网络链接的三种模式

网段 相同数字序列开头的可以相互通信

张三的母本安装了虚拟机

桥接模式：直接给他分配一个IP地址 容易分配满

NAT模式：主机和虚拟机通信用一个网段ip然后外部通信 主机用另外一个网段

虚拟机想要与外部通信要通过主机

主机模式：直接把虚拟机隔离 当一个独立的系统 爱怎么搞怎么搞

VMTOOLS

可以在windows系统下与虚拟机共享文件夹

cd 进入目录

tar -zxvf解压

./执行当前目录下某个程序

/opt目录下放解压文件

/mnt/hgfs文件夹里有与主机共享的文件

实际开发中文件的上传与下载需要用远程方式登录的

###第四章linux的目录结构 linux中 一切皆文件 没有界面操作记住树状目录图

root root的用户的目录

home 家目录 就是每个用户的合计

bin 目录 常用指令在这个文件下

etc 文件配置目录

/bin 常用 这个目录里面放常用的命令和指令

/sbin 系统管理员 用到的系统管理指令

/home 存放普通用户的主目录

/root 系统管理员的目录

/lib 系统开机所需要的最基本的动态链接共享库，其作用类似于dll文件 所有程序都要这些库

/lost+found 系统非法关机后存放一些日志文件

/etc系统管理需要的配置文件 比如mysql需要的配置文件

/usr 用户安装的应用程序的默认目录

/boot linux启动时候需要的核心文件

#下面的不动

/proc 这个目录是系统内存的映射 通过访问这个目录获取系统信息

/srv service缩写该目录存放一些服务启动之后需要提取的数据

/sys 内核相关

#上面的不能动

/tmp临时文件

/dev device设备的缩写 硬件用文件形式存储

/media linux系统识别到的一些设备

/mnt 让用户临时挂载外部的文件系统

/opt安装包放置的目录 先拷贝到这里

/usr/local 软件安装好后放的地方

/var 日志目录

/selinux 360目录

###linux的实操篇 远程登陆到linux服务器

xshell 远程操控linux

文件upload and update xftp

尚未搞定

####第六章vi和vim

类似于记事本 内置的 vim有程序编辑的能力

vim有三种模式 正常模式 用于移动光标和删除字符

插入模式 编辑模式/输入i I O o a A r R 进入编辑模式

esc回到正常模式

：进入命令行模式

vim hello.java编写一个名称为hello的java程序

命令行一： wq 写入并且退出 q q！ 不保存退出

yy拷贝当前行 5yy拷贝当前向下的五行 yank的缩写

p是粘贴 粘贴板上的东西

dd 删除当前行 5dd 同上

/关键词 查找关键词 n查下一个该关键词 #命令行下使用

set nu set nonu 设置行号 #命令行下使用

编辑/etc/profile文件 G最末行 gg首行

hello 的撤销 u

/etc/profile 并将光标移动到二十行 20 shift+g

正常模式下使用上面的指令 光标要指到你想操作的行

#第七章 开机 重启 用户的登录与注销

shutdown -h now 立刻进行关机

shutdown - 1 一分钟后关机

shutdown -r now 重启计算机

halt 关机

reboot 重启计算机

sync 把内存的数据同步到磁盘

不能做哪件事前 都必须要先执行 sync #小心使得万年船

用户登录和注销 登录后少用root账号登陆

su - 用户名命令来切换成系统管理员 切换用户相当于在原用户栈里面加入新的用户

所以用logout 会回退到上一个用户 当用户栈里面没有用户的时候退出系统

logout 注销用户 把用户退出 图形界面使用logout 图形界面无效

#第八章用户管理

useradd 用户名 加入用户 默认该用户的目录在/home/用户名下 默认没有密码

passwd 用户名 给用户名修改密码 如果不加用户名 是给当前登录状态下的用户改密码 输入完后就可以修改密码了

pwd显示当前用户所在的目录

删除用户

userdel 用户名 这种删除会保留家目录

userdel -r 用户名 目录也删除

查询用户信息

id 用户名 查询用户的uid

whoami 显示当前谁在登录

用户组 类似于角色 系统可以对有共性/权限的角色都进行管理

groupadd 组名

新增组

groupdel 组名 删除组

useradd -g 用户组 用户名 新建用户放到组里

修改用户的组 帮用户放到某个组里面

usermod -g 用户组 用户名

#/etc/password 文件 存储了用户的各种信息 用户名 口令 用户标识号组号

用户名/密码/自己的id/组id/注释说明/家目录/解释器

etc/shadow 可以看到密码加密后的样子

etc/group 组名/密码/组id

##第九章 实用指令##

运行级别

0-关机

1-单用户 找回丢失密码

2-多用户状态没有网络服务

3-多用户状态有网络服务

4-系统未使用保留给用户

5-图形界面

6-系统重启

init [0-60]切换不同的指令级别

systemctl get-default

systemctl set-default multi-user .target 会将系统设置为默认级别三有网络服务的多用户

经典面试题目

找回root密码

重启到选择内核的时候动一下光标，然后选第一个输入E

找到linux16 在行的最后输入init=/bin/sh 输入ctrl+x 进入单用户模式

在光标闪烁的地方输入 mount -o remount,rw/ 然后按回车

输入passwd回车 输入密码 再次确认密码

帮助指令

man 命令/配置文件 获得帮助信息 查询手册

help

ls -a 列出所有文件包括隐藏文件 以.开头的文件是隐藏文件

选项可以组合使用 ls -al all line

help 命令 获得shell内置命令的帮助信息 但是用英文的 用百度更方便

文件目录类#

pwd 显示当前工作目录的绝对路径 没有盘这个概念

ls 选项 目录或是文件

cd [参数]

cd ~回到自己的家目录 root回到/root

cd .. 返回上一级目录

mkdir make directory创建目录

-p创建多级目录 用的相对路径 绝对路径都可以

rmdir 指令删除空目录

rmdir [选项] 要删除的空目录

rm -rf 要删除的目录 可以非空

touch 文件名称 创建一个空文件 再用vim编辑

拷贝指令

cp [选项] source要拷贝的哪个文件 dest拷贝到哪里去bbb/拷贝到当前路径下bbb目录里面去

cp -r /home/bbb/ /opt/ 用-r选项来复制整个目录

连续使用 会提示是否要覆盖 每个文件都要覆盖确认很麻烦 用\cp来进行强制覆盖

rm 指令一处文件或目录

rm [选项] 要删除的文件或者目录

-r删文件夹 -f强制删除不提示

mv指令 移动文件与目录或者重命名 mv 旧名 新名

mv 源文件 移动到的文件路径 剪切

管道命令

cat 查看文件内容

cat [选项] 要查看的文件

-n 显示行号

|more 管道命令 为了浏览方便 可以让查看方便能用 空格翻页

less 要查看的文件 /字串 查找子串 ?字串 查字串 q离开字串

page up /down键盘上 键盘下

echo 输出内容到控制台

echo [选项] [输出内容] $HOSTNAME 主机名字

head用于显示文件开头的部分内容 head 文件 head -n number任意行数代码 默认诗行

tail 与head相反 tail -f实时跟踪文件的变化

>>追加（尾加） >输出重定向（覆盖里面的内容） echo "hello">mydate.txt 把本该输出到终端的内容输出到文件里面去

ln指令 软连接也称为符号链接 文件指针！快捷方式

ln -s [源文件/目录] [软连接名称]

ln -s /root/ /home/myroot 有没有后面/是区分他是文件目录还是文件的

cd myroot会进入 root目录

动态链接库的时候会看到很多这种方式

history 查看过去的指令 history 10 最近十条指令

！5看历史的时候输入这个会执行历史的第五条指令

时间日期类相关指令

data +%Ymd看年月日

还有一个显示年月日时分秒的

data -s "2020-1103-03 20:02:10"设置时间日期

cal 查看日历 cal 2020看一整年

find指令 从指定目录将满足条件的目录显示在终端

find [文件目录] 选项-name按文件木查找 -user按用户查找 -size按文件大小查找

ls -LH 按人类可读方式显示

locate快速定位文件路径 第一次使用locate的时候要下载locate的数据库 要用updatedb

which指令 可以查看指令在哪个目录下 如which ls

grep指令和管道符号 写在|后面 将前一个命令的结果给管道后面继续做

grep[选项-n显示行号 -i忽略大小写] 查找内容 源文件

cat a.txt|grep "hello" 源文件就是前面的哪个文件

#！压缩和解压类

gzip 文件 压缩文件 压缩为.gz

gunzip 文件 解压文件

zip 压缩文件和文件夹

zip [选项]xxx.zip 将要压缩的内容的路径

unzip [选项]xxx.zip 解压缩

-r递归压缩 -d指定解压的存放目录

tar [选项] xxx.tar.gz 打包的内容

-c打包 -v 显示详细信息 -f指定压缩后的文件名 -z打包同时压缩 -x 解包.tar文件

-zcvf 四个选项一起用来打包 zxvf用来一起解压 解压的时候在最后加上-C 文件路径说明解压缩到哪里

tar -zxvf /home/myhome.tar.gz -C /opt/tmp2/ 要写绝对路径

##第十章 组管理和权限管理

linux一定属于一个组，不能独立于组外

每个文件都有三个概念 所有者 所在组 其他组的概念 谁创建默认谁的可以更改 所在组就是所有者在的组 所在组里面的用户对他有一定的权限 其他组也有一定的权限

三个不同的概念 对应了三种不同程度的权限

ls -ahl查看文件的所有者 a全部 h human

chawn 用户名 文件名 把文件权限给用户

组的创建

groupadd 组名

groupadd monster

useradd -g monster fox

chagrp 组名 文件名 修改文件所在的组 文件和目录都可以一样操作

chagrp fruit orange.txt

chagrp fruit /orange/

usermod -g 新组名 用户名 修改用户所在组

usermod -d 目录名 用户名 改变该用户登录的初始目录 ！用户需要有进入到新目录的权限才可以

权限的基本介绍 重点

当使用 ls -l的时候会有

0123456789

0确定文件类型 l->link d->directory c->字符设备鼠标键盘 b->设备 比如硬盘 -代表普通文件

1-3确定所有者有用的权限 r read w write x execute可执行的权限

4-6确定所属组拥有的权限

7-9确定其他用户对该文件的权限

对目录有写的权限才可以删除文件 对文件的删除相当于对目录的写

修改权限

chmod u=rwx，g=rx，o=x 文件/目录名 user代表所有者 g代表所在组 o代表其他人

chmod +w代表赋予写的权限 文件/目录名

chmod a-x去除权限

chown newowner 文件目录 改变所有者

chown tom： /home/test/abc.txt

chown newowner：newgroup 改变所有者和所在组

chown -R tom： /home/test/

-R 将目录和文件都递归生效

chgrp newgroup新组名 文件/目录 改变他的组

chgrp shaolin /home/abc.txt

#第11章定时任务调度

crond 进行 定时任务的设置

任务调度：指定系统在某个时间执行特定的命令或者程序

例如当我有shell脚本是进行完成数据库的备份 可以用crond定时去调用这个脚本

crontab -e 编辑crontab定时任务

-l 查询任务

-r 删除任务

ls -l /etc/ > /tmp/to.txt 每分钟查看一次 etc的内容

指定执行的时间

\*/1 \* \* \* \*

\*/1表示每分钟 直接写是一个小时当中的第几分钟执行

第二个\*表示第几个小时

第三个\*表示一个月第几天

第四个\*表示一年当中的第几个月

第五个\*表示一周当中的星期

用rm删是不可以的

\*代表任何时间 就代表一小时中每分钟执行一次

代表不连续的时间 比如 0 8,12,16 \* \* \* 表示每天的八点十二点十六点都执行一次命令

-表示连续的时间 0 5 \* \* 1-6代表五点零分的周一到周六执行命令

\*/n \* \* \* \* 代表隔多久执行一次

尽量不要让几号和星期几一起用

\*实例：每隔一分钟就让当前日期信息追加到/tmp/mydate

\*/1 \* \* \* \* date>>/tmp/mydate

每隔一分钟将当前日期和日历都玩写道mycal去

用脚本完成my.sh 对root有执行权限

at定时任务 crontab编辑的任务是反复执行的

而at定时任务是一次性定时计划任务

维护一个任务队列如果里面有任务而且任务与当前时间匹配则启动

at执行要保证atd在运行

ps -ef |gerp atd查看当前的进程然后过滤看看有没有atd

at [选项] [时间]

ctrl+d 结束at命令的输入

-midvcVqft

时间指定hh:mm的时间的指定 指定当天某时某刻执行

midnight noon teatime 等比较模糊的词语来指定

具体指定 04：00 mm/dd/yy 2021/01/03年月日指定 日.月.年

或者用today tomorrow指定还有相对计算时间法

atrm 编号 删除任务

文件路径 指令/脚本

#第十二章 linux挂载分区和磁盘相关的知识

linux 归根结底只有一个根目录

采用了一个载入的处理方法 文件系统中包含了一整套的文件和目录且

将一个分区和一个目录联系起来

这时要载入的一个分区将是他的存储开机在一个目录下获得

lsblk查看挂载 一个目录对应一个硬盘分区 linux份ide和scsi

标识符hdx~ hd表示分区所在的设备类型 x为盘号 ~表示分区

1-4是主分区或者扩展分区 5开始就是逻辑分区

x有四种 基本盘 基本从属盘 辅助主盘 辅助从属盘

挂载的操作

虚拟机添加硬盘

分区 fdisk /dev/sdb dev是设备文件 sdb表示的是新的那块硬盘

然后选一个命令 n 添加新的分区 p是主分区 默认一 一个分区

回车后 再回车 再回车 再输入命令w 写入分区退出

格式化 mkfs -t ext4 /dev/sdb1

ext4是分区类型

挂载 先创建一个nedisk的目录 mkdir newdisk

mount /dev/sdb1 /newdisk/

设置可以自动挂载

卸载 用umount

'用命令行挂载的方式重启后会失效'

在/etc/fstab文件中写入如何挂载信息 然后重启就一直存在了

df -h可以查看磁盘的信息

使用率到了百分之八十以上就不正常了

du -h /目录 查目录的磁盘的占用情况

-s 指定目录的大小汇总

-h 带计量单位

-a 含文件

--max-depth=1子目录深度

-c s+a的汇总

#linux网络配置

虚拟机有一个ip

母鸡跟他有一个相同网段的ip 两者可以ping通

母鸡上有一个无线网卡负责与外界沟通

虚拟机ip-》母鸡对应ping通ip-》无线网卡

编辑和修改地址

linux网络环境配置 通过界面设置自动获取ip 每次启动的ip可能不一样不适合做服务器

/etc/sysconfig/network-scripts/ifcfg-ens33

BOOTPROTO=static

IPADDR=192.168.200.130

网关

GATEWAY=192.168.200.2

域名解析器

DNS1=192.168.200.2

然后再vm里面也要修改母机与虚拟机网关沟通的ip改成相同的网段

VMNET8的网关沟通的对象是： 192.168.200.2

linux默认网关是.2

设置主机名和hosts映射

hostname可以看到主机名

设置主机映射

就是想办法将ip地址和主机名字邦栋

windows剋再c/widnows/sysytem32/drivers/etc/hosts 里面指定

192.168.200.130 hypnus 这样我们的主机就能从主机名ping通了linux虚拟机

/etc/hosts 文件指定 让linux能pin通主机

192.168.200.1 windows主机名

hosts是一个文本文件用来记录ip和hostname（主机名）的映射关系

DNS是域名系统 是一个域名与IP地址相互映射的一个分布式数据库

#14章进程管理

linux每一个执行的程序都分配一个id号 每个进程都可能以两种方式存在的

#ps -aux 查看进程正在执行的进程 的详细信息

pid进程识别号

%CPU占用cpu的百分比 %MEM占用物理内存的百分比 占用物理内存的情况 vsz 占用虚拟内存的情况 RSS TTY终端信息？

STAT运行状态 S->sleep r->running command 执行的命令 z-僵尸进程 T-被跟踪 D-短期等待 N-低优先级进程 s-表示该进程是会话的先导进程

ps -aux |grep sshd

ps -ef 以全格式显示当前所以的进程 PID 进程号 PPID父进程ID C是当前任务优先级

pid 1进程就是系统后台进程 并且由他创建了许多进程

终止进程 kill [选项] 进程号 终止进程

killall 进程名称（诛九族 还支持通配符 者使得系统因负载过大而变慢时候可以杀死

-9强制进程立即停止

1.踢掉非法登录的用户 ps -aux|grep sshd sshd能找到用户登录进程号 然后 kill 非法用户的进程号

2.终止远程登录服务sshd 在适当的时候再次重启sshd服务 /bin/systemctl start sshd.service

pstree指令 查看进程树木

pstree [-g -u] ID号和用户

Service管理 服务管理 本质就是进程但是运行在后台

通常都会监听某个端口 等待其他程序的请求 如mysql sshd 防火墙 因此我们又称为守护进程 是linux中非常中啊哟的知识点

service 服务名 [start|stop|restart|reload|status]

centos7以后用systemctl管理

3 /etc/init.d 下面的程序都能用service管理

start是让服务启动 stop是关闭

setup 可以查看所有的系统服务 有\*号的都会自动启动 选中\*号然后空格就会去掉

systemctl get-default 查看当前的运行级别

multi-user.target 运行级别3

graphical.target 运行级别五

systemctl set-default 可以设置

chkconfig 可以给服务的各个运行级别设置自启动/关闭

chkconfig的命令 在init.d里面查看

chkconfig --list

chkconfig 服务名 --list

chkconfig --level 3 network on 在运行级别三进行自启动

sysyatemctl管理

sysytemctl [start|stop|restart|status] 服务名

他能管理的服务在/usr/lib/systemd/system下面查看

sysytemctl list-unit-files 查看开机自启动状态

systemctl enable 服务名 设置开机自启动

systemctl disable 服务名 关闭服务开机启动

systemctl is-enabled 服务名 查询某个服务器是否自启动

代表控制了三和五

防火墙

防火墙打开某个端口

fierwall-cmd --permanent --add-port=proto(协议) add添加 remove关闭 query是查看端口协议

netstat -anp|more 看端口协议

firewall-cmd --reload 打开关闭后要重新载入一次

动态监控进程

'top 用来显示正在执行的进程 且top可以更新 ps相当于快照不可以更新'

top [-d] 秒数 每隔xs刷新

-i不显示任何限制或者僵尸进程

top会显示 当前时间 系统运行时间 多少用户在使用这个系统 负载均很（0.00 +0.01 +0.05）/3<0.07表示正常

cpu us 用户占用 sy系统占用 id空闲占用

men 内存占用

动态监控的交互

p cpu占用排序

m 内存使用率排序

n number pid排序

q 退出top

u 选择用户

k kill 击杀pid

'监控网络状态'

netstat -an all number 按顺序排列输出 -p显示哪个进程在调用

ping 检查网络能不能通

#第15章 rpm与yum

rpm用于互联网下载包的打包以及安装工具，他包含在某些linux 就是软件的程序 centos里面有很多rpm包

rpm -qa|grep xx 当前系统安装了某个xx程序有没有 查询后可以看到版本号 系统 支持什么系统i686 i386是三十二位系统

noarch表示通用 centos7.x表示64为系统

more是分页指令

加软件包名直接看他能不能被查询到

rpm -qi 软件包名 查询软件包信息

-ql 查询软件包中的文件

-qf 查询文件归属于哪个软件包

rpm -e 包名称

rpm -ivh 指向rpm包全路径名称要有rpm的安装文件

光盘的packages里面有rpm包

Yum是shell前端软件包管理器 能够从指定的服务器自动下载RPM包并且安装

可以自动处理依赖性关系，并且一次安装所有依赖的软件包

yun list|grep xx软件列表

rpm -e 包名称删除

yum list|grep firefox 查询yum里面有没有firefox的安装包

yum install firefox并且下载

#软件安装 我没干 第16章我没干

#17章 shell编程

shell是一个命令行解释器，他能给用户向linux内核发送请求以便运行程序的界面系统级程序

用户可以用shell来启动 挂起 停止甚至是编写一些程序

当不止一条程序的时候 我们可以用.shell文件一次执行多个指令

shell脚本

1.以#!/bin/bash开头

2.脚本有可执行权限chmod u+x 文件路径

赋予+x去除权限或者sh+脚本

shell变量

系统变量和用户自定义变量

$HOME $PATH $SHELL $USER

用set可以看到

定于变量 变量名=值

unset 变量名 readonly变量 不能unset

输出变量前面要加上$

echo--printf

变量名称不能以数字开头

不允许在变量赋予的时候有空格

将命令的返回值赋予给a

A=$(date)==`date`

export 变量名=变量值 将shell变量输出为环境变量 也就是全局变量 这样就可以被多个文件共享

要写到 /etc/profile文件夹里面去

然后source刷新下

source 配置文件 让修改后的配置信息立即生效

echo $变量名 查询环境变量的值

<<!

多行注释可以这么写

!

find-name 文件名 找文件

位置参数变量

如果希望获取到命令行的参数信息 就是希望我动态传入信息给脚本 类似cin

$n n代表数字$1-9 ${>10}代表命令行中的第几个参数 $0代表命令本身

$\* 这个变量代表拿走命令行中所以的参数 一次拿完

$@ 也是拿走所有参数 但是每个参数会区分对待 一次拿完

$# 这个变量代表命令行中所有的参数个数 统计传入了多少个参数

预先定义变量

$$ 获取pid

$! 最后一个后台进程的进程号

$? 返回上一个命令对不对 0对 非错误

运算符

$((运算式)) $[运算式子] expr 就是expression m + n 第三个式子需要有空格隔开 而且如果希望将结果赋给别的变量使用``这个是波浪号键打出来的

expr \\*,/,% 乘除 取余 \是转义字符

条件判断

基本语法

[ condition ] 注意条件前后都要有空格

非空返回true #?验证 0为真 >1false

[ condition ] && echo OK ||echo notok 条件满足执行后面的语句

判断语句

= 字符串比较

两个整数比较

-lt -le -eq -gt -ge -ne不等于 替代了大于小于符号

按照文件权限进行判断

-r -w -x 可读 可写 可执行

4 按照文件类型进行判断

-f 常规文件存在

-e 存在

-d 文件存在并且是目录

if [ condition ]

then

语句

fi fi表示if的结束

多分支加 elif [ 条件判断式 ]

then

流程控制 格式一定要一样

case $变量名 ini

"值1")

如果变量值等于值1则执行程序1

;;

"值二"）与1一样

;;

\*)

如果值都不是上面的则执行这个

;;

esac

for 变量 in 值1 值2

do

程序/代码

done

范围怎么办 for (( 初始值; 循环控制条件; 初始值变化))

do

代码

done

while [ 条件判断式 ]

do

程序

done

注意程序的空格

read读取控制台输入

read(选项)(参数)

-p 指定读取值时候的提示符

-t 指定等待读取时间

函数

1.系统函数

basename [pathname] [suffix] suffix是后缀

返回完整路径斜杠后面的部分 常用来获取文件名

basename /home/aaa/test.txt .txt用来获取文件名

dirname 常用于路径

dirname 路径

自定义函数

基本语法

function funname()

{

Action;

[return int]

}

funname [值] 值就是变量 $n1 这样传

local 关键词创建局部变量防止被修改

传参数仍然是用 $n1 来传递

#21 日志管理

日志文件是重要的信息文件 记录了许多重要的系统事件

日志对于安全也很重要 他·记录了系统每天发生的各种事情，通过日志来检查错误发生的原因，或者受到攻击时攻击者留下的痕迹

常用日志放在 /var/log/ 目录就是系统文件的保存的位置

boot.log 系统启动的日志

cron 记录了与系统定时任务相关的日志

cups文件夹记录了打印信息的日志

demesg 记录了系统在开机时内核自检的信总，也可以使用dmesg命令直接查看

btmp 记录了错误登录的值日志 是二进制文件 要使用latb命令查看 命令如下#lastb

lasllog记录系统中所有用户最后一次登录时间的日志，这个文件也是二进制文件.要使用lastlog命令查看

mailog记录邮件信息的日志

message记录系统重要消息的日志 这个日志文件中会记录linux系统的绝大多数重要信息 如果系统出现问题，首先要检查这个日志文件

sevure 记录验证和授权方面的信息，只要涉及账户和密码的程序都会记录比如系统的登录 ssh的登录 su切换用户 sudo授权 添加用户和修改用户密码都会记录

wtmp 永久记录所有用户的登录 注销信息 同时记录系统的启动 重启 关机时间 要使用lastlog命令查看umlp 记录当前已经登录的用户的信息是动态的，治记录当前登录用户的信息

这个文件要使用 users who w等命令查看

rsyslogdv 日志管理服务 他启动了才会记录日志

/etc/rsyslog.conf 记录了rsysklogdv 要怎么管理

查询 ps aux|grep "rsyslog"|grep -v "grep"查询某个服务器是否自启动

systemctl list-unit-files|grep rsyslog

了解 rsyslog.conf的配置文件

\*.\* 第一个表示日志的类型 第二个表示日志的级别

配置文件会写把什么样的文件 \*.\* 所有文件 存到指定的文件目录 还能自定义

鹰击长空鲸霸海，不试怎知龙与蚯，祝各位曹老师的学生们都能考到自己理想的成绩，剩下的这段时间一定要相信自己，不到最后一刻你的成绩都没有定型，会创造奇迹，

日志记录 时间 主机名 事件发生的程序或者服务 对事件的描述

日志论替

把旧的日志文件移动并改名，同时建立新的空日志文件

当旧日志文件超出保存的范围之后，就会进行删除

centos7使用logrotate进行日志轮替管理，想要改变日志论题文件名字，通过/etc/logrotate.conf的配置文件

中的 "dateext"参数;

/etc/logorotate.conf [全局的日志轮替策略，也可以单独给]

单独的日志文件配置可以写到

/etc/logrotate.d/ 这个文件夹中 推荐 创建的文件无后缀

2.如果配置文件中有 这个参数 那么日志会用日期来作为文件的后缀如secure-20201010

这样日志文件名就不会重叠，也就不需要日志文件的改名，只需要指定保存日志个数

删除多余的日志文件就可以了

如果配置文件中没有这个参数 那么日志文件就要改名了 当第一次进行轮替时

当前的日志惠子自动改名为.1 然后新建新的文件保存新的日志

文件中已经写了默认策略了

如果想单独写

绝对文件路径{

日志管理

}

daily weekly monthly轮替

rotate number 保留的日志文件个数

compress 日志轮替时 旧的压缩

create mode owner group 建立新的日志 指定新日志的权限与所有者和所属组

当日志轮替时，输出内容通过邮件发送到指定的邮件地址

mail adress当日志轮替时输出内容通过邮件发送

missingok 日志不存在则忽略该日志的警告信息

notifempty 空文件不轮替

minisize 大小 轮替发生的最小值 不超过不轮替

size 大小 轮替大于指定大小才轮替 而不是按照时间

dateext 日期当文件后缀

sharedscripts 在此关键字后的脚本只执行一次

prerotate/endscript 日志轮替前执行脚本命令

postrotate/endscript 日志轮替后执行脚本命令

脚本就是shell脚本

日志轮替的机制原理

日志轮替之所以可以在指定的时间备份日志，是依赖系统的定时任务

在/etc/cron.daily/目录下会发现有个logrotate文件可以执行 logrotate通过这个文件依赖定时任务执行的

我们会跑crond 这个后台程序

他回去定时的执行 cront.daily

在cron.daily里面有个logrotate程序他会去读取轮询策略然后去进行轮替

查看内存日志

journalctl 可以查看内存日志

-n 3 查看最新三条

-since 19:00 --until 19:10查看起始时间到结束日期的日志

-p err看报错日志

-o verbase看日志详细内容

\_PID=1245 \_COMM=SSHD

journalctl| grep ssh 查看包含这些参数的日志

重启后会情况

22章 定制自己的linux系统

通过裁剪现有的linux系统centos7.6

在制作linux小系统前

首先自己检查 检查设备有没有故障

如果有有多块启动盘的话，需要在bios中选择启动磁盘 启动mbr的bootloader引导程序

加载内核文件

执行所有进程的父进程，老祖宗systemd

欢迎界面

制作min linux思路分析

在现有的linux系统上加一块硬盘/dev/sdb，在硬盘上分两个分区，一个是/boot

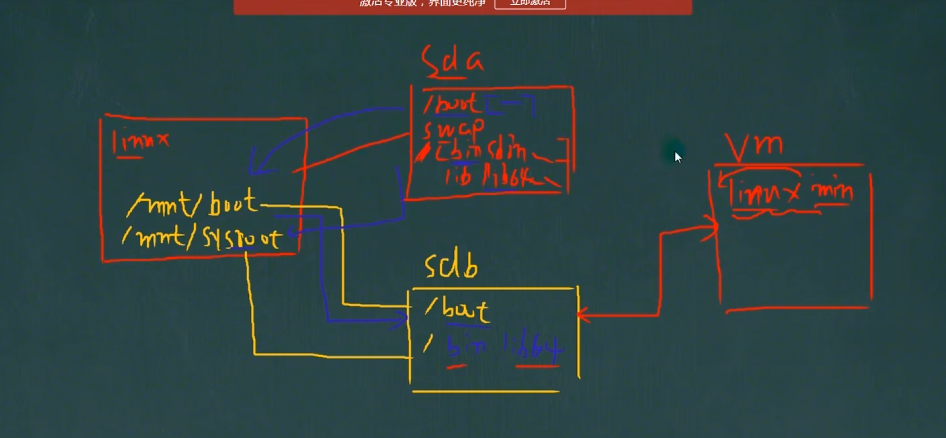
一个手/，并且将其格式化。需要明确的是现在加的这个硬盘在现有的linux系统中是/dev/sdb

但是当我们把东西全部设置好的时候，要把这个硬盘拔除，然后放在新的系统上就是/dev/sda

在/dev/sdb硬盘上，将其打造成独立的linux系统，里面所有的文件是需要拷贝进去的

作为能独立运行的linux系统 内核一定不能少 ，要把内核文件和initramfs 文件 也一起拷到/dev/sdb上

以上步骤完成，我们的自启制linux就完成了，创建一个新的linux虚拟机 将其硬盘指向我们创建的硬盘启动即可



首先先增加一块硬盘

Vm里面选择 注意要把虚拟磁盘存储为单个文件

因为他过会要作为一个引导盘

分好以后进入系统分区

首先用lsblk+

首先分区

fdisk /dev/sdb

N

P 主磁盘

+500m boot分区

剩余给当做root分区

mkfs.ext4/dev/sdb1

mkfs.ext4/dev/sdb2

对分区进行格式化

然后创建目录 并且挂载新的磁盘

mkdir -p /mnt/boot /mnt/sysroot

Mount /dev/sdb1 /mnt/boot

Mount /dev/sdb2 /mnt/sysroot/

安装grub 内核文件拷贝到目标磁盘

Grub2-install --root-directory=/mnt/dev/sdb

Hexdump -C -n 512 /dev/sdb 验证一下有没有装上

然后开始拷贝 boot文件

# rm -rf /mnt/boot/\*

#cp -fr/boot/\* /mnt/boot/

然后修改 grub2/grub.cfg

#lsblk -f 查看

将grub里面的 两个ext4 分区的 uid改成新开辟的分区 boot 改新boot root 对应的新boot

要改所有对应的地方

然后有linux16的地方 utf-8 后面 加上 selinux=0 init=/bin/bash

Linux 16 那一行的后面 也要加上 selinux=0 init=/bin/bash 防止系统走默认路线 应该走我自己配置的引导路线

然后创建目标主机的根文件系统

#mkdir -pv /mnt/sysroot{etc/rc.d,usr,var,proc,sys,dev,lib,lib64,bin,sbin,boot,srv,mnt,media,home,root}

拷贝需要的bash 命令处理器

#cp /lib64/\*.\* /mnt/sysroot/lib64/

#cp /bin/bash /mnt/sysroot/bin/

搞定以后基本的都有来了

现在创建新的虚拟机 将默认的磁盘分配移出 指向我们之前建造的这个 然后 想加新的 先挂载 再拷贝过去

#linux内核源码介绍和内核升级

看几个核心模块

看0.01

阅读技巧

1看得懂c语言

1. 应该知道内核源码的整体分布情况
2. 现在操作系统一般由进程管理，内存管理，文件系统，驱动程序和网络组成
3. Linux内核源码的各个目录大致与此相对
4. 阅读方法和顺序有横向和纵向

纵向就是按执行顺序逐步进行

横向就是按模块来看

对于linux的启动代码就按顺序一步步读 内存管理部分可以单独拿出来进行阅读分析实际上这是一个反复的过程

Init有核心初始化方法 kernel与内核相关 lib与库相关 tools与

Mm与内存相关 makefile 与编译相关

内核升级

Unmae -q 查看当前内核版本

首先 yum info kernel -q //检测内核版本 显示可以升级的内核版本

Yum update kernel 升级内核

Yum list kernel -q 查看已经安装的内核

重启后会选择进入哪个内核 所以非覆盖

实体机没有办法做快照 所以应该如何备份呢？

把需要的文件或者分区 用tar打包

使用dump和restore 指令进行备份

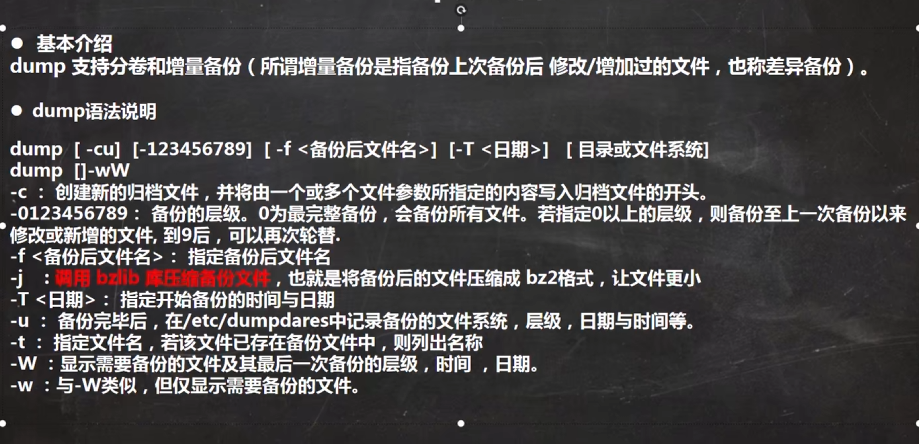
Yum -y install dump

Yum -y install restore

dump支持分卷和增量备份 是指只备份与上次备份不同的文件地方

0 为最初的层级 1-9是九次小版本更新 到9以后进行更替

Dump -0uj -f /opt/boot.bak.bz2 /boot 将boot分区备份到opt下面去



-f 加备份后的文件名 需要备份的文件是什么

分区可以增量备份 文件目录不支持增量备份

数据恢复 用dump生成的文件中恢复源文件

Restore 选项-C 对比模式 备份文件和已经存在的文件相互对比，在进行还原操作时候 restors -i使用交互模式 -r使用还原模式 -c查看备份文件中有哪些文件

-f <备份设备> 选择从哪个备份文件中进行恢复

#25章 webmin和bt工具

Webmin是功能强大的基于web的unix/linux系统管理工具 管理员通过浏览器访问webmin的各种管理功能并完成相应的管理操作

