名
<Directory "/share/doc">
    Options Indexes FollowSymLinks 支持文件索引功能、支持链接文件
    AllowOverride AuthConfig/None 是否认证
    Order allow,deny 定义访问主机
    Allow from all
</Directory>
2、建立访问控制文件（放在发布目录下.htaccess）AccessFileName
vi /share/doc/.htaccess
AuthName 描述
AuthType 认证方式 Basic/Digest(个别浏览器不支持)
AuthUserFile /share/doc/.htpasswd 指定密码文件路径
require valid-user 指定授权用户访问
require user zhangsan lisi
3、设定用户访问权限
/usr/local/apache2/bin/htpasswd -c /share/doc/.htpasswd 用户名
create
/usr/local/apache2/bin/htpasswd /share/doc/.htpasswd 用户名
用户为虚拟用户

4、重启测试

虚拟主机

修改 Apache 配置文件/etc/httpd/httpd.conf，设置：
/etc/httpd/extra/httpd-vhosts.conf
取消注释，使其生效
编辑 /etc/httpd/extra/httpd-vhost.conf
NameVirtualHost 192.168.1.1
<VirtualHost 192.168.1.1>
ServerAdmin webmaster@163.com
DocumentRoot /usr/local/apache/163
ServerName www.163.com
ErrorLog logs/error_log_163
CustomLog logs/access_log_163 common
</VirtualHost>
增加多域名访问设置
<VirtualHost 192.168.1.1>
ServerAdmin webmaster@google.com
DocumentRoot /usr/local/apache/google
ServerName www.google.com
ErrorLog logs/error_log_google
CustomLog logs/access_log_google common
</VirtualHost>

4.01.2 LVM 逻辑卷管理——硬盘空间动态扩展

1、安装部署
分区：/boot 分区必须单独划分 128M
然后依次建立物理卷 PV(功能设置)－卷组 VG(容器，类似扩展分区)－逻辑卷 LV(数据分区)
1）物理卷：“新建”－文件系统类型 LVM－使用全部剩余空间
2、动态扩容
添加新硬盘分区 fdisk－创建物理卷 pvcreate－扩展卷组 vgextend－扩展逻辑卷 lvextend -L－umount 挂载点－扩容生效 resize2fs -f－重新挂载 mount

示例：
硬盘存在空闲空间
a. lvextend -L +1000M /dev/mapper/VolGroup00-LogVol02
b. umount /website

c. resize2fs -f /dev/mapper/VolGroup00-LogVol02
d. mount /website
e. df -h

无空闲空间增加新硬盘
a. fdisk /dev/sdb ---> t for 8e(LVM) 创建 LVM 分区
b. pvcreate /dev/sdb1 设置物理卷功能
c. vgscan/vgdisplay 查看物理卷信息
vgextend VolGroup00 /dev/sdb1 将分区空间加入卷组
d. lvextend -L +1000M /dev/mapper/VolGroup00-LogVol02
e. umount /website
f. resize2fs -f /dev/mapper/VolGroup00-LogVol02
g. mount /website
h. df -h

注意：
1、resize2fs 前必须做完整备份
2、系统分区如/分区等扩容需要在光盘修复模式下进行
a. linux rescue 进入光盘修复模式
b. lvextend -L +1000M /dev/VolGroup00/LogVol02
c. resize2fs -f /dev/VolGroup00/LogVol02
d. df -h
e. exit

试验：
1、安装 Linux 启用 LVM 逻辑卷管理
留空闲空间练习空间增容
2、添加新硬盘
分区、设置物理卷、加入卷组、LVM 增容
3、/分区增容
光盘修复模式

4.02.1 RAID

Redundant (多余的) Array （排列）of Independent（独立的）Disks 独立磁盘冗余阵列
RAID 的基本目的是把多格小型廉价的磁盘驱动器合并成一组阵列来达到大型昂贵的驱动器所无法达到的性能或冗余性。组成磁盘阵列的不同方式成为 RAID 级别（RAID Levels）。

RAID：
1、加快数据读取速度
2、冗余备份（实时）

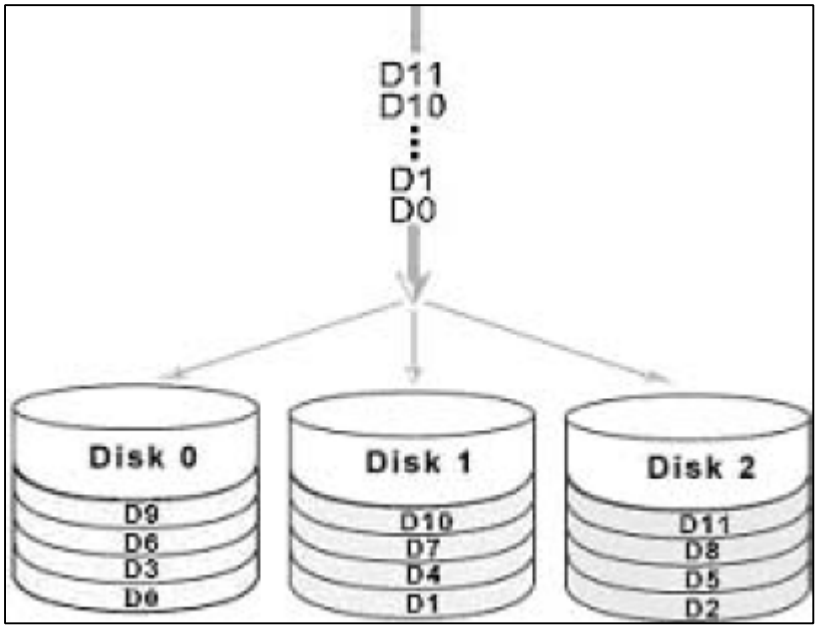
Linux 支持的 RAID 种类：RAID0、RAID1、RAID5、RAID0+1
RAID+LVM

RAID0 加快数据读取速度
RADI1 实时备份
RAID5 加快数据读取速度/实时备份（奇偶校验）

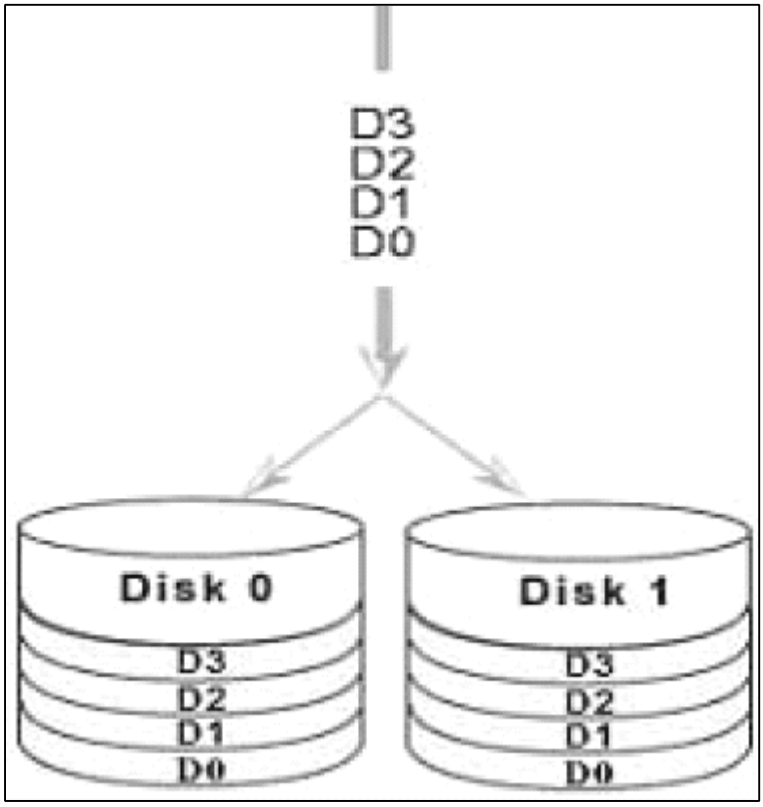
RAID 工作方式

- RAID 是一种在多个磁盘上分散信息的方法。它使用磁盘分条（disk striping，RAID 级别 0）、磁盘镜像（disk mirroring，RAID 级别 1）、和带有奇偶校检的磁盘分条（disk striping with parity，RAID 级别 5）之类的技术来达到冗余性，减低潜伏时间，并且（或者）增加磁盘读写的带宽，提高从硬盘崩溃中恢复的能力。
- RAID 的基本原理是，数据必须使用一致的形式被分散到阵列中的驱动器上。要达到这个目的，数据必须被分割成大小一致的“快”（大小通常是 32K 或 64K，也可以使用不同的大小）。每一块都会根据所用的 RAID 级别而写入启动的一个硬盘驱动器。当数据要被读取时，这个进程就会反过来进行，造成一个多个驱动器好像是一个大驱动器的假象。

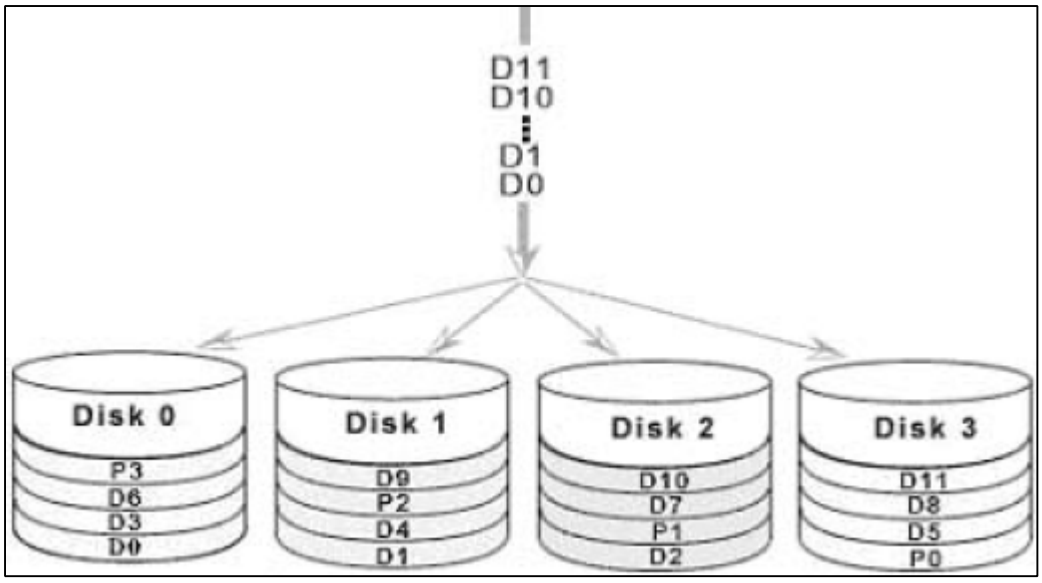
RAID 0



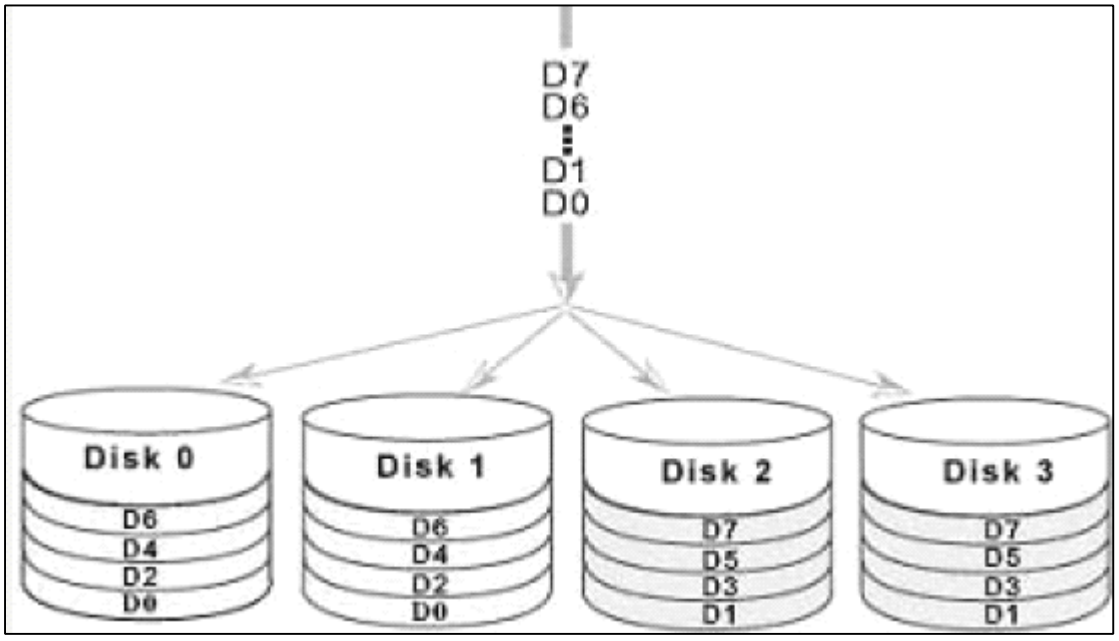
RAID 1


D3
D2
D1
D0
Disk 0
D3
D2
D1
D0
Disk 1
D3
D2
D1
D0

RAID 5


D11
D10
D1
D0
Disk 0
P3
D6
D3
D0
Disk 1
D9
P2
D4
D1
Disk 2
D10
D7
P1
D2
Disk 3
D11
D8
D5
P0

RAID10 (0 + 1)

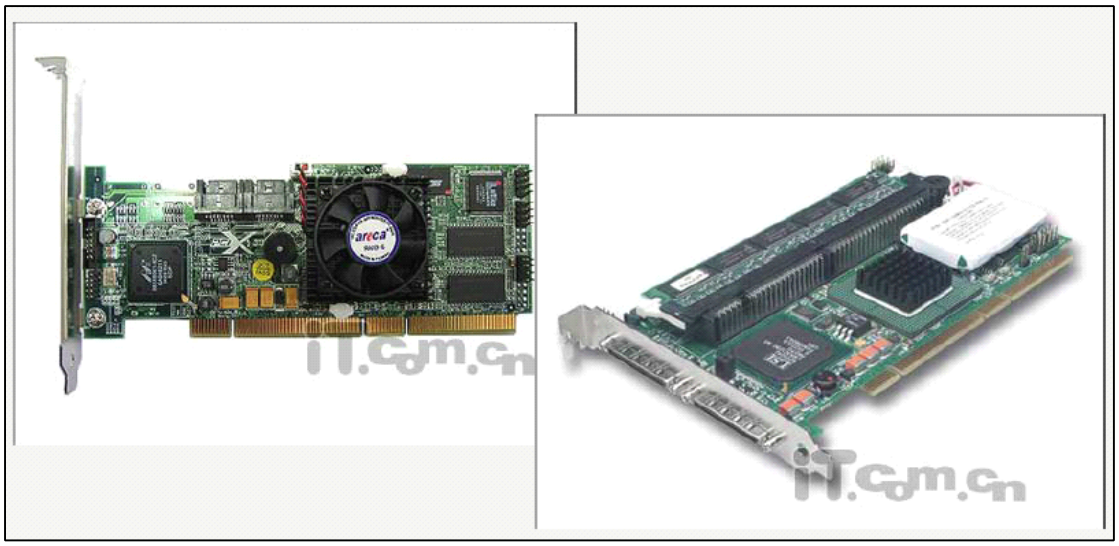


谁应该使用 RAID

任何需要使大量数据触手可及的人（如一般的系统管理员）都可以从 RAID 技术中收益。使用 RAID 的主要原因包括：

- 加快速度
- 使用一个虚拟的磁盘，从而增加存储容量
- 减少磁盘失效带来的不利影响

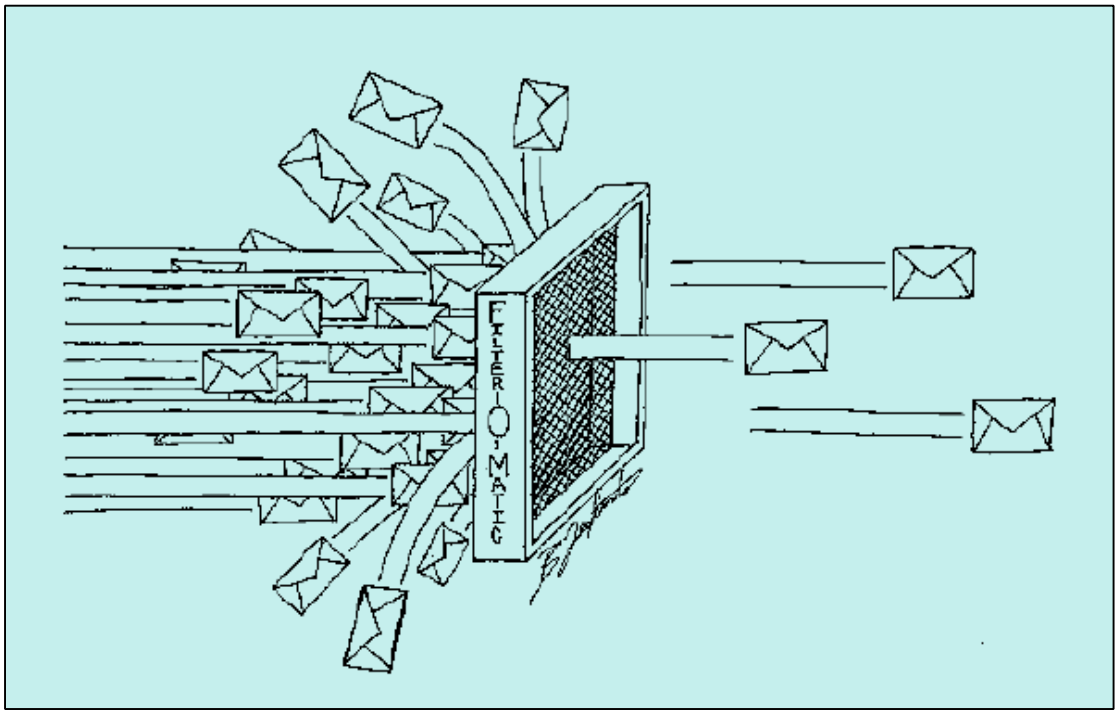
RAID 卡（SATA/SCSI）



4.02.2 防火墙设置 Netfilter/Iptables

Iptables 防火墙

Iptables 是一种基于包过滤的防火墙
Iptables 需要 2.4 以上版本的内核支持



Iptables 包过滤防火墙
SELinux 管理细化（文件），针对网络服务管理

iptables 命令

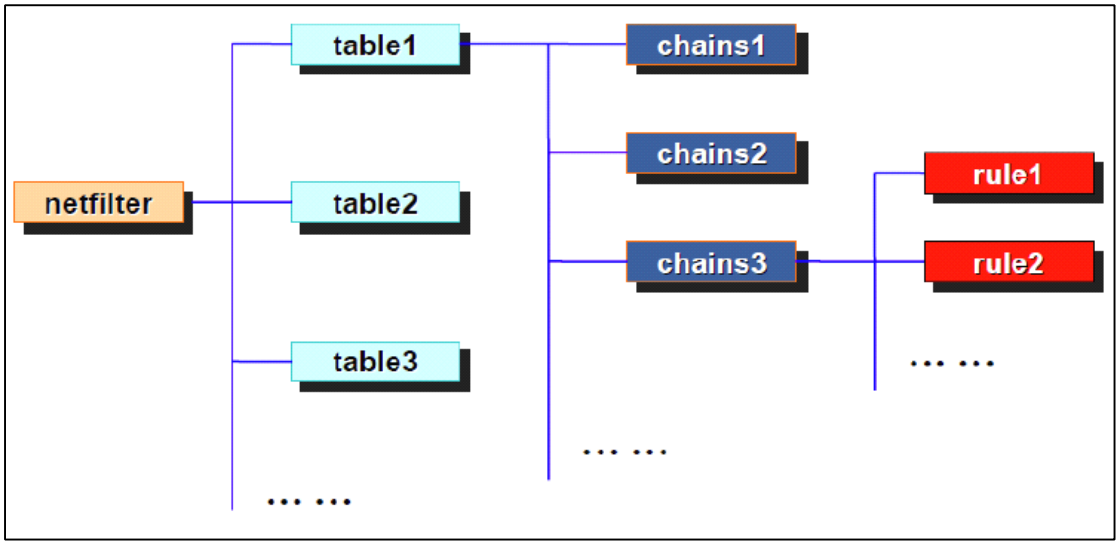
iptables -t filter -A INPUT -p tcp --dport 23 -j REJECT

iptables 语法

iptables [-t 要操作的表] <操作命令> [要操作的链] [匹配条件] [-j 匹配到以后的动作]

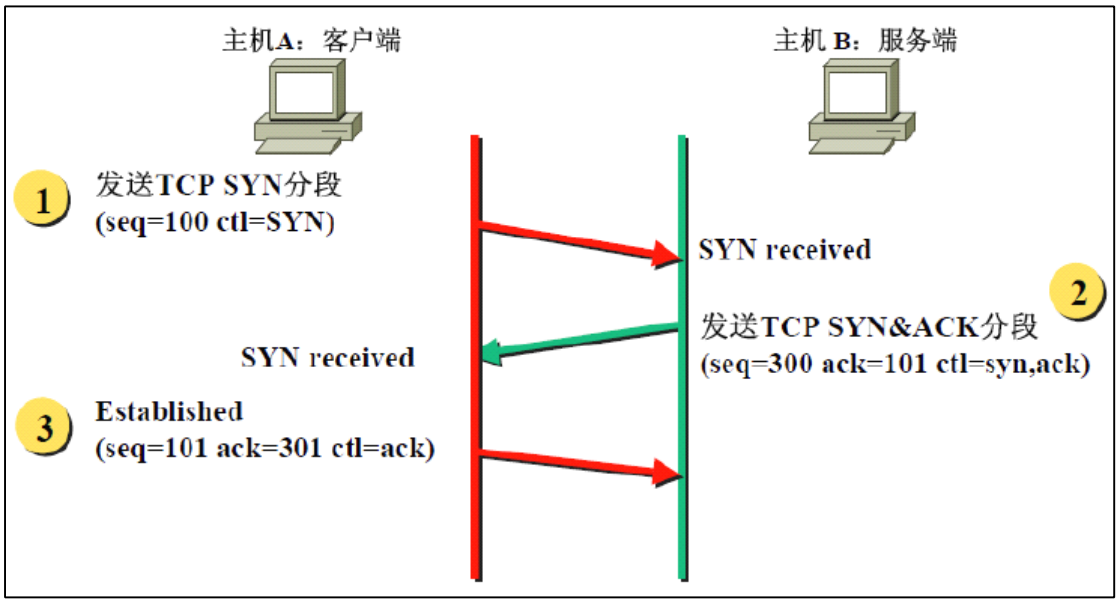
iptables 的表

包过滤中包含 3 个表
filter table ， 过滤表
nat ， 用于地址转换
mangle 表，俗称矫正表



filter 表

包含 INPUT、OUTPUT 和 FORWARD 链，用户处理输入、输出和转发包。
filter 表是缺省的表。



在我们使用 iptables something 的时候等同于下面的命令 iptables -t filter something

规则和链

每个链中的规则是按顺序的，处理一个包时，从第一条规则到最后一条规则，一次匹配。
顺序很重要
可以动态添加、删除和修改规则
可以查看当前规则

简单的 iptable 命令

iptables -F 清除所有的规则
iptables -L 列出当前所有的规则

iptables 命令

iptables -t filter -A INPUT -p tcp --dport 23 -j REJECT

红色部分定义使用的表
紫色部分定义匹配的规则
绿色部分定义了采取的措施

filter 表中可以采取的措施

ACCEPT 接受，等于不进行过滤
DROP 丢弃，弃之不理。别人可以判断出您的系统使用了防火墙
REJECT 弹回，貌似根本没有打开这端口
LOG 进行日志，/var/log/message

匹配条件

流入、流出接口（-i 、-o ）网卡
来源、目的地址（-s、 -d）IP
协议类型（-p）tcp udp icmp
来源、目的端口 （--sport 、 --dport）

简单地添加规则

根据源地址进行匹配的 -s 参数
[!] addr [net mask]
根据目的地址进行匹配 -d
[!] addr [net mask]

使用“!”的时候，需要在两端加空格

根据协议进行匹配的 -p 参数
[!] icmp
[!] tcp
[!] udp

根据端口进行匹配，这时必须指定协议，必须是 tcp 或 udp 协议。
根据封包来源的端口进行匹配的
--sport [!] port
--source-port = --sport
port 可以用 /etc/service 中的协议名来代替

根据封包的目的端口进行匹配
--dport [!] port
--dport = --destination- port
port 可以用 /etc/service 中的协议名来代替

INPUT 和 OUTPUT 的差别

对于 INPUT 而言
--dport -d 都是指你自己的端口和地址
--sport 和 -s 指的是发起连接者的端口和地址

对 OUTPUT 而言
--sport -s 都是指你自己的端口和地址
--dport 和 -d 指的服务器的端口和地址

任务：使得本机不能 ssh 到 192.168.9.50，分别使用 INPUT 和 OUTPUT 方法实现。

方法一：
iptables -t filter -A INPUT -s 192.168.9.50 -p tcp --sport 22 -j REJECT
方法二：
iptables -t filter -A OUTPUT -d 192.168.9.50 -dport 22 -j REJECT

方法一中，你发起的 ssh 请求被服务器所接收，但服务器返回给你的封包被 iptables 所阻挡。
方法二中，你发起的 ssh 请求本身就被 iptables 所阻挡，服务器当然收不到你的请求。

常用的参数

Iptables -A
增加一条规则，
iptables -t filter -A INPUT -s 192.168.0.1 -j DROP
iptables -t filter -A INPUT -s 192.168.0.2 -j DROP
用 iptables -L 可以看到两条规则
iptables -D 可以删除规则

方法一：
iptables -t filter -D INPUT -s 192.168.0.2 -j DROP
要和原来一样，但把-A 换位-D
方法二
iptables -D INPUT 2
可以删除第二条 INPUT 规则
iptables -D OUTPUT 1
可以删除第一条输出规则

iptables -L 命令
iptables -L ，列出所有 filter 表的规则
等同于 iptables -t filter -L
iptables -t nat -L 列出所有 nat 表的规则

iptables -L INPUT 列出所有 filter 表中的 INPUT 规则

iptables -F 命令
清除规则
iptables -t filter -F INPUT
iptables -t filter -F OUTPUT
iptables -t nat -F

对比性匹配的扩展

通过 -m 参数来调用 . 主要用法有
基于 Mac 地址的匹配 -m mac
基于封包数量的匹配 -m limit
基于 uid、gid 的限制 -m owner

基于 Mac 地址的匹配

格式：-m mac --mac-source mac-add iptables -A INPUT -p tcp --dport 23 -m mac --mac-source 00:0C:29:BC:BB:DB -j REJECT

仅仅对 PREROUTING 和 INPUT 链起作用

四个扩展匹配：MAC、用户和用户组、包次数、包状态

限制别人 ping

允许每秒通过的 icmp 包
iptables -A INPUT -p icmp -m limit --limit 1/s -j ACCEPT
超过部分全部拒绝
iptables -A INPUT -p icmp -j DROP

根据 uid 或者 gid 进行限制

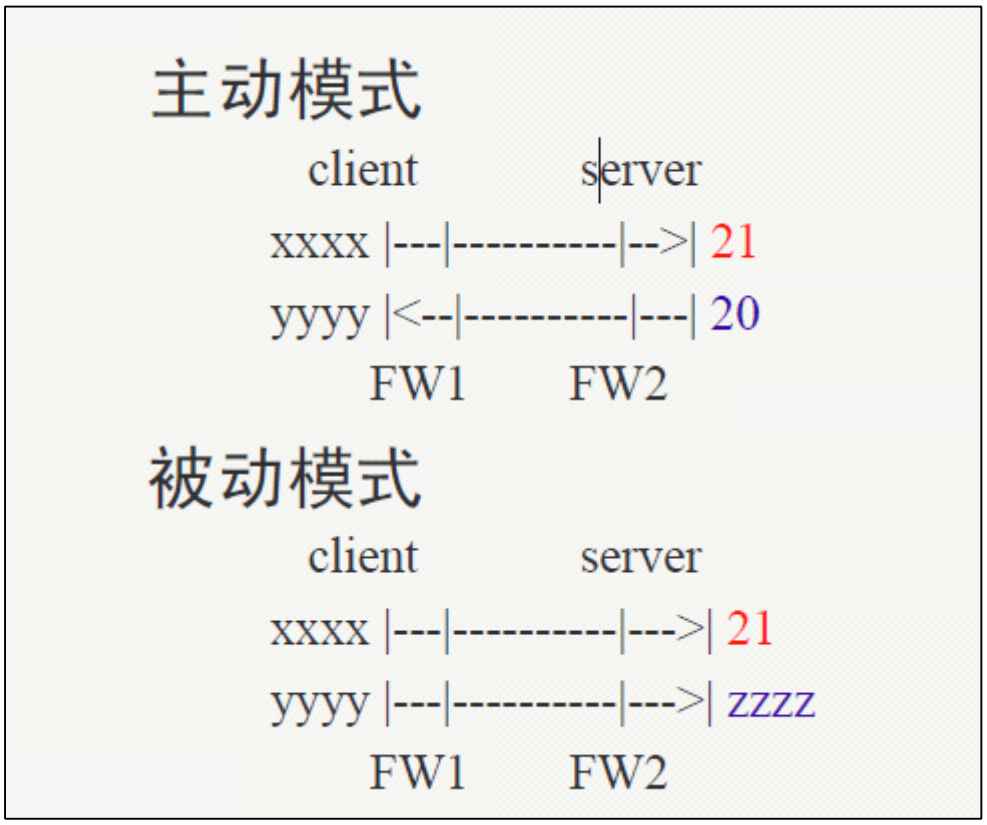
-m owner 参数
-m owner --uid-owner $AN_UID
iptables -A OUTPUT -p tcp --dport 23 -m owner --uid-owner 500 -j REJECT
iptables -A OUTPUT -p tcp --dport 23 -m owner --gid-owner 500 -j REJECT
注意： -m owner 仅仅对 OUTPUT 链有效

实例：一个典型的 web 服务器配置

```
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -p tcp --dport 80 -j    ACCEPT
iptables -A INPUT -p tcp --dport 22 -j    ACCEPT
iptables -A INPUT -p tcp --dport 873 -j    ACCEPT
iptables -A INPUT -p tcp --dport 139 -j    ACCEPT
iptables -A INPUT -p tcp --dport 21 -j    ACCEPT
iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -P INPUT DROP
```

数据链接四种状态：
1、NEW　建立连接（TCP SYN）
2、RELATED 在已经建立的连接中创建新连接
3、ESTABLISHED 已经建立完连接（完成三次握手）
4、INVALID 无效连接

ftp 协议传输原理



实例：禁止用户访问域名为 www.taobao.com 的网站。
iptables -I FORWARD -d www.taobao.com -j DROP
实例：添加 iptables 规则禁止 IP 地址为 192.168.0.200 的客户机上网
iptables -I FORWARD -s 192.168.0.200 -j DROP
禁止 192.168.1.0 子网里所有的客户机使用 FTP 协议下载
iptables -I FORWARD -s 192.168.1.0、24 -p tcp --dport 21 -j DROP

附表

- 图形远程管理工具：VNC、Xmanager for windows（商用）；
- 基于浏览器的模块化管理工具：Webmin；
- 命令行远程管理工具：Putty 、 SecureCRT、SSH Secure Shell Client、Winscp。

3.15-3.25 知识小结

Linux 基础知识

1、Linux 简介

X Window

2、Linux 安装

VMware 使用
分区 *1）文件系统结构 2）哪些分区单独划分
密码
安装软件包 server
时间 NTP
安全设置 SELinux Netfilter/Iptables

3、Linux 常用命令

文件命名规则

1）文件处理命令

文件基本权限设置 ls -l 、硬链接和软链接（创建、特点、inode）
touch/mkdir/cat/more/cp/mv/rm/cd/pwd
cp -Rpui rm -rfi - 二进制文件 d 目录 l 软链接 b 块设备 c 字符设备

2）权限管理命令

rwx 权限对于文件和目录不同的含义、缺省创建文件的权限不能有 x 可执行权限
chmod/chown/chgrp -R 继承 umask
基本授权：a. 授予一个用户 rwx 权限，可以更改所有者 b. 授予多个用户 rwx 权限，更改文件所属组对组授权

3）文件查找命令

find -name 文件名 -size 文件大小 -user/group 所有者所属组 -cmin 时间值 -inum i 节点 -type 文件类型 -perm 权限
stat 查看文件时间值
-a -o -exec -ok
which/whereis/grep

4）帮助命令

man/info 命令/配置文件，帮助分类，help shell 内置命令，命令 --help whatis/makewhatis
示例 sample/example 说明文档 README/INSTALL

5）网络命令

ping -c -s ifconfig/write/wall

6）关机重启

shutdown -h now/reboot init 0/6

7）压缩解压

特点、.gz .tar .zip .bz2

8）技巧

重定向、命令连接符、管道 *
命令替换符、别名、快捷键、命令历史记录、命令补全
rpm -e `rpm -qa | grep openssh`

4、文本编辑器 Vi

工作模式、常用操作、技巧（定义快捷键、连续行注释）、配置文件

Linux 系统管理

5、Linux 引导流程解析

引导过程 *
固件检测－加电自检－硬盘 MBR　软硬件时钟设置查看同步
　|
自举程序 GRUB　单用户模式、光盘修复模式 *　GRUB 命令操作、GURB 密码设置
/boot/grub/grub.conf
　|
内核　a. 驱动硬件　内核保存目录、内核版本　dmesg
　|
init　PID=1 内核调度器 PID=0　父子进程（孤儿进程、僵尸进程）
　|
/etc/inittab　文件格式　运行级别 *
　|
initdefault
　|
/etc/rc.d/init.d/rc.sysinit　系统初始化脚本
　|
/etc/rc.d/rc
　|
/etc/rc.d/rcN.d　如何更改自启动程序 *　ln -s、chkconfig、ntsysv　手动启动服务 *
/etc/rc.d/init.d　/var/log/messages
　|

/etc/X11/prefdm　　启动 X Window
　|
/etc/passwd
　|
/etc/shadow

6、软件包管理

RedHat 系 Linux 软件包管理方式：二进制包（rpm、yum）、源代码（configure、make、make install）、脚本安装（shell、java）
安装、升级、卸载：yum
查询、校验、提取文件：rpm
源代码特点、编译安装过程（编译工具）、卸载
脚本安装，查看 README/INSTALL
Debian 系软件包管理 dpkg、apt-get

7、用户和用户组的管理

授权：sudo 授权 *　普通用户授权 a. 授予一个用户 rwx 权限，可以更改所有者　b. 授予多个用户 rwx 权限，更改文件所属组对组授权
　　ACL 授权 *
SetUID/SetGID/粘着位 t　　权限的含义
用户配置文件 /etc/passwd *　/etc/shadow /etc/group *　/etc/gshadow /etc/skel /etc/motd
用户管理命令 useradd/usermod/userdel　groupadd/groupmod/groupdel
usermod -G 组名（多个组） 用户名
gpasswd -a 用户名 组名
手动添加用户/用户组，手动删除用户/用户组
pwconv/pwunconv/vipw/vigr/pwck/grpck/id/groups/newgrp/finger/who/w/su

8、进程管理

父子进程/前台后台进程/程序和进程区别
周期性计划任务 crontab *
一次性计划任务 at/batch
进程管理命令 w/uptime/ps/kill/xkill/pgrep/pkill/nice/renice/nohup/ctrl+z/jobs/fg/bg/top
进程处理方式：a. standalone b.xinetd (/etc/xinetd.d) c. 计划任务

9、文件系统管理

添加硬盘分区（分区 fdisk、创建文件系统 mkfs/mkswap、写入/etc/fstab） *
磁盘配额（开启分区配额 userquota、生成配额数据库文件 quotacheck、启动配额 quotaon、编辑配额 edquota、查看 quota/repquota） *

/etc/fstab 基于文件系统权限设置 ro/nosuid/noexec/acl/userquota
文件系统结构
文件系统管理命令 df/du/stat/md5sum/fsck/e2fsck
备用恢复 备份意识-权限-压缩-校验-加密 cp/tar

知识补充

这里大部分是原来使用 linux 中遇到的问题，一起总结下。

1.更改 telnet 登录慢的问题【telnet 不推荐使用】

vi /etc/resolv.conf

将里面内容清空

resolv.conf 是一个域名解析器使用的配置文件（一个根据主机名解析 IP 地址的库）

登录慢是因为域名解析和反向解析而导致的，当在使用 telnet 登录时间很长，而登录完成后

速度正常，这就是在试探域名反解造成的，未配置域名反解导致必须等 timeout 才能登录，这就是登录慢而登录后正常的原因。

2.更改 root 用户不能直接远程登录问题

vi /etc/pam.d/login

注释掉这行

#auth required pam_securetty.so //该行注释掉

3.禁止在后台使用 CTRL-ALT-DELETE 重起机器

cd /etc/inittab

vi inittab 在文件找到下面一行

Trap CTRL-ALT-DELETE

ca::ctrlaltdel:/sbin/shutdown -t3 -r now （注释掉这一行）

如： # Trap CTRL-ALT-DELETE

#ca::ctrlaltdel:/sbin/shutdown -t3 -r now

4.显示系统运行了多长时间

uptime

5.查看端口

netstat –an

netstat -anp

6.端口的详细列表

/etc/services

7.查看物理信息

lspci

8.看已经安装的字符集

locale –a

9.抓包命令 tcpdump

例：抓获 192.168.1.32 的主机收到和发出的所有的数据包

tcpdump host 192.168.1.32

截获特定的端口

tcpdump tcp port 21 host 192.168.1.32

10.查看端口现在运行什么程序

lsof –i:8001

11.察看实时的日志

tail –f /var/log/messages

12.当 mount 出现死的现象

1.fuser –m /mnt/share 查出该程序的进程，然后 Kill 掉

2.再 umount /mnt/share

13.Linux 挂载 Windows 分区

mount ntfs 分区

一.单机挂 windows 的 NTFS 分区

1. 上 www.google.com 搜索并下载 kernel-ntfs-2.4.18-14.i686.rpm

2. rpm -ivh kernel-ntfs-2.4.18-14.i686.rpm

3. mkdir /mnt/share

4. mount -t ntfs /dev/hda1 /mnt/share

要挂载 Windows 分区，首先新建一个目录/mnt/share，修改/etc/fstab，在最末尾添上（假设

Windows 安装在硬盘的第 1 个分区）

/dev/hda1 /mnt/share ntfs defaults 0 0

二.网络上一台 windows 和 linux 机器，linux 机器挂载 windows 上的共享文件

windows IP:192.168.1.1

1.linux 挂载 192.168.1.1(windows)上共享文件 dbf,挂在 linux 的/mnt/share 目录下,在/mnt 下建

立 share 目录

mount -t smbfs -o username=massky,password=massky //192.168.

1.1/dbf /mnt/share

2.机器重启自动挂载，vi /etc/fstab 最后加入：

//192.168.1.1/dbf /mnt/share smbfs defaults,auto,username=m

assky,password=massky 0 0

14.网卡的激活与停止

超级用户;

ifconfig eth0 down 停止

ifconfig eth0 up 启动

15.查看是否网络环境

ethtool 可以查看网络环境

如：ethtool eth0

16.Linux 下 cvs 的安装配置

1.安装 CVS 软件包.

2.groupadd cvs

3.useradd -g cvs cvsroot

4.chmod 777 -R /home/cvsroot

5.cd /etc

6.vi profile

新增以下二行:

CVSROOT=/home/cvsroot export CVSROOT

CVSEDIT=vi export CVSEDIT

7.查看/etc/services 文件中 cvspserver 所在行的注释状态(有则把注释去掉)

8.进入/etc /xinetd.d, vi cvspserver 该文件不存在,内容如下:

service cvspserver

{

disable = no

socket_type =stream

wait =no

user =root

env =HOME=

server =/usr/bin/cvs

server_args =--allow-root=/home/cvsroot pserver

log_on_failure +=USERID

}

9.vi /etc/xinetd.conf 内容如下,每次开机自动启动服务:

service cvspserver

{

port = 2401

socket_type = stream

wait = no

user = root

server = /usr/bin/cvs

server_args = -f --allow-root=/home/cvsroot pserver

bind = 168.168.1.110

}

重新登录换 cvsroot 用户

10./etc/init.d/xinetd restart

11.cvs init（初始化：CVS 版本库的初始化）

12.cvs -d :pserver:cvsroot@192.168.1.110:/home/cvsroot login(用户登录）没有任何提示信息就

成功。

13.首先要导入库,假设项目名称为 LinuxISQuote.

步骤：

1、进入 LinuxISQuote.

2、一个项目的首次导入

cvs import LinuxISQuote lch V_0_0_1

此时到$CVSROOT 目录下,可以看到多了一个 LinuxISQuote 的目录。

17.修改 grub 启动时的背景图片

1 将一图片转化成 640*480,14 色的 XPM 文件：

#convert abc.jpg -colors 14 -geometry 640x480! abc.xpm

2 压缩生成的 xpm 文件，使用 gzip

#gzip -9 abc.xpm

3 将 abc.xpm.gz 拷到/boot/grub 下

4 修改/boot/grub/menu.lst

splashimage=(hd0,0)/boot/grub/abc.xpm.gz

18.显示硬件信息

lsdev

19.显示当前加载的核心模块

lsmod

20.列出系统内核所有可用的模块

modprobe –l

21.根据进程名显示进程号

pidof vsftpd

22.查看密码过期信息

change –l longinname

23.显示最后一个登录到系统的用户

last

24.以 3 秒钟执行一个 ls 命令

watch –n 3 ls

25.编译内核的步骤

1.源码/usr/src/linux-2.4

2. make mrproper (清除从前编译内核时残留的.o 文件和不必要的关联)

3. make menuconfig (字符界面内核配置菜单中正确设置个内核选项)

make xconfig (图形界面内核配置菜单中正确设置个内核选项)

4. make dep (设置关联文件)

5.make bzImage (对于大内核,如需要 SCSI 支持的编译)

make zImage (对于小内核的编译)

6.make modules (编译模块)

7. make modules_install (安装模块)

8.make install (针对 grub 启动，自动装载到 grub.conf 上，直接重新启动就 OK）

26.查询一个系统最近何时被引导过

who -b

27.查系统硬件类型

uname -m

28.查系统的 CPU 类型

uname -p

29.查系统 OS 版本号

uname -r