du -sh [filename]单个文件大小

vim ~/.bash\_history

screen -D -r qiz

c4dev@sles12-liux14-dev:unity (integration/nighthawk)> git status -uno

/c4\_working/unity/output/image/GNOSIS\_DEBUG/

more /proc/meminfo

more /proc/cpuinfo

cat /proc/version

df –h查看设备使用情况

uname -a

c4\_working (10.244.197.9 (Samba 4.1.12-3.2-3293-SUSE-SLE\_12-x86\_64))

## 文件挂载

**首先执行id 出dev查看uid和gid。**

uid=0(root) gid=0(root) groups=0(root)

uid=1002(c4dev) gid=100(users) groups=16(dialout),33(video),333(docker),100(users)

mount -t cifs -o username=qiz,rw,uid=1002,gid=100,dir\_mode=0777,file\_mode=0777 //10.244.103.204/share /mnt/myshare

umount /mnt/myshare

第5章

process and program

!+历史纪录命令的第一个字母，就可以直接执行

IDE hda1 hda2

SATA SCSI sda1 sda2

eth0 eth1表示网卡个数

以点开头的文件都是隐藏文件

ctl alt f1-f7 f7是GUI

tty查看在哪个界面

ctl L清屏

--表示该选项是多个字母

cal显示日历

cal 2012

cal 12 2012

bc命令使用scale=4设置保留的小数，quit退出

ctl C不执行当前输入

history显示最近的1000个命令，使用!n引用

ctl 左右方向键，每次跳一个单词

ctl K删到最后

Ctrl U清除到开头

esc 再按点应用上一个命令的参数

选中之后，按鼠标中间键就直接粘贴了

ctl D停止输入或exit

ctl shift T创建终端，ctl D关闭终端

为了控制滚屏，可以按Ctrl+S键，停止滚屏；按Ctrl+Q键可以恢复滚屏

Ctrl E 到结尾

Ctrl A 到开头

ctl J 执行命令

命令是区分大小写的

whatis 只是显示该命令的作用，也没有属性介绍

[tom@localhost ~]$ whatis passwd

passwd (5) - (未知的主题)表示在man的第五章

sslpasswd (1ssl) - compute password hashes

passwd (1) - update user's authentication tokensman cd

man 5 passwd表示打开第五章的man page

用法：ls [选项]... [文件]...省略号表示表示有多个选项

man –k abc 模糊查找命令，k表示keyword

man hosts.allow可以查看系统文件的使用，不需要加路径

shutdown reboot halt poweroff

/etc/rc.d查看服务，查看目前在哪个init？

init 0是关机

init 1单用户模式，可以也可以找回root密码

init 6重启

init 5 GUI

init 4 未定义

在级别3可以使用startx命令开启GUI

每个init有7个tty控制台，在第5级别，GUI运行在第7个控制台

第6章

/etc/passwd用户名

/etc/shadow用户密码

/etc/group用户组

ln –s res des创建快捷方式s表示软连接

chmod更改文件权限chmod a+r filename表示对ugo操作相当于chmod ugo+r filename

ls –ld dir查看文件夹的属性，不加d是查看目录里的东西

chmod XXX filename只该表文件自身，不改里面的内容

chmod -R XXX filename 也改变里面的文件

chown username file 改变文件的所有者改为username

chown username:group file 改变文件的所有者改为username,所属组改为group，使用chown username.group file点也可以

chgrp group filename改所属组，可以加-R

x权限，对文件来说表示是否可以执行它，例如脚本文件。对文件夹来所表示是否可以cd进入文件夹

Linux最后扩展名不重要。使用file filename命令可以查看文件类型，除了一些特殊的文件需要特定的文件名

bin类似system32

boot系统的启动文件，和内核

dev，设备文件

etc配置文件

home用户的家目录

lib链接库文件

media默认挂载点

mnt挂载点

opt安装软件目录

proc内核参数

root

sbin和bin类似，system bin

usr软件默认安装位置

var日志，缓存，数据库日志

直接输入cd进入家目录，cd –进入上一个目录

cd ~tom切换到tom的home目录，跟当前用户无关

mkdir创建文件夹

mkdir –p a/b 如果parent文件夹不存在则要加-p

mkdir –p a/{b,c,d}创建a下面b，c，d多个文件夹

rmdir –p a/b 删除b之后，如果a为空则a也被删除，否则不会被删除

rm –rf强大的删除命令

$PATH环境变量，可以直接修改。PATH=“XXXX”来修改

ls –s显示大小，-l长模式，-h，-d表示查看文件夹的信息不会显示文件夹里面的文件，ls不会显示文件夹里的大小

ls –R递归显示所有

cp /a/b /c/d 如果d是文件夹则好办，如果不是或者是一个不存在，则把b拷过去并且重命名为d，其中b不能为文件夹

cp –rf 可以拷贝文件夹，-p拷贝文件信息，例如创建日期等

cp –a可以做到cp –rfp

cp拷贝链接文件时默认拷贝的是连接的源文件，加入-d才是拷贝链接文件自身

父目录不存在?

mv是剪切,重命名

alias rm别名查看

查看文件内容的命令cat，tac，head，nl，tail，more，less，od

more不能向前翻页

less可以向前看，end跳到文档开头，end到最后，支持查找，n向下找，N向上找

head默认看前10行

tail默认查看尾10行

tail –f检测文件，实时查看文件结尾变化，等同于tailf

od –t c /bin/ls查看二进制文件

strings /bin/ls

touch 如果文件存在，则会把文件的修改时间修改为当前时间，-t可以修改为其它时间

创建文件的默认权限时664-rw-rw-r--，文件夹是775 drwxrwxr-x

umask 默认为0022，使用umask –S显示保留那些权限

lsattr查看隐藏文件属性

chattr 来改变隐藏属性，+a只能添加不能删除（对文件夹）

+i不能添加也不能删除

hostname命令查看主机名 echo “aaaa `hostname` aaaaa”或

echo “aaaa $(hostname) aaaaa”

-rwsr-xr-x 1 root root 47032 Jan 27 2016 /usr/bin/passwd\*

s（s在u位上）表示其它用户使用该文件时具有所有者的权限

s（s在g位上），则任何用在该文件夹下户创建的文件都是该文件所属的组

t权限只能在o位上，该文件只有root和自己才能删除，其它用户删除不了，即使其它用户有权限

which ls

whereis ls包含帮助文档

locate filename

find 目录 –属性 值 -i忽略大小写

默认在当前目录查找

属性：name，size，user，perm（权限）

find /etc –iname tom;find / -user root –group root

find / -user root -or –group root

dd if=/dev/zero of=file1 bs=1M count=1

find / -size 2M

find / -szie +2M

find / -szie -5M

find / -szie -5M –size +2M

find –ctime +1 大于一天

find –cmin +1 大于1分钟

find –newer file1比file1要新的文件

find /etc –perm 222精确匹配必须是--w--w-w--

find /etc –perm +222，ugo中只要有w即可，是或的关系

find /etc –perm -222，ugo中每个必须至少有w，是与的关系

id username查看用户UID，echo $UID

su – username切换用户

第8章磁盘文件存储

ls –i filename查看文件的inode，使用filefrag –v filename 查看文件占用的block

ext3比ext2多了日志功能

dumpe2fs /dev/sda1 查看分区信息包括inode，block等信息

du –T可以显示分区类型

du –s只显示该目录的大小而不显示文件夹中每个文件的大小

df –hT /dev/sda1查看文件分区信息挂载点和使用情况

fdisk –l /dev/sda查看硬盘信息，是硬盘名不是分区名partition分区

划分分区之后需要更新分区表，partprobe，还要格式化 mkfs –t ext3 /dev/sda5 或mkfs.ext3 mkfs.vfs

先卸载分区fsck /dev/sda5

badblocks –sv /dev/sda5

mount也可以查看挂载点

mount –o remount,{ro,rw,noexec,exec} /mnt

fuser –mv /mnt查看那个进程在使用，导致无法umnout

开机自动挂在vim /etc/fstab需要修改

<file system> <mount point> <type> <options> <dump> <pass>

/dev/fd0 /media/floppy0 auto rw,user,noauto,exec,utf8 0 0

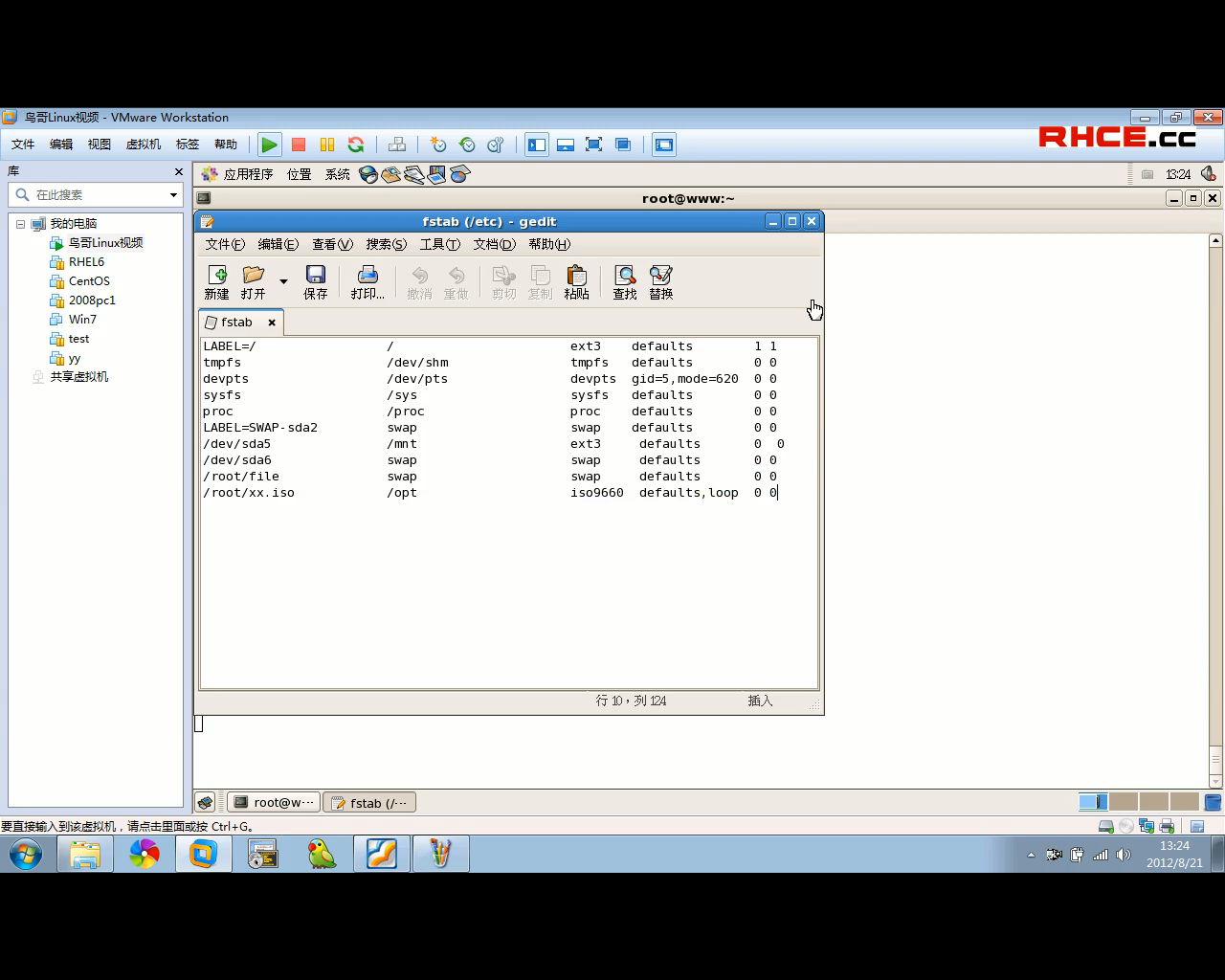
使用tune2fs –l /dev/sda5查看分区信息，当挂在次数达到最大挂在次数时会系统会强制自检这个分区

e2label /dev/sda5 xx 设置卷标

挂载镜像文件mount –o loop xx.iso

cat /proc/swaps查看交换分区

也可以创建一个文件作为swap



第九章压缩与解压

-v显示过程

-c保留源文件

gzip res直接压缩不保留源文件

gzip –c res > des.gz 保留源文件

zcat res直接查看被压缩文件的内容，gzip –l查看哪些文件被压缩了

gzip –d res解压

gzip不能对目录压缩，所以先要把目录打包成tar格式

bzip2也是同样的命令，但后缀为bz2

bzcat查看压缩文件里的内容

zip des res压缩，后缀为zip

tar是打包（归档）工具，前面是压缩的，它自身是没有压缩功能的，打包之后大小可能会变大

tar c（创建）v（显示过程）f（指明打包后的文件） des.tar res此时是保留源文件的，不保留源文件*--remove-files(files)*

tar tvf des.tar 查看归档后的内容而不解档

tar xvf des.tar解包到当前，-C /root 解档到指定目录

tar xvf des.tar aaa只解档里面的aaa文件

tar zcvf des.tar.gz res既压缩又归档，调用gzip压缩，或tgz结尾

tar zxvf res.tar.gz解压解档

tar jcvf des.tar.gz.bz2既压缩又归档，调用bzip2压缩

tar jxvf res.tar.bz2解压解档

dump -0u –f des res将res的文件备份到des中，0表示全备份，f表示指定目录

dump -1u –f des res

dump -2u –f des res 0-1-2表示增量备份。0-9-8-7差异备份

进入到要回复的目录，即des，使用restore –r –f /root/tmp/1.dump来恢复文件

restore –i –f /root/tmp/1.dump可以选择地恢复部分文件，add files，然后用extract来执行恢复

mkisofs –o name.iso res1 res2 res3把这3个文件打成name.iso文件

mount –o loop name.iso /mnt挂载iso文件

cp /dev/cdrom abc.iso可以直接把cdrom中的文件拷贝为iso文件

mount –o loop abc.iso /mnt然后挂载它

dd if=/dev/sda1 of=test把sda1中所有的数据放到test中

dd if=test of=/dev/sda1

dd if=/dev/sda1 of=/dev/sda4把sda1中的数据放到sda4中

管道和cpio打包

find –name \*|cpio –o > xxx.cpio –o（打包）（-i解包）

cpio –iu <xxx.cpio解包

vim ~.vimrc 加入set nu显示行号

unix2dos 把Linux的文件转化为win下可以看的格式

dos2unix把win转化为Linux可以看的格式

第十一章shell

cat /etc/shells查看系统支持哪些shell

直接输入csh切换到csh

cd是shell的命令，不是系统命令

type cd查看命令的出处

同一行多个命令用分号分开

aaaaaaaaaaaaaaaaaaaaaaaaa \

bbbbbbbbbbbbbbbbbbbb\

cccccccccccccccccccc来换行

定义变量时=两边不能有空格，unset var取消变量

直接输入bash可以切换shell进程使用echo $$输出PID查看

export定义的是环境变量，它会影响到子shell，在子shell中可以使用export变量

set会显示所有变量包括本地变量，env，export，export –p都可以显示环境变量

declare –x a=10也可以定义环境变量，declare +x a取消

在用户还没有登录时，$USER，$HOME是没有值的，切换用户时也是对这些值的初始化

常见环境变量

$HISTFILESIZE

$PS1（[\u@\h \W]\$ W表示短目录，w表示全目录）[root@roor home]$

$PS2

locale –a查看系统支持的编码

$LANG查看当前的编码

read后面没有参数时，默认赋值给了$REPLY

数组x=(123,aaa,bbb) echo ${!arr[\*]} 输出数组编号 0 1 2 3 4…

declare –r aa只读变量readonly aa也可以，也是无法unset的

xx=只声明没有赋值

${arr#tmp}从arr中从前往后删除tmp，只删除第一个

${arr##tmp}从arr中从前往后删除tmp，只删除最后一个

%从后往前删除第一个,%% 从后往前删除最后一个

设置别名：alias a=”grep -nr”，取消unalias a

alias可以列出所有的别名

alias echo=“echo -n”

修改/root/.bashrc使alias永久生效

type –a echo可以看出执行的先后顺序

/etc/issue更改登陆前的提示信息 /d表示当前时间

/etc/motd登陆后的信息

/etc/issue.net表示telnet登录的提示信息

一登录执行/etc/profile，/etc/inputrc, /etc/profile.d

然后进入home目录执行.bash\_profile

set +u不提示与-u相反，可以先设置set –x然后再执行脚本

通配符：

一对[abcd]匹配一个字符，abcd只匹配一个，[a-z]a到z，[a\-z],a,\,z

[0-9]

?一个任意字符

[^0-9]或[!0-9]不是数字

ls [0-9].mp3

&>正确的和错误都会进行重定向,&>>不行

tr ‘a-z’ ‘A-Z’ < file 可以把file中的字母变为大写

tr –d ‘:’ file 删除文件里的冒号

cat > file <<end(支持变量)

cmd1;cmd2

cmd1|cmd2

cmd1|tee [-a] file1 |cmd2

cmd1&&cmd2

cmd1||cmd2

cut –d: -f1 /etc/passwd d表示分隔符，f表示第几部分

grep –i忽略大小写

-n加入行号

-v排除关键字的行

-A3下面3行，B3上面3行

sort默认按照字母顺序

sort –n按照数字排序，-r反向排序，-u排序重复的行

sort –t: -k3 file t表示分隔符，k第几部分

unique –c统计重复出现的次数

wc filesname 行数 单词数 字符数

wc –l filename行数

wc –w filename单词数

wc –c 字符数（包含换行）

ls -1（一行一个显示）/root |wc –l 统计文件夹里有多少个文件

tr ‘a-z’ ‘A-Z’ < hosts >newfile把hosts改为大写，两条命令

mv newfile hosts

cat –A可以显示Tab

^I表示Tab键

col –x file表示把tab换成空格，或expand file

paste file1 file2按行粘贴

join默认每行开头相等才能结合在一起，-t分隔符

join –t “:” file1 file2

join –t “:” -1 4 file1 -2 3 file2 file1的第4部分与file2文件的第四部分一样再结合

split分割文件

split –b 10k res des分割后每个文件以des开头

cat des\* > des重新组合文件

split –l 100 res des每100行分割成一个文件

不是所有的命令都支持管道

find –name ‘file\*’ | ls –l find找到的结果不支持管道，使用xargs即可

find –name ‘file\*’ | xargs ls –l

touch file{1,2,3,4}会生成file1，file2，file3，file4,4个文件

减号-的作用

tar cvf - /home | tar xvf - 先打包成一个文件然后再解包，减号代替了中间文件

第十三章

grep –p不显示结果，可以用来判断文件中是否包换有某个字符串

第十四章用户，组管理与ACL（访问控制列表）

用户管理：

1-499是系统用户

500-65535普通用户

groups username查看用户所在的组，一个用户可以属于多个组，第一个是默认组，不加username查看本用户

gpasswd –a username groupname把用户加入到组中

newgrp groupname临时切换组，exit退出

使用root用户，useradd bob

useradd –D查看创建用户的默认信息

vim /etc/default/userdd修改创建用户时的默认值

-c描述

-s /sbin/nologin指定shell

-d指定家目录

-g指定主组，-G附属组

-u指定UID

usermod –s /etc/bash bob更改用户信息

grep bob /etc/bash

passwd usrname根据提示设置密码

passwd –S username查看密码状态

passwd –d username删除密码

echo “aa” |passwd --stdin bob设置bob密码为aa

usermode –l username锁定密码

-u解锁

userdel –r username删除用户

change该修改密码更改时间设置

finger username也可以查看用户信息

chfn username也可以更改用户信息

id username查看用户id信息

组的管理：

groupadd name

groupdel name

groupadd bob添加组

groupdel bob删除组

gpasswd –a tom bob将tom添加到bob组中

gpasswd- d tom bob将tom从bob组中删除

可以将组的权限给组里的成员，即组长

gpasswd –A tom bob即tom为bob组的组长，可以对组添加和删除成员

getfacl filename查看文件的权限

setfacl –m（modify）u:tom:rw- file,把tom的权限更改为rw-，覆盖原来的权限

使用acl之后，ll命令文件文件描述符有个+

setfacl –x u:tom file 去除tom文件的acl权限

把u换成g则是对组操作

setfacl –m –d:u:bob:rw- file 其中d表示default，bob对在file中文件有个指定rw-的默认权限，不管是谁创建的

setfacl –x d:u:bob file 取消，之前已经有的某些文件权限还是保留的

用户切换

su tom切换用户，但不会切换家目录，但是环境还是上一个用户的环境

su – tom家目录也会切换

root切换到其它用户都是不要密码的

su – 默认切换到root

普通用户没有mount权限

su - -c ‘mount xxx’ 使用root来执行这个命令，需要输入密码

su - bob -c ‘mount xxx’ 使用bob来执行这个命令，需要输入密码

sudo

/etc/sudoers文件用来保存哪些文件有哪些权限

tom [www.baidu.com=/bin/mount](http://www.baidu.com=/bin/mount) 即tom在百度主机上可以使用mount命令

还是要使sudo – tom mount命令来执行，要输入tom的密码，虽然当前用户是tom也要输入密码

密码会有一定的缓存时间，sudo –k清空缓存

sudo –l列出当前用户可以使用哪些sudo

tom [www.baidu.com=NOPASSWD/bin/mount](http://www.baidu.com=NOPASSWD/bin/mount)不需要输入密码

/sbin/nologin提供服务不可登录

chsh –s /sbin/nologin tom

服务收到请求会发送给PAM

pam的配置文件/etc/pam.d

/lib/sercurity/所有的pam模块

w=who查看当前系统哪些用户已经登录

last显示登录用户的登录历史

lostlog也可以。/var/log/lastlog

write tom tty2可以向tom发送消息，ctrl d提交

mesg y，mesg n开启，关闭，root特殊

pwc查看系统的默认信息，比如用户的家目录

visudo，

vipw编辑passwd文件

第15章磁盘配额quota，高级文件管理

watch命令以周期性的方式执行给定的指令，指令输出以全屏方式显示。

语法

watch(选项)(参数)

选项

-n：指定指令执行的间隔时间（秒）；

-d：高亮显示指令输出信息不同之处；

-t：不显示标题。

逻辑卷管理器LVM（logical volume manger）

做快照

第16章例行工作crontab

service atd status必须有

at + 时间(MMDDYY,更多格式查看man at)

输入命令，最后按ctrl d

at 12132012

>ll

atq或at -l查看计划列表

at –c 2查看第二个计划

atrm 2或at –d 2删除第二个任务

出现在/etc/at.deny中的用户不能使用

/etc/at.allow中的用户可以使用，若都出现则也被允许

crontab –l查看计划列表，-u指定用户，默认是自己

crontab –e 编辑计划任务，#注释

分 时 天 月 周 命令

0 7 \* \* 1-5 ls

0-30/5 7 \* \* \*

0到30分钟，每5分钟执行一次

30 9 1-10,15 2 \* zz

30 9 1-10,15 2 1,3,5 zz 2月份的星期1,3,5,2月份的1到10号和15号

周与月天是或的关系

crontab –r删除

/etc/crontab系统计划任务

# For details see man 4 crontabs

# Example of job definition:

# .---------------- minute (0 - 59)

# | .------------- hour (0 - 23)

# | | .---------- day of month (1 - 31)

# | | | .------- month (1 - 12) OR jan,feb,mar,apr ...

# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat

# | | | | |

# \* \* \* \* \* user-name command to be executed

0anacron 也会被执行

开机时检测anacron与当前时间，这样就知道错过的crontab

/var/cron.daily/

/etc/anacrontab anatab来不起计划任务的任务表

第17章

pidof ping 可以直接查看进程的pid

pgrep –U root 查看root的进程

killall 进程名

kill是加pid

top默认3秒刷新一次

top –d 1改为1秒

top wa表示系统输入输出

PR优先级，数字越大级别越低，pr=p（优先系数）+nice

nice可以在top中更改，nice默认为0

优先级从-20到19

cat /dev/zero > /dev/null

top

按r输入pid

再输入renice值

或者renice -20 pid更改正在运行的进程

nice –n -19 cat /dev/zero > /dev/null

指定初始nice值

free –m单位为MB

vmstat 1 10每隔1秒执行一次，执行10次

pstree –u进程树 和用户

/etc/fstab

SELinux

DAC

selinux使用MAC，进程被称为主题，资源被称为对象

ls –Z file

seinfo –u查看系统的用户，system-u，user-u代表其他进程

seinfo –r

object-r代表文件或目录

system-r代表进程

ps –auxZ

不可能一一指点每个进程的资源，进程要被指定策略，策略中规定了可以访问哪些资源。

seinfo查看策略，上下文

sestatus查看selinux系统的状态，工作模式（强制，允许，禁止）

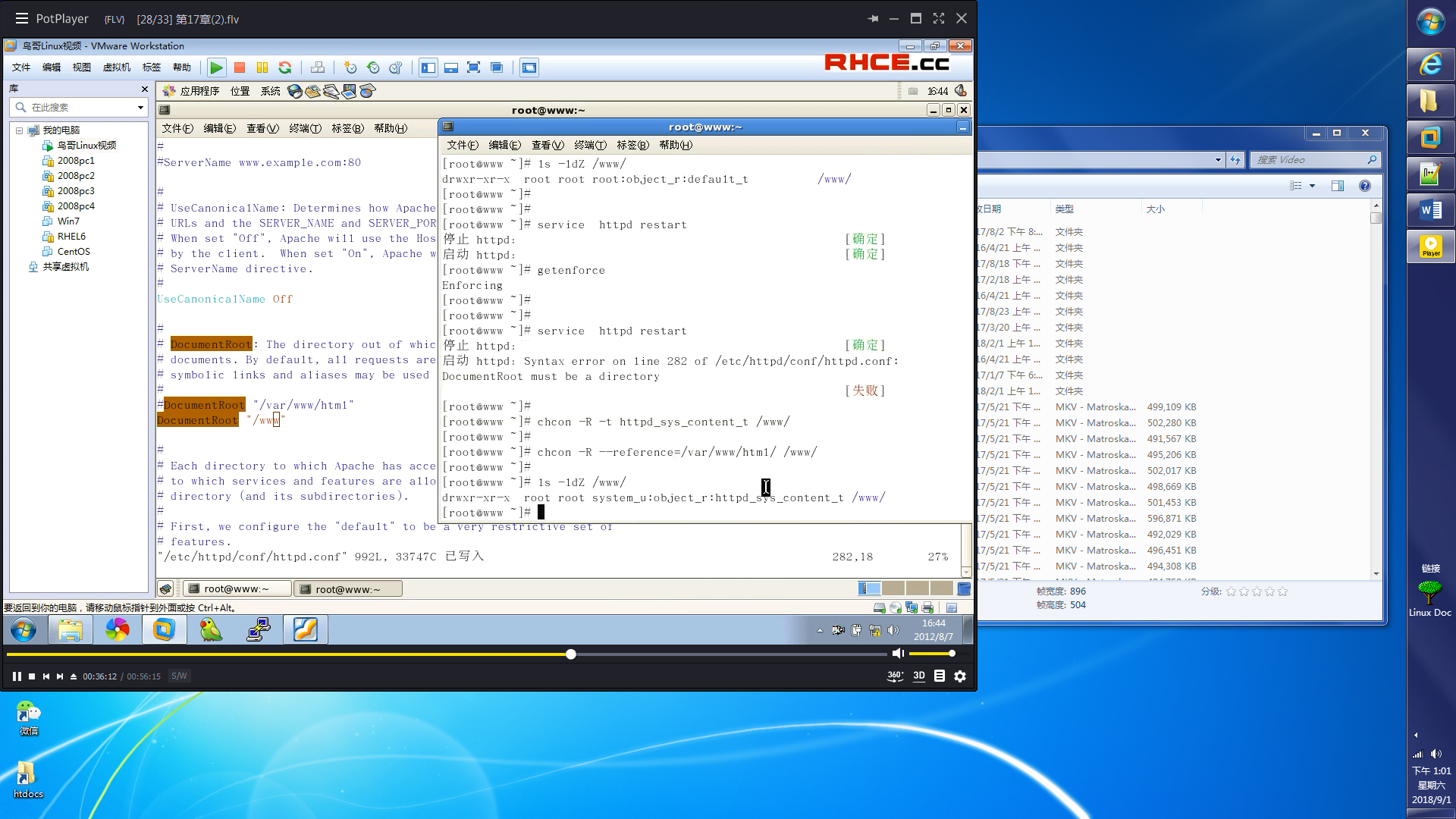
getenforce查看模式

setenforce 1强制，0允许，这两种模式都不能切换到“禁止模式”

/etc/selinux/config设置默认模式和策略

命令行只是临时有效，开机会变为默认模式

禁止模式意思是关闭selinux



restorecon -R dir更改为默认值

set

audit2why < /var/log/audit.log分析日志

setsebool –a查看系统配置，或seinfo –b

set boolname on/off开启关闭se的配置，

或set boolname=0/1

-P写入配置文件，重启也会生效

第18章认识系统服务（daemon）

独立服务操作：

service servicename restart/start/stop 不是配置开机也会启动

service servicename status

或/etc/init.d/network restart

非独立服务：

xinted是独立服务，管理了许多非独立服务

ls /etc/init.d/查看独立服务

ls /etc/xinet.d 查看非独立服务

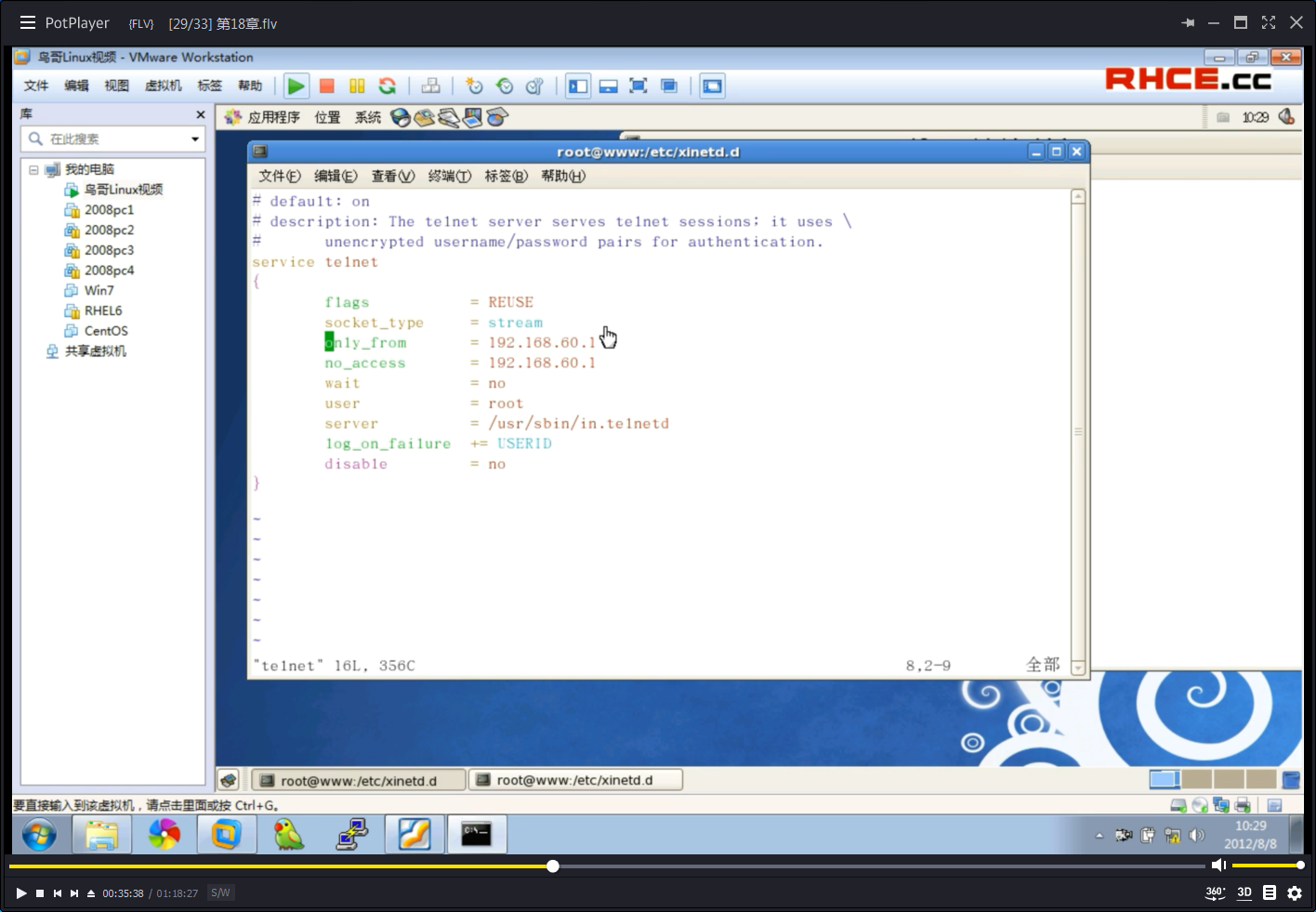
/etc/xinetd.conf

cps=50 10，防止dos

instance=50最多50个连接

per\_source=10

如果匹配冲突，还是不允许



rlogin和telnet都是非独立服务

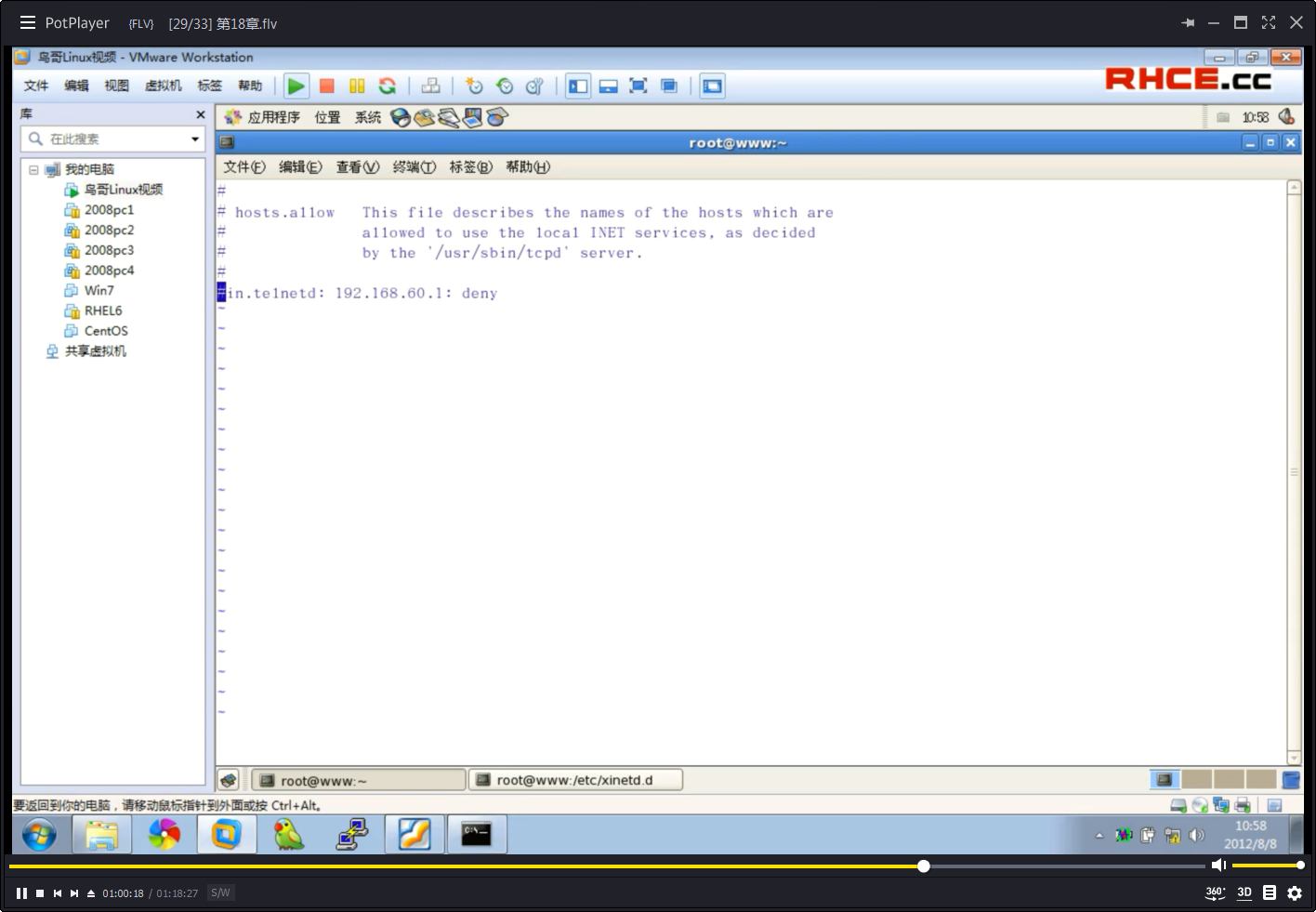
安装rlogin和telnet都需要重启xinetd

/etc/hosts.\*

/etc/hosts.allow这个匹配成功则不会匹配deny

/etc/hosts.deny 某个网段10.10.1.

cat 可以显示多个文件



如果在文件中加入选项，是选项生效

netstat –nutlp查看系统有哪些监听程序

setup

ntsysv

chkconfig --list查看开机启动

chkconfig影响的是开机启动项，service是临时生效

chkconfig [--level 5] NetworkManager off关闭开启服务

非独立服务根据xinetd启动

chkconfig xinetd restart

第19章日志文件

service syslog restart开启

/etc/syslog.conf配置文件

事件.级别

/etc/logrotate.conf

/etc/logtrtate.d/syslog

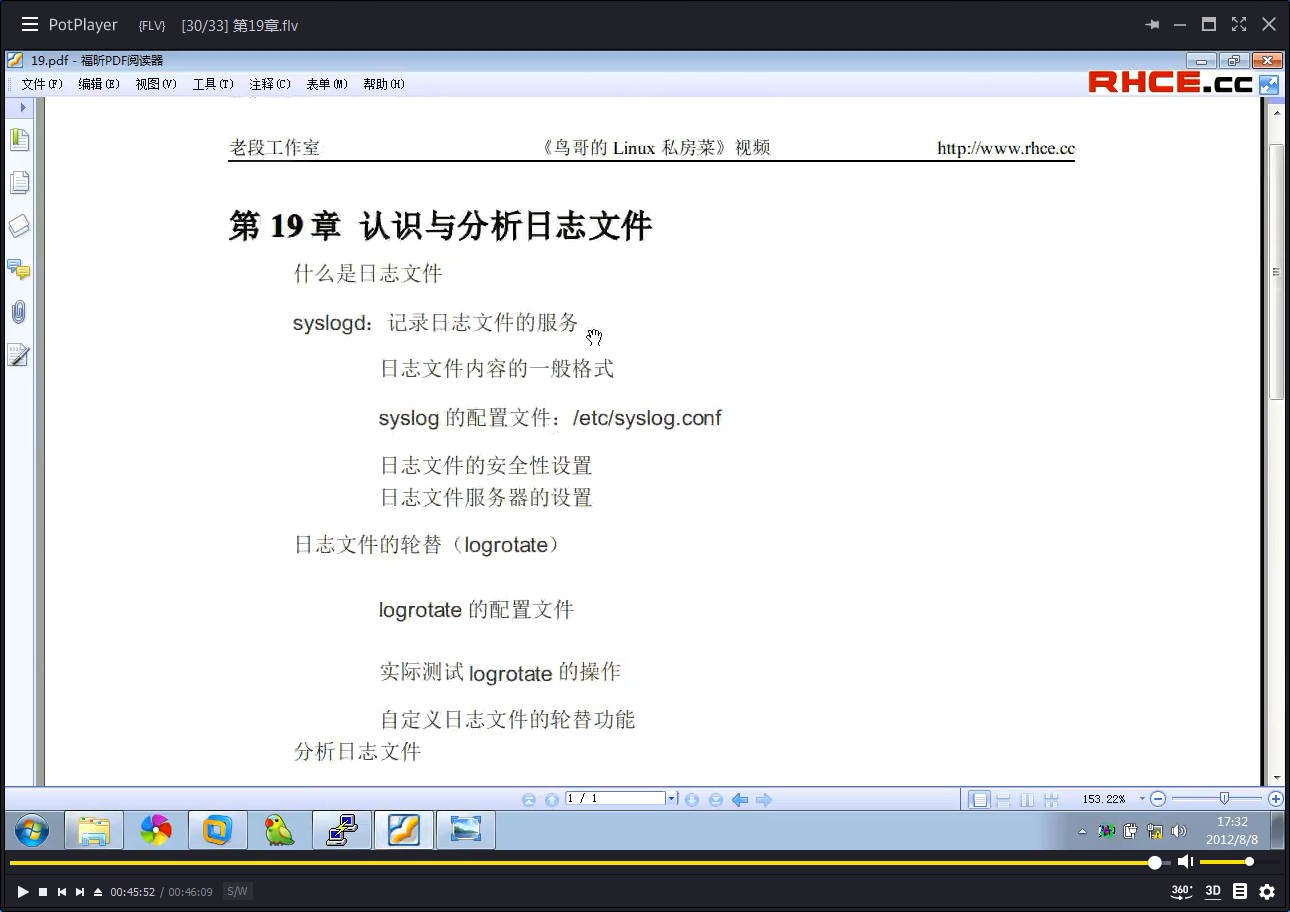
chattr +a message 只能增加不能删除

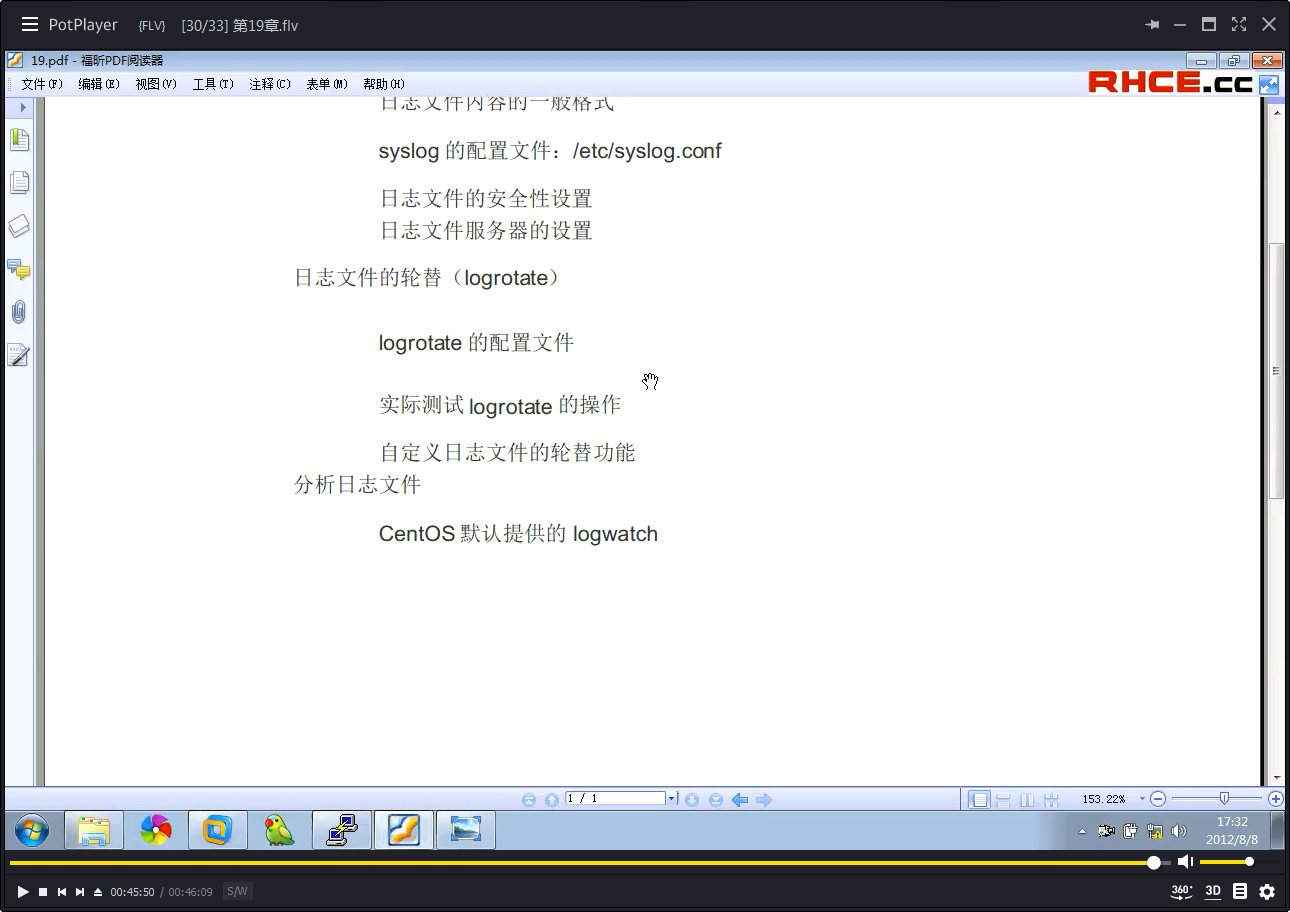
chattr -a message 只能增加不能删除

/var/log/message

logwatch分析日志

/etc/logwatch/conf/logwatch.conf





第20章启动流程、模块管理、loader

MBR 512B

1、446B grub

2、64B分区表

3、2B

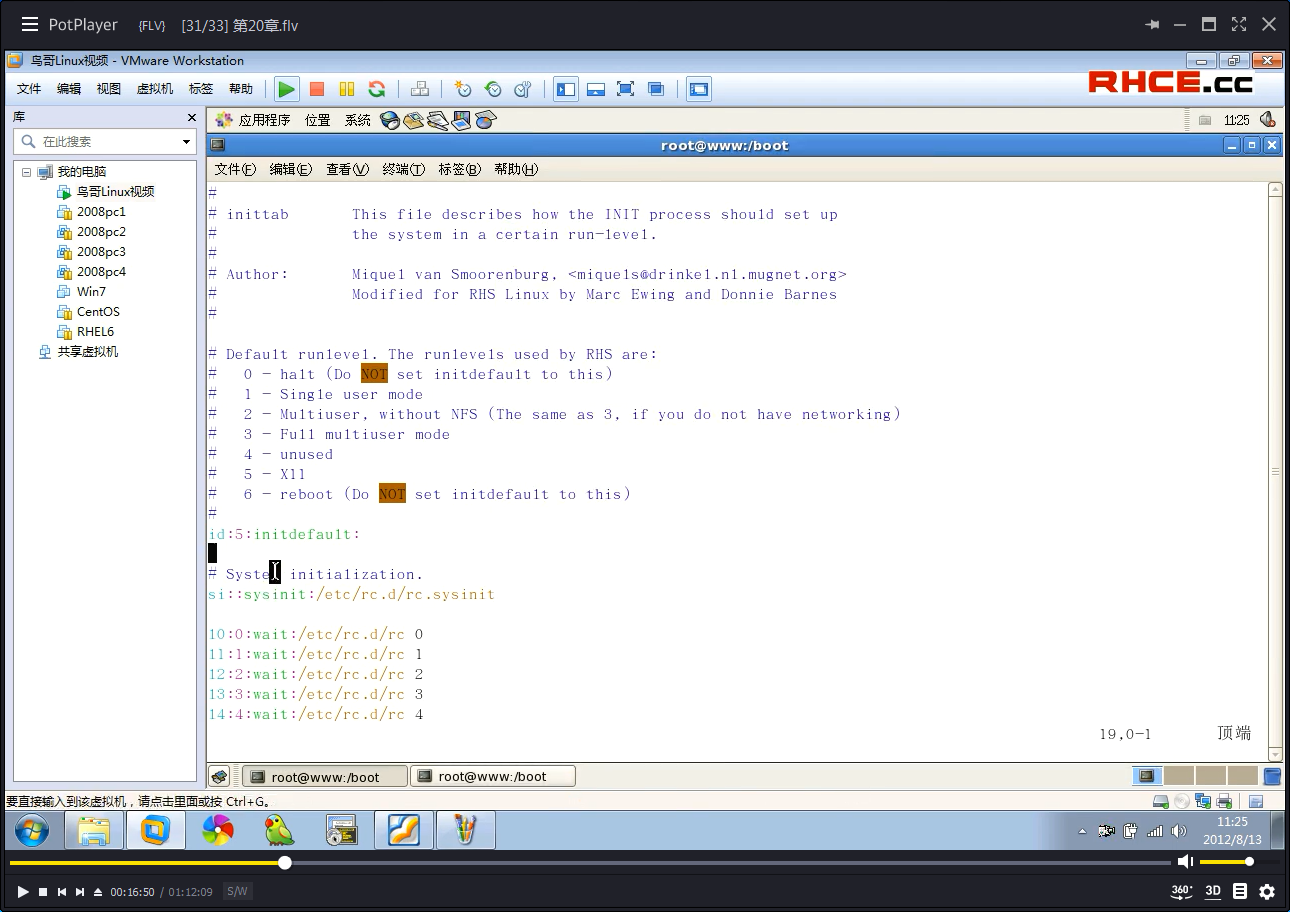
/boot/grub

最先加载stage1大小为512B，只能识别ext3，stage1.5，stage2提供开机界面

/lib/modules系统模块，包括Kernel，

加载内核之后执行

/sbin/init所以init的pid=1，配置文件为



/etc/rc.d/rc.sysinit

/etc/rc.d/rc5.d 以K开头的不运行S开头的才运行

etc/rc.d/rc.local

/etc/inittab

/etc/fstab

/etc/sysconfig 设置服务运行

/etc/sysconfig/i18n

/etc/modprobe.conf

/etc/modprobe.d

runlevel命令查看当前运行级别

init 5切换级别

级别5的最后一个tty是图像化界面，在其它tty输入startx不能启动GUI，因为GUI已经在tty6上运行了

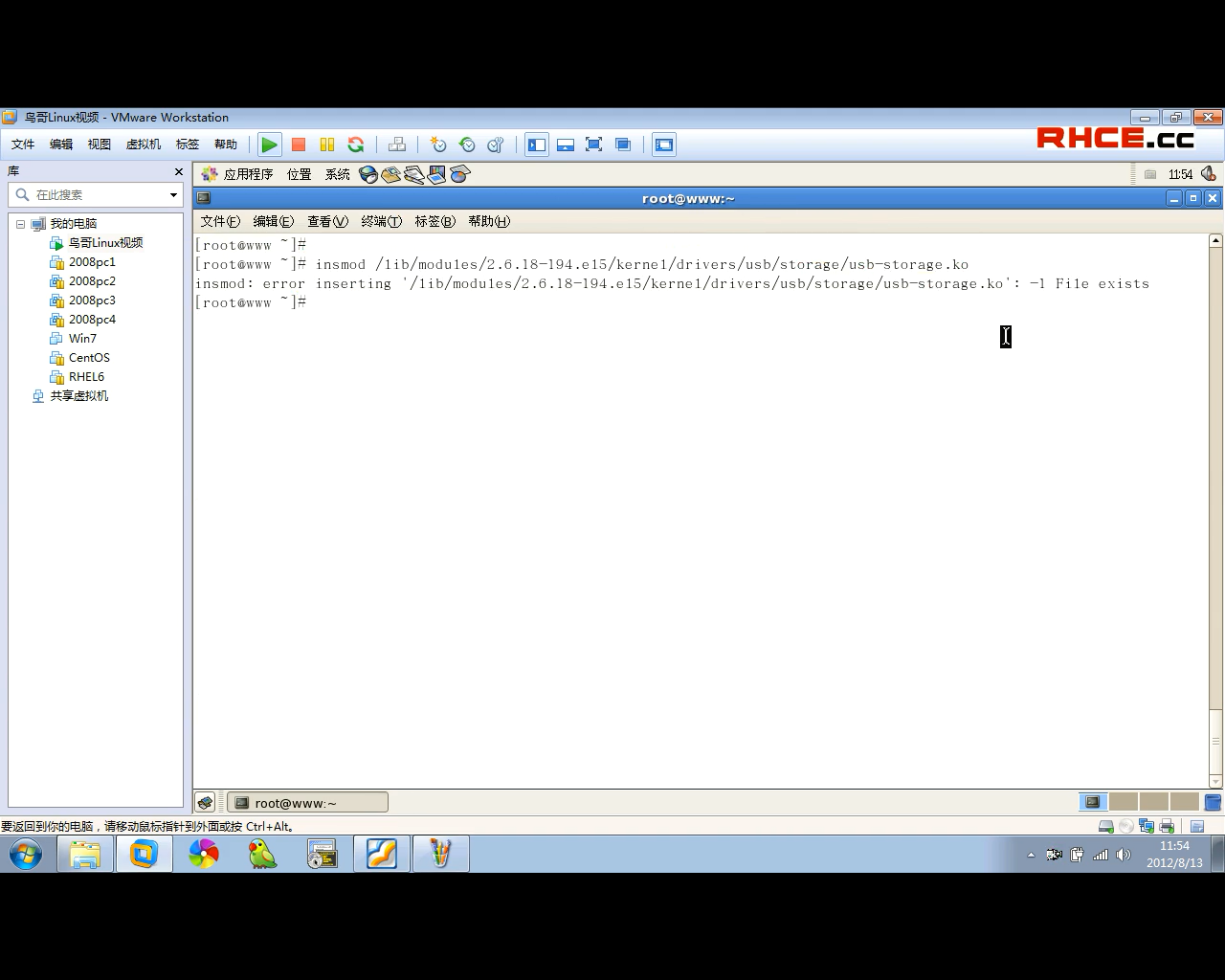
在级别3输入startx可以启动GUI

lsmod查看系统内核已经加载的模块

modprobe modname加载,不需要加ko后缀，也不要路径

modprobe -r modname删除

insmod 需要输入全路径



modinfo pcspkr查看模块信息

/lib/modules

cd kernel

* screen专题

使用screen cmd直接创建一个screen并且进入这个screen，执行完自动退出，无法再次进入

screen -S yourname 新建一个叫yourname的session

screen -ls 列出当前所有的session

screen -r yourname 回到yourname这个session

ctrl a d保持并退出这个screen

exit销毁这个screen，所以创建之后不要使用exit，当不知道是否在screen中时，使用ctrl a d命令

screen的快捷键都是以ctrl a开始的，使用快捷键不能切换到使用-S创建的screen

ctrl a c创建一个screen，只能在一个screen中使用，把ctrl a命令发送给screen，先使用-S创建一个再使用快捷键创建

ctrl a w显示使用快捷方式创建的screen

ctrl a ctrl a 切换到上一个screen

所以要把所有的工作都要放在screen，使用快捷键创建的screen再使用ctrl a d来保存，使用ctrl a w查看

C-a ? 显示所有键绑定信息

C-a w 显示所有窗口列表

C-a C-a 切换到之前显示的窗口

C-a c 创建一个新的运行shell的窗口并切换到该窗口

C-a n 切换到下一个窗口

C-a p 切换到前一个窗口(与C-a n相对)

C-a 0..9 切换到窗口0..9

C-a a 发送 C-a到当前窗口

C-a d 暂时断开screen会话

C-a k 杀掉当前窗口

C-a [ 进入拷贝/回滚模式

C-a A 给窗口起名字

<http://man.linuxde.net/screen>

<https://www.ibm.com/developerworks/cn/linux/l-cn-screen/index.html>

nohup ./program >/dev/null 2>&1 &

于使用nohup时，会默认将输出写入nohup.out文件中

ssh user@server bash < /path/to/local/script.sh

在远程机器上运行一段脚本。这条命令最大的好处就是不用把脚本拷到远程机器上。

ssh user@host cat /path/to/remotefile | diff /path/to/localfile -

比较一个远程文件和一个本地文件

第二十二，二十三章

configure加参数生成makefile

使用make命令根据makefile编译，时间比较长

然后make install

rpm安装、查询、卸载、更新，验证

rpm –ivh xxx.rpm（顺序不能变）

rpm –qa查询安装的包

没法知道rpm的安装位置

rpm –ql linuxqq查看安装的路径

rpm –qc linuxqq安装的配置文件

rpm –qd linuxqq帮助文档

rpm-qi linuxqq安装信息

rpm –qf /etc/passwd查看该文件是由哪个软件生成的

rpm –K 验证签名

rpm –qlp xxx.rpm查看信息

rpm –e linuxqq卸载

rpm –Uvh linuxqq更新软件（会先删除老的软件，再安装新的，不要用它来安装Kernel）

包依赖

yum解决包依赖

/etc/yum/repos.d/aa.repo

[aa]

name=bb

baseurl=ftp://10.10.10.10,或本地file:///mnt

enable=1是否启用

gpgcheck=0是否做签名验证

yum clean all

yum search 包名

yum remove 卸载

yum install 包名

yum list gcc来测试yum

yum whatprovides \*/iostat查看iostat是哪个安装包里有

yum grouplist

yum groupinstall ‘KDE’

yum groupremove ‘KDE’

yum源网站

<https://pkgs.org/>

vsftpd

service vsftpd restart

yum install createrepo

createrepo –v /var/ftp/pub

修改repo即可

* 进程和信号专题

ctrl z放入后台，jobs查看，ps –aux列出系统所有进程详细信息

bg 2从后台放到前台

fg 2

killall-9 %2 其中%代表kill后台的进程，不加的话代表kill pid=2的

ps显示的是当前的shell运行的进程，所以很少

grep –v “aa” 不包含aa的行

ps –ex –o comm，pid，%cpu，%mem，

S表示sleep

R running

Z 是僵尸进程

s stop

http://www.cnblogs.com/f-ck-need-u/p/7058920.html#blog94

Signal Value Comment

─────────────────────────────

SIGHUP 1 终端退出时，此终端内的进程都将被终止

SIGINT 2 中断进程，可被捕捉和忽略，几乎等同于sigterm，所以也会尽可能的释放执行clean-up，释放资源，保存状态等(CTRL+C)

SIGQUIT 3 从键盘发出杀死(终止)进程的信号

SIGKILL 9 强制杀死进程，该信号不可被捕捉和忽略，进程收到该信号后不会执行任何clean-up行为，所以资源不会释放，状态不会保存

SIGTERM 15 杀死(终止)进程，可被捕捉和忽略，几乎等同于sigint信号，会尽可能的释放执行clean-up，释放资源，保存状态等

SIGCHLD 17 当子进程中断或退出时，发送该信号告知父进程自己已完成，父进程收到信号将告知内核清理进程列表。所以该信号可以解除僵尸进

程，也可以让非正常退出的进程工作得以正常的clean-up，释放资源，保存状态等。

SIGSTOP 19 该信号是不可被捕捉和忽略的进程停止信息，收到信号后会进入stopped状态

SIGTSTP 20 该信号是可被忽略的进程停止信号(CTRL+Z)

SIGCONT 18 发送此信号使得stopped进程进入running，该信号主要用于jobs，例如bg & fg 都会发送该信号。

可以直接发送此信号给stopped进程使其运行起来

SIGUSR1 10 用户自定义信号1

SIGUSR2 12 用户自定义信号2

kill -9 pid1 pid1

kill –s TERM pid1 pid2

kill TERM pid1 pid2

值得注意的是nohup命令只是使得程序忽略SIGHUP信号，还需要使用标记&把它放在后台运行。

nohup <command> [argument…] &

fuser可以查看文件或目录所属进程的pid，即由此知道该文件或目录被哪个进程使用。例如，umount的时候提示the device busy可以判断出来哪个进程在使用。而lsof则反过来，它是通过进程来查看进程打开了哪些文件，但要注意的是，一切皆文件，包括普通文件、目录、链接文件、块设备、字符设备、套接字文件、管道文件，所以lsof出来的结果可能会非常多

fuser [-ki] [-signal] file/dir

-k：找出文件或目录的pid，并试图kill掉该pid。发送的信号是SIGKILL

-i：一般和-k一起使用，指的是在kill掉pid之前询问。

-signal：发送信号，如-1 -15，如果不写，默认-9，即kill -9

不加选项：直接显示出文件或目录的pid

在不加选项时，显示结果中文件或目录的pid后会带上一个修饰符：

c:在当前目录下

e:可被执行的

f:是一个被开启的文件或目录

F:被打开且正在写入的文件或目录

r:代表root directory

例如：

[root@xuexi ~]# fuser /usr/sbin/crond

/usr/sbin/crond: 1425e

文件描述符

([彻底搞懂shell的高级I/O重定向](https://www.cnblogs.com/f-ck-need-u/p/8727401.html))

http://www.cnblogs.com/f-ck-need-u/p/8727401.html#comment\_form\_container

>file 2>&1里分两个过程：先打开file，再将fd=1重定向到file文件上，这样file文件就成了标准输出的输出目标；之后再将fd=2绑定(注意，是绑定不是重定向)到fd=1上，而fd=1此时已经重定向到file文件上，因此fd=2也重定向到file上。所以，最终的结果是标准输出重定向到file上，标准错误也重定向到file上。

2>&1 >file, 这里也分两个过程：先将fd=2绑定到fd=1上，而此时fd=1重定向的文件是默认的/dev/stdout，所以fd=2也重定向到/dev/stdout；之后再将fd=1重定向到file文件上(注意，不是绑定是重定向)。也就是说，这里的标准错误和标准输出仍然是分开输出的，只不过是使用/dev/stdout替代了/dev/stderr，使用file替代了/dev/stdout。所以，最终的结果是标准错误输出到/dev/stdout，即屏幕上，而标准输出将输出到file文件中。

echo 1234567890 > File # (1).写字符串到"File".

exec 3<> File # (2).打开"File"并且给它分配fd 3.

read -n 4 <&3 # (3).只读4 个字符.

echo -n . >&3 # (4).写一个小数点.

exec 3>&- # (5).关闭fd 3.

cat File # (6).1234.67890

[n]<&word ：将文件描述符n绑定到word 代表的文件或描述符。可以理解为文件描述符n重用word代表的文件或描述符。n不指定则默认为0(标准输入就是0)，表示标准输入也将输入到word所代表的文件或描述符中。

[n]>&word ：将文件描述符n绑定到word 代表的文件或描述符。可以理解为文件描述符n重用word代表的文件或描述符。n不指定则默认为1(标准输出就是1)，表示标准输出也将输出到word所代表的文件或描述符中。

1.5 关闭文件描述符

[n]>&-

[n]<&-

exec 6>&1 # (1)

exec > /tmp/file.txt # (2)

echo "---------------" # (3)

exec 1>&6 6>&- # (4)

echo "===============" # (5)

exec > /tmp/file.txt

echo "---------------"

exec >/dev/pts/2

echo "==============="