第一章：系统安装

分区选择

**/boot 500MB 主分区 Ext4:**

这个挂载点包含了操作系统的内核和在启动系统过程中所要用到的文件,单独分区好处在于若是主分区磁盘损坏，不会导致不能boot，这个空间稍微大一些，linux升级内核时，不会自动删除旧的内核，所以空间会越来越大。

内核镜像文件：

linux-image-: 内核镜像

linux-image-extra-: 额外的内核模块

linux-headers-: 内核头文件

uname –a 显示当前内核信息

dpkg --list | grep linux-image 显示所有已存在内核版本

apt-get remove --purge linux-image-4.3.3 删除不要用的内核，释放boot空间

**下面的都是扩展分区下的逻辑分区**

**/ 15G Ext4：**

相当于window的系统盘，存放系统文件，单独分区的好处在于重装系统时，只需要对此分区格式化，默认安装的软件都在这下面。

**/home 剩余所有空间(留下swap空间) Ext4：**

存放个人所有数据,相当于window下的我的文档，重装系统时不需要格式化此分区

**Swap 物理内存的2倍（内存小于2G）物理内存+2G（内存大于2G） :**

相当于window的虚拟内存，当系统内存不够用时，取一部分硬盘资源作为内存使用

ps: linux /目录是固定的，上述分区只是决定该目录是否单独放在一块磁盘位置，而不是决定/下有几个目录。

系统启动流程：进入主板的BIOS程序，程序操纵指针从硬盘读取主引导记录MBR，MBR中包含引导程序(Grub则由于大小原因在MBR中只保存位置信息)和分区信息（64kb，每一个分区信息16kb，所以最多4个分区(主+扩展)，扩展分区可以划分更多的逻辑分区）。

第二章：用户管理

用户分类：一个用户可以属于多个组

普通用户：只能在其home目录工作，安装系统时的用户xyh也是其中一个，但其默认属于admin组，具有修改root密码的权限。

根用户：root，刚安装需要通过xyh用户sudo passwd初始化root密码

系统用户：程序运行时必须要有的用户，系统用户mysql运行mysqld进程

登录方式：tty:设备终端 pts:虚拟终端

切换用户：

su –（有没有这个参数重要，不加则还是原来的shell环境（切换的用户由于无法读取/etc/profile生成环境变量，所以在profile中配置的环境变量无法使用），加了则shell环境变量也切换到该用户（相当于登录了该用户）） \*\*\* 即可切换用户，但必须输入该用户密码

sudo 是以root身份执行后面的命令，只有配置了该权限普通用户才能执行(在/etc/sudoers配置，或者加入sudo这个组中)

管理用户：/etc/passwd

用户名:口令:用户标识号:组标识号（主组）:注释性描述:主目录:登录Shell

1在/home目录下增加一个用户： adduser \*\*（会要求建立密码，passwd命令即可修改）

2删除一个用户：userdel –r \*\*\* 不加入-r则/home下目录文件不会删除

3加入某一个group: usermod –aG g\*\*（可以多个）u\*\* 加入某一个组 (如sudo命令需要加入这个sudo group才可以)

管理组：/etc/group

组名:口令:组标识号:组内用户列表（只显示附加组用户）

1将一个用户从一个group中删除： gpasswd –d u\*\*\* g\*\*\*

2将一个用户加入group中：gpasswd –a u\*\* g\*\*\*（附加组）

3命令groupadd del 都和用户操作一样

4命令groups u\*\***(若是不加u\*\*显示内容不同，则需要先newgrp以主组方式加入该组，然后再恢复之前主组，才算拥有该组权限)** 可以查看该用户所属的所有组

5命令newgrp g\*\* 使得当前用户以g\*\*为主组方式登录（id命令之后gid会显示g\*\*组号）

第三章 文件管理

Linux根目录：

/bin:用户指令 、/sbin:root用户指令、/etc：系统服务的配置文件、/lib:系统函数库，作用类似windows里的.dll文件、/dev磁盘等设备（系统连接光驱或者USB盘时，都在这下面出现该设备）、/mnt该目录用来挂载光驱或者USB盘、/home用户主目录，放置用户数据、/opt 安装可选程序（该程序所有东西都在该目录下，不影响系统）、/usr：放置应用程序的总目录（可执行命令都在/bin or /sbin目录下,若是通过apt-get安装，系统自动把软件中不同文件放到不同位置lib/bin…,若下载tar.gz包，则自己读说明文档）

/usr/local/ 本地自己安装的可执行文件

/usr/ 系统商开发可运行程序

\*\*/bin命令所在的目录都在$PATH环境变量中，所以在终端可以直接输入

文件操作：

head/tail –n \*\* : 只显示多少行 tail –f \*\* 可以动态的加载文件最后的变化内容

more 可以分页显示

sed –n ‘1,10p’ 显示第1到10行

mkdir –p : 创建多层目录

rmdir：不能删除空目录 rm –r/-rf：可以强制删除目录，包括下面的文件

cp: 复制文件，若是文件夹则加–r参数

wc –l filename 统计文本行数 –c字节数 –L最长行长度 –w打印单词数，通常与ls结合使用 ls filename|wc –l 统计某个目录下有多少文件

xargs 很多命令不支持|传递参数，则需要这个命令传递，比如 find ./ -name \*|xargs rm 这样才能删除,当然find有更好的选择-exec，参数-t显示命令 –i将前面输出每一项赋值给{}，然后command中即可使用{}，find –exec 中{}同理

rm –r（删除目录和下面的文件，但要一一询问） -rf（直接删除） 没参数则后跟filename

mv 即可移动文件，也可重命名文件

文件权限：

user/group/others 代表不同拥有者 r/w/x(4/2/1) 代表不同权限 -R 表示递归设置目录下文件和目录

chmod u/g/o（代表修改哪个） –x/+r filename 同一时刻只能设定一个拥有者权限

chmod 777 filename 同时设定

chown –R（递归处理）owner filename

chgrp –R（递归处理）grouper filename

文件查找：

find PATH –name/perm/type …. \*\*\* 通过不同参数+通配符组合 设定查找的条件,无正则

-exec参数 对查找到的文件进行额外操作，格式 –exec [command] {} -args \; {}是替代符，代表前面的输出

which \*\*\* 查找shell命令所在位置

文件压缩和打包：

gzip/gunzip: 压缩和解压单个文件，不能是文件夹

tar –z（整合或者解压时是否使用gzip压缩或者gunzip解压）–c/x (压缩、解压) -v（压缩过程显示文件） -f（指定压缩、解压名字）

若是压缩-f后面紧跟压缩后名字.tgz + Path（压缩目录）.

若是指定解压路径 –f后面紧跟压缩文件 + –C Path

第四章 文件系统

磁盘管理工具：默认都是kb单位计算，可以调整

df用于检查文件系统磁盘占用情况 –T 显示文件系统类型 –h 大小单位最优选择，du检查文件/目录的空间大小（-s 显示总计 –h单位最优），而fdisk用于检查磁盘使用情况，free查看内存使用情况，top查看cpu使用情况

磁盘分区，挂载：

新添加的磁盘需要先分区，再格式化文件系统（扩展分区是不能格式化的，必须要建立逻辑分区才能使用），然后将该分区挂载到某个目录下才能使用。

fdisk –l 查看磁盘情况

fdisk /dev/sdb 输入n建立新分区（扩展分区的话还需要n建立逻辑分区），d删除一个分区，最后w写入，通过命令partprobe让此分区立即生效

mkfs –t ext4 /dev/sdb1 格式化磁盘sdb1的文件系统为 ext4类型

mount /dev/sdb1 directory 挂载到目录上（但这是临时，重启机器就会失效，永久挂载需要修改/etc/fstab 命令： echo “/dev/sdb5 /newDisk defaults 0 0” >> /etc/fstab）

umount /dev/sdb1 取消挂载，相当于window下的弹出

物理卷和逻辑卷：都是一开始就需建立

普通分区建立的都是物理卷，一旦建立无法改变大小，逻辑卷是建立在物理卷之上，通过修改fdisk磁盘 修改type 83变成8e，交给LVM管理，建立几个物理卷合起来的卷组，然后可以在卷组上建立逻辑卷，逻辑卷好处是随意扩展磁盘空间大小，加大容易，减少可能会损坏数据。

第五章 字符处理

**|**：管道上一个命令的输出(通常是文本)作为下一个命令的输入

grep –icnv ‘\*’ Filename：基于行的文本搜索， -i不分区大小写 –c输出统计行数 –n输出行号 –v反向匹配，与正则表达式结合功能强大，且都需要’’单引号

**Command分割原则：**

IFS（命令分词解读）: space控制

CR（命令执行）: enter控制

**引号问题：所有command都需要注意**

无引号：元字符起作用 = $ > {} () space enter…

双引号：只有少量元字符还起作用 $ “” \

单引号：都不起作用，如enter放在 ’ 后面则不会具有元字符作用，必须要‘关闭

\：只能让紧挨着的一个元字符不起作用

反引号：位于Esx下面， `command`将该命令执行结果暂时保存，如echo `date`即可输出日期

第六章 网络管理

网络配置：

iwconfig: 查看无线网卡的信息（ra0 wlan0标识）

Ifconfig ens33（以太网，新系统ens33表示，原来的系统都是eth0，需要具体看）查看设备信息，或者-a列出所有网卡信息

Ifconfig ens33 192.168.80.132/24 动态修改ip，但下次启动系统或者网卡关闭就丢失了

将ip配置信息写入配置文件，这样才能永久修改ip，修改/etc/network/interfaces,加入如下信息：

auto ens33 //设置自动启动ens33接口

iface ens33 inet static //配置静态IP

address 192.168.100.13 //IP地址

netmask 255.255.255.128 //子网掩码

gateway 192.168.100.1 //网关

配置DNS:

vi /etc/resolvconf/resolv.conf.d/base

nameserver 192.168.1.1(网关)

resolvconf –u 让新的host生效，此时在/etc/resolv.conf中出现最新的DNS

sudo service networking restart

sudo /etc/init.d/networking restart (有些机器不支持上面的操作)

Ifdown/up ens33网卡关闭和安装（网卡信息会根据配置文件决定），ubuntu桌面版上面命令无法更新网卡信息，只能通过此命令或者reboot，ubuntu服务器版只能reboot

网络故障：

网关ip : 应该是本网络内具有路由功能的机器ip（比如局域网内就是路由器ip地址）

host命令可以用来查看域名对应的IP地址

排除步骤：

ping 127.0.0.1 确认本机TCP/IP协议是否正常

ping 本身的ip 确认网卡是否有物理或者驱动故障

ping 同一个网段的其他主机 确认交换机是否工作正常

ping 网关ip 确认数据包能够发送到网关

ping 公网ip 确认路由设置是否正确

ping 公网域名 确定DNS解析正确

第七章 进程管理

进程观察：

ps –e 所有进程 –aux所有进程详细信息 –a本终端进程 –u 本用户进程

top 动态显示，默认cpu占比排序

kill -9（加则表示强制退出，不加则自行退出） PID 根据进程号杀死对应进程

lsof filepath 打开正运行在该文件上的所有进程 -u username 列出被用户打开的进程及文件信息 –c command 列出包含command字符串所有进程及文件信息 –i : 25列出占用端口25的进程

远程终端窗口：操作的时候只有一个终端

screen -S yourname -> 新建一个叫yourname的session

screen -ls -> 列出当前所有的session（前面的是该process的pid）

screen -r yourname -> 回到yourname这个session（必须是detached状态，若不是要先-d）

screen -d yourname -> 远程detach某个session（本screen下可以不加name）

exit 永久退出，不可恢复（也可以kill PID）

kill -9 PID 即可杀死该screen进程，里面的所有程序也都杀死了

Ctrl+a c: 创建一个窗口（一个session可以创建多个window，不然一个window若是在等待脚本执行，那么这个session将没有作用）

Ctrl+a w: 显示当前窗口号

Ctrl+a n: 切换窗口

Ps: 或者在linux命令后+&，表示守护进程，在后台运行，这样就不会占据终端，进程的输出依然会显示在当前终端，但进程接收无法输入。Session关闭时，守护进程不会关闭，前端进程则会关闭

系统服务：

/etc/init.d/就相当于service

service \*\* start/stop/restart/status 操作服务

第八章 软件安装

安装方法：

apt：建立在dpkg基础上（本质通过dpkg安装，即系统自动下载了.deb包安装）

apt–get install packagename=version: 自动从软件仓库下载软件，但是得知道软件的确切名

remove --purge（配置文件一同卸载，依赖包不会卸载） packagename

autoremove 删除为了满足其他软件包的依赖而安装的，但现在不再需要的软件包，和上面的remove –purge一起用使得软件完全卸载。

upgrade 更新所有软件（也可以单独更新某软件）

update 更新仓库镜像列表（在使用dpkg方式安装好之后，系统默认不会记录该软件信息，所以还需要update）

apt-get clean 删除安装包

dpkg: 系统绕过apt本地软件仓库，安装用户已下载好的.deb文件，系统自动管理软件各个包位置

.deb格式（ubuntu系统）：

dpkg –i xxx.deb -r 移除 –P清除（配置文件一同卸载）这种方式不会自动解决依赖关系，遇到这种问题时，apt-get –f install命令来修正依赖关系

dpkg –l | grep linux-image 能够列出系统已安装的包含linux-image关键字的软件，常用来查找已安装的包

安装好之后apt-get update 更新软件仓库，让系统知道安装了这个软件

.tar.gz: 所有配置文件都在一个目录下（这种方式软件仓库无法知道是否安装了此软件）

(最常用的方式，官网下载tar.gz压缩包，然后整个解压到/opt目录下或者/usr/lib,自己用则解压到/home/xyh/usr/下也可以，然后根据文档配置其参数变量即可，但这种方式无法像apt-get安装在PATH中直接使用，需要手动配置PATH才行，否则只能在当前目录下运行其可执行文件 ./或者全路径+命令)

下载命令：桌面版直接上浏览器下载，但服务器版需要命令行操作

wget 自动下载，即用户退出系统后仍会执行，目录同步机制

-O filename http://\*\*/\*\*/xyh 指定文件名，否则默认以网址的最后一个/后的名字xyh命名

-P /download http\* 指定下载目录

-nd 不建立目录，默认同步机制

-c http\* 断点续传，之前下载文件由于网络中断了，要继续下载，否则会重新下载

-b http\* 后台下载，不会占用界面，和$差不多，会在当前目录产生wget-log日志文件，可以使用tail –f wget-log动态查看

环境变量：env命令显示所有环境变量及其值

如果不配置环境变量，在非软件安装的目录下运行命令，将会报告找不到文件。

1、修改 /etc/profile 文件，所有用户的shell都有权使用这个环境变量。

在/etc/profile的最下面添加： export PATH=$PATH:/NEW\_PATH （注意：$PATH是把原来的也加入，且先后顺序有关系，若$PATH放在最后则先会寻找前面的命令，当前者都不存在时才会在原来的$PATH（/usr/bin等都在这$PATH中）中找，分隔符是冒号：）

2、修改~/.bashrc文件，可以把使用这些环境变量的权限控制到用户级别，只需要修改其个人用户主目录下的 .bashrc文件就可以了，当和系统变量冲突时将会采用用户变量。

在下面添加： export PATH=$PATH:/NEW\_PATH

上述文件中export的变量，在shell中echo $JAVA\_HOME 即可显示对应的值

常用软件：

JAVA安装：

1. 下载jdk-8u91-linux-x64.tar.gz, 网址如下

http://download.oracle.com/otn/java/jdk/8u91-b14/jdk-8u91-linux-x64.tar.gz

1. sudo tar –zxvf ~/下载/jdk-8u91-linux-x64.tar.gz -C /usr/lib/jvm/
2. sudo vi /etc/profile 在末尾加上:

#jdk environment

export JAVA\_HOME=/usr/lib/jvm/jdk1.8.0\_91

export CLASSPATH=$CLASSPATH:.:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib

export PATH=$PATH:$JAVA\_HOME/bin:$JAVA\_HOME/jre/bin

1. source /etc/profile 立即更新配置文件（有时候需要重启才能生效）
2. 因为java命令加入了PATH路径，所以大家都能使用

Python安装：

Linux系统自带python2/3，或者直接使用apt-get install/upgrade 安装/更新

Tomcat安装：

1、下载apache-tomcat-9.0.0.M9.tar.gz，解压到/opt目录

2、sudo vi /etc/profile 在末尾加上:

export CATALINA\_HOME=/opt/apache-tomcat-9.0.0.M9

4、source /etc/profile 立即更新配置文件

5、sudo vi catalina.sh 在正式代码开始之前（100行左右）

export JAVA\_HOME=/usr/lib/jvm/jdk1.8.0\_91

6、root通过$CATALINA\_HOME/bin/startup.sh shutdown.sh即可操作tomcat

7、更改权限（默认只能root），创建一个tomcat组，让需要的用户加入这个组（gpasswd），chmod修改/opt/ apache-tomcat-9.0.0.M9目录下所有文件（-R参数）的权限为770，并且chgrp修改其group为tomcat

**MySQL安装：**通过apt安装mysql-server

mysql命令教程：

**1 连接mysql：**

本机 mysql –u root –p

远程 mysql –h ip –u root –p p\*\*

2 修改密码：

mysqladmin -u root -p password

3 管理用户：只能root用户执行，信息都在mysql.user表中

增加并授权：

grant 权限（select/update/all）on database（\*代表全部）.table（\*代表所有） to ‘username’@’host’（’%’代表所有/’localhost’/具体’ip’）identified by ‘password’

flush privileges; 刷新权限

删除：delete from mysql.user where user=’username’ and host=’localhsot’；

**4 执行脚本文件：**source /path/\*\*.sql；

命令版本切换工具：update-alternative，安装的命令需要--install之后才能在这里操作，并不是可执行的命令就会出现在—display中，这个命令可以代替java等命令配置环境变量的过程，即采用这种方式配置java，则java不再需要额外修改profile .bashrc等文件配置环境变量了。适用于系统中存在多个java版本，且环境变量没有主动配置，或者配置方式是$PATH放在了%JAVA\_HOME%前面，即采用的是原$PATH中的命令）

--display显示一个命令链接的所有可选命令，包括链接的模式(自动还是手动)、链接priority值、所有可用的链接命令等等。

--install选项的功能就是增加一组新的系统命令链接符。

update-alternatives --install <link> <name> <path> <priority>其中link为系统中功能相同软件的公共链接目录，比如/usr/bin/java(需绝对目录); name为命令链接符名称,如java；path为你所要使用新命令的所在目录；priority为优先级，当命令链接已存在时，需高于当前值，因为当alternative为自动模式时,系统默认启用priority高的链接。

--config选项用来显示和修改实际指向的候选命令，为在现有的命令链接选择一个作为系统默认（手动）。

--remove选项的功能是删除一个命令的link值

update-alternatives --remove name path （和install一样）

第九章 vi/vim使用

桌面版的vi是不完整（操作命令不同），需要安装vim才能使之恢复

服务器版本vi没有问题

模式切换：命令模式（：+command），插入模式（i，Esc退出）

基本操作：都在命令模式下

u 恢复上一次操作

num 调转到指定行

set nu 列出行号

/关键字 查找关键字，n下一个

正则表达式：

第十章 shell编程

概念：shell是解释性脚本语言，用户输入shell命令，解释后系统执行相应操作。Bash shell则是流行的shell版本。

两种工作模式：互动模式（用户输入一条命令，系统就执行）/脚本模式（用户将多条命令写入脚本文件.sh，系统统一执行）

脚本运行：

bash helloworld.sh

./helloworld.sh（这种方式脚本开头必须加入 #!bin/bash）

若要成为系统默认命令，则加入$PATH变量包含的任意目录即可，如/bin /usr/bin。

**变量**： variable=”value” ${variable}

**特殊变量**：$0（文件名） $num（脚本后面第num个变量值） $#（参数个数） $$（PID） $\* $@（所有参数输出）$HOME（家目录）

**运算符：**算数运算（3 +/\\*/%/== 3 乘法需要转义） 关系运算（2 –gt/lt/ge/le/eq/ne 3） 布尔运算（! true true –o/a false） 字符串运算符（=相等 ！=不等 –z长度为0则true 直接本身则是否为空） 文件测试运算（-d/r/w/x/s大小是否为空/e是否存在 $file//此变量指向文件路径）

条件语句：[ $a != $b ]必须放在括号中，且有空格

**字符串：**${#string} 获取长度 ${string:1:4}截取子字符串

**数组：**array=(num1 num2 num3) or array[0]=num1 array[2]=num3 声明的两种方式 ${#array[\*]} 获取长度

**If else语句：**两种用法

If [ ]

then

else

fi

if[ ]

then

elif[ ]

then

…

fi

**case esac语句：**switch case

case $num in

模式1）command..

;;

模式2）command..

;;

\*）command..相当于defaults

;;

esac

**for循环：**

for file in $HOME/test\* //以test开头的文件

do

command…

done

**while语句：**

while read string

do

command

done

**shell函数：**

function fun() {

${num} 获取第num个参数

return

}

fun param1 param2 //调用，并带参数

**IO重定向：**

command > file输出到文件且覆盖 >>追加文件末尾

command < file从文件读取

**引入外部脚本：** . file.sh

Vmware使用

宿主机通信的3种方式：

Bridged模式：通过VMnet0虚拟交换机实现，虚拟系统就像主机所在的局域网中另一台独立的机器，但需要多余的IP地址（内网中方便，但是公网就不可能了），手工为虚拟系统配置IP地址，子网掩码，DNS。

NAT模式： 通过VMnet8虚拟网络实现，虚拟系统就像主机建立的一个局域网内的机器，虚拟机可以访问外网，但外网不能访问虚拟机（宿主机可以和虚拟机通信，其他主机不能），ip地址直接由VMnet8提供的DHCP服务动态获取。

Host-Only模式：通过VMnet1虚拟网卡实现，虚拟机处于主机建立的一个全封闭的网络中，虚拟机互相可以通信，和主机也行。

开启复制粘贴：进入虚拟系统，然后install Vmware tools，会以一个cdrom形式载入虚拟系统，然后更具系统特点安装此软件。重启生效。

开启共享文件夹：安装好vmware tools，在虚拟机settings中添加共享文件夹，重启后，window下直接在我的电脑出现贡献文件夹，ubuntu下在/mnt/hgfs下。

克隆备份：分为链接克隆（源主机坏了则链接也不能用），完全克隆（完全脱离源主机）

快照模式：建立快照，然后就和git一样可以回到原来的位置

.vmdk文件作用：和iso镜像有点类似，会自动保留该系统在.vmdk文件中，下次新建系统时，不要选择先建立磁盘，磁盘选择直接导入.vmdk文件即可加载系统上所有数据。