# Linux的发行版本

1. Red Hat Linux(红帽)

红帽是目前最为认可的linux品牌。Red Hat 公司的产品主要包括 RHEL（Red Hat Enterprise

Linux，收费版本）和 CentOS（RHEL 的社区克隆版本，免费版本）以及Fedora Core（由

Red Hat 桌面版发展而来，免费版本）。

2. Ubuntu Linux(乌班图)

Ubuntu 是基于 Debian Linux 发展而来，界面友好，容易上手，对硬件的支持非常全面，是

目前最适合做桌面系统的 Linux 发行版本，而且 Ubuntu 的所有发行版本都免费提供。

3. SuSE Linux

SuSE Linux 以 Slackware Linux 为基础，早期由德国的 SuSE Linux AG 公司发布，只有商业版

本。2004 年被 Novell 公司收购，成立了 OpenSUSE 社区，推出了社区版本 OpenSUSE。

4. Gentoo Linux

Gentoo 最初由 Daniel Robbins创建，首个稳定版本发布于 2002 年。在所有的linux发行版

中Gentoo是安装最复杂的，到目前为止仍采用源码包编译安装操作系统。

# CentOS版本的下载地址

CentOS7.6：

<http://mirrors.163.com/centos/7.6.1810/isos/x86_64/CentOS-7-x86_64-DVD-1810.iso>

CentOS8.1：

<https://mirrors.aliyun.com/centos/8.1.1911/isos/x86_64/CentOS-8.1.1911-x86_64-dvd1.iso>

得打开迅雷下载，直接打开网页不行。

# Linux7.6的安装

视频地址：

<https://www.bilibili.com/video/BV1Sv411r7vd?p=6&spm_id_from=pageDriver&vd_source=11e4dce1b79b451398ab2c230943e3e6>

本地：E:\video\linux\centos7.6安装.mp4

# 虚拟机的基本操作

## 1.虚拟机的克隆

比如：你想要做集群，则需要多个Linux操作系统环境，如果你已经安装好了一个linux环

境，则可以通过克隆来生成多个一模一样的linux环境。

方式1：直接拷贝一份安装好的虚拟机文件。

本地是路径F:\Virtual Machines\Red Hat Enterprise Linux7下的Red Hat Enterprise Linux7

文件夹，当你把这个文件夹拷贝一份放到其他地方，改下文件夹的名字，然后就可以用

vmware直接打开文件夹里面后缀为vmx的文件即可。

方式2：使用vmware的克隆操作。注意：克隆时，需要先关闭linux系统。

步骤：右击左边列表中你的虚拟机名称->管理->克隆，选择 创建完整克隆，指定虚拟

机的名称和位置，点击完成即可。

## 2.虚拟机的快照

快照也就是虚拟机的运行状态，保存快照也就是保存当前虚拟机的运行状态。

比如：在安装好linux系统后，先创建一个快照A，如果之后误操作导致系统异常，则可以

选择回到快照A，来消除异常。

创建快照：右击左边列表中你的虚拟机名称->快照->拍摄快照，然后指定名称和描述，点击

拍摄快照即可。

使用快照：右击左边列表中你的虚拟机名称->快照->点击 恢复到快照 或者在快照管理器中

选择一个快照，点击 转到。

## 3.虚拟机的迁移和删除

迁移：虚拟系统安装好之后，它本质就是一个文件(存放在文件夹中)，因此虚拟系统的迁移

只需要拷贝这个文件夹到另外的位置即可。

删除：可以直接删除虚拟系统安装的文件夹。也可以用vmware进行删除。操作：右击左边

列表中你的虚拟机名称->管理->从磁盘中删除。

## 4.安装vmtools

(1)安装vmtools后，可以更好的在windows下管理vm虚拟机。

(2)可以设置windows和linux共享文件和文件夹。

(3)视频地址：

https://www.bilibili.com/video/BV1Sv411r7vd/?p=11&spm\_id\_from=pageDriver&vd\_source=11e4dce1b79b451398ab2c230943e3e6

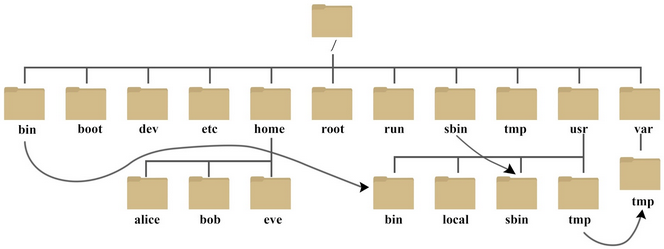


(4)设置文件夹共享：右击左边列表中你的虚拟机名称->设置->选项->共享文件夹->总是启用

->添加需要共享的文件夹，然后在linux中的计算机/mnt/hgfs文件夹下就能看到共享的

文件夹。

# Linux的目录结构(了解)



/bin：bin是Binaries (二进制文件)的缩写，该文件夹存放常用命令(即二进制可执行程序)。

/boot：这里存放的是启动 Linux 时使用的一些核心文件，包括一些连接文件以及镜像文件。

/dev：dev是Device(设备)的缩写, 该目录下存放的是Linux的外部设备，在Linux中访问设备的方式和访问文件的方式是相同的。

/etc：etc是Etcetera(等等) 的缩写，这个目录用来存放所有的系统管理所需要的配置文件和子目录。

/home：用户的主目录，在Linux中，每个用户都有一个自己的目录，一般该目录名是以用户的账号命名的，如上图中的alice、bob 和eve。

/lib：lib是Library(库) 的缩写，这个目录里存放着系统最基本的动态连接共享库，其作用类似于Windows里的DLL文件。几乎所有的应用程序都需要用到这些共享库。

/lost+found：这个目录一般情况下是空的，当系统非法关机后，这里就存放了一些文件。

/media：linux系统会自动识别一些设备，例如U盘、光驱等等，当识别后，Linux会把识别的设备挂载到这个目录下。

/mnt：系统提供该目录是为了让用户临时挂载别的文件系统的，我们可以将光驱挂载在该文件夹上，然后进入该目录就可以查看光驱里的内容了。

/opt：opt是optional(可选)的缩写，这是给主机额外安装软件所摆放的目录。比如你安装一个ORACLE数据库则就可以放到这个目录下。默认是空的。

/proc：proc是Processes(进程)的缩写，/proc是一种伪文件系统(也即虚拟文件系统)，存储的是当前内核运行状态的一系列特殊文件，这个目录是一个虚拟的目录，它是系统内存的映射，我们可以通过直接访问这个目录来获取系统信息。  
这个目录的内容不在硬盘上而是在内存里，我们也可以直接修改里面的某些文件，比如可以通过下面的命令来屏蔽主机的ping命令，使别人无法ping你的机器：

echo 1 > /proc/sys/net/ipv4/icmp\_echo\_ignore\_all

/root：该目录为系统管理员，也称作超级权限者的用户主目录。

/sbin：s就是Super User的意思，是Superuser Binaries (超级用户的二进制文件)的缩写，这里存放的是系统管理员使用的系统管理程序。

/selinux：这个目录是Redhat/CentOS所特有的目录，Selinux是一个安全机制，类似于 windows 的防火墙，但是这套机制比较复杂，这个目录就是存放selinux相关的文件的。

/srv： 该目录存放一些服务启动之后需要提取的数据。

/sys：这是Linux2.6内核的一个很大的变化。该目录下安装了2.6内核中新出现的一个文件系统sysfs。sysfs文件系统集成了下面3种文件系统的信息：针对进程信息的proc文件系统、针对设备的devfs文件系统以及针对伪终端的devpts文件系统。该文件系统是内核设备树的一个直观反映。当一个内核对象被创建的时候，对应的文件和目录也在内核对象子系统中被创建。

/tmp：tmp是temporary(临时)的缩写这个目录是用来存放一些临时文件的。

/usr： usr是unix shared resources(共享资源)的缩写，这是一个非常重要的目录，用户的很多应用程序和文件都放在这个目录下，类似于windows下的program files目录。

/usr/bin：系统用户使用的应用程序。

/usr/sbin：超级用户使用的比较高级的管理程序和系统守护程序。

/usr/src：内核源代码默认的放置目录。

/var：var是variable(变量) 的缩写，这个目录中存放着在不断扩充着的东西，我们习惯将那些经常被修改的目录放在这个目录下。包括各种日志文件。

/run：是一个临时文件系统，存储系统启动以来的信息。当系统重启时，这个目录下的文件应该被删掉或清除。如果你的系统上有/var/run目录，应该让它指向run。

补充：在Linux系统中，有几个目录是比较重要的，平时需要注意不要误删除或者随意更改内部文件。

/etc：上边也提到了，这个是系统中的配置文件，如果你更改了该目录下的某个文件可能会导致系统不能启动。

/bin, /sbin, /usr/bin, /usr/sbin: 这是系统预设的执行文件的放置目录，比如 ls 就是在/bin/ls 目录下的。

值得提出的是/bin、/usr/bin 是给系统用户使用的指令（除 root 外的通用用户），而/sbin, /usr/sbin 则是给 root 使用的指令。

/var：这是一个非常重要的目录，系统上跑了很多程序，那么每个程序都会有相应的日志产生，而这些日志就被记录到这个目录下，具体在 /var/log目录下，另外 mail 的预设放置也是在这里。

# Xshell7

Xshell7是常用的远程连接Linux的一个工具软件。个人用户(非商用)可以下载免费版。网址：

<https://www.xshell.com/zh/free-for-home-school/>

在刚开始时需要创建连接，然后填写Linux的用户名/密码才可以正式连接到Linux服务器。在连接时需要填写Linux服务器的IP地址。

# Xftp7

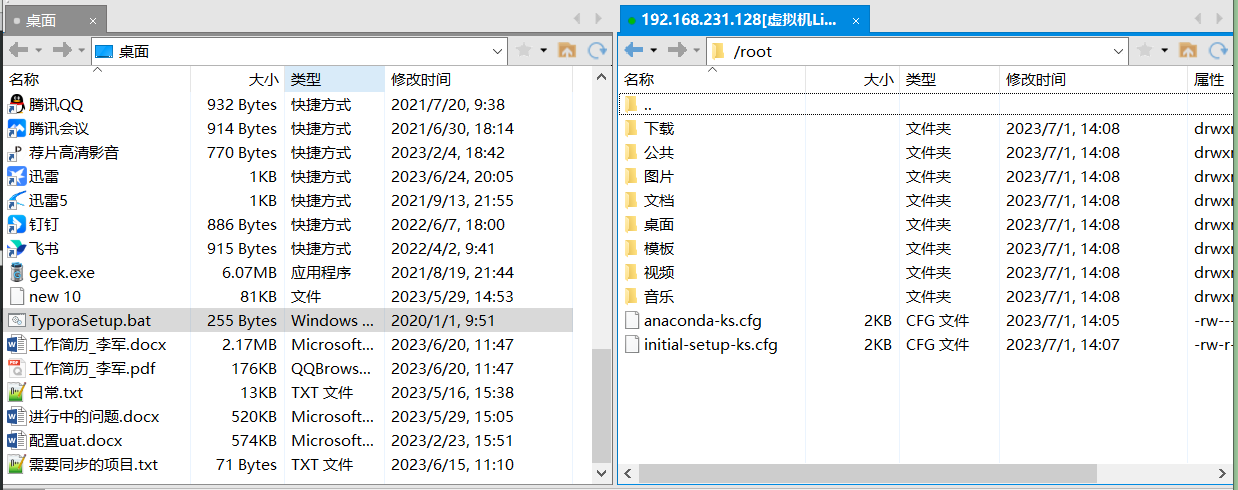
Xftp7是一个文件传输软件，可以在windows和Unix/Linux系统之间传输文件。网址：

<https://www.xshell.com/zh/free-for-home-school/>

跟Xshell7一样，在刚开始时需要创建连接，然后填写Linux的用户名/密码才可以正式连接到Linux服务器。在连接时需要填写Linux服务器的IP地址。

补充：如果连接后出现的中文文件夹的汉字名称出现乱码，解决方式是：

左上角文件>当前会话属性>选项里选择编码集为UTF-8。



上图是连接成功且解决中文乱码之后的视图，其中左框是windows10的桌面，右框是虚拟机的Linux的root目录，想传输文件，直接互相拖拽即可，或者在文件上右击，选择传输。当然，正常的Linux工作服务器一般是不会安装可视化桌面的，只有终端命令行。

# FinalShell

FinalShell也是远程连接的工具软件，使用java开发，有Xshell7和Xftp7二者的功能，因此选择FinalShell更适用。优点是免费，缺点是占用内存比较大。

# Linux基础命令

1.查看ip：ifconfig

2.vim相关(文本命令)：

(1)vim 文件名 ：打开文件，如果文件不存在则创建并打开文件，打开文件后都是处于命令模式。

(2)按下i、I、o、O、a、A等任何一个字母之后会进入编辑模式。在编辑模式下可以向

文本中手动添加内容。

(3)添加完内容后，按下esc，可以切换到命令模式。

(4)此时输入:wq再回车表示保存退出；

输入:q再回车表示退出；

输入:q!再回车表示强制退出。

3.yy拷贝行(文本命令)

(1)使用vim打开文件后(不进入编辑模式)，输入yy则可以拷贝当前光标所在行，再输

入p则粘贴拷贝的行。

(2)如果想拷贝多行，比如拷贝光标所在行以及其下共5行，则输入5yy即可，再输入p

则粘贴拷贝的行。

4.dd删除行(文本命令)

(1)使用vim打开文件后(不进入编辑模式)，输入dd则可以删除当前光标所在行。

(2)如果想删除多行，比如删除光标所在行以及其下共5行，则输入5dd即可。

5.文本查询关键字命令

(1)使用vim打开文件后(不进入编辑模式)，输入/，然后输入要查询的关键字，然后回

车即可，查询到的关键字会高亮显示，再输入n则可以切换到下一个匹配的文本中的关

键字。

6.文本命令行号显示

(1)使用vim打开文件后(不进入编辑模式)，再输入:set nu 回车即显示行号；

(2)如果输入:set nonu回车即关闭行号。

7.定位文本行首、行尾

(1)使用vim打开文件后(不进入编辑模式)，输入gg即可定位到行首。

(2)输入G即可定位到行尾。

8.文本撤销命令

(1)使用vim打开文件后并输入任意一个字母进入编辑模式，在编辑一段文字后，如果

不想要这段文字了，可以按下esc键进入命令模式，再输入u即可撤销刚才编辑的文字。

9.快速移动光标

(1)使用vim打开文件后(不进入编辑模式)，此时想让移动到第20行，可以输入20，再

按下shift+g 即可。

10.关机/重启命令

(1)shutdown -h now 立即关机

(2)shutdown -h 1 1分钟后关机

(3)shutdown -r now 立即重启

(4)halt 关机

(5)reboot 立即重启

(6)sync 将内存中的数据同步到磁盘。

提示：目前shutdown、halt、reboot等命令均在关机前进行了sync，但还是推荐在使用

关机/重启命令前手动使用sync命令同步下数据。

11.显示当前所在路径

(1)使用pwd，可以显示当前所在路径。

12.切换用户命令

(1)登录时尽量不要使用root用户，因为此用户权限最大，容易误操作。

(2)在普通用户登录时，如果想使用root用户，可以用 su - 用户名 的方式切换用户。

13.添加用户

(1)使用root用户登录时，使用useradd 用户名 的方式可以添加用户，添加用户后默认在home文件夹下创建一个以用户名命名的文件夹，此文件夹为用户的家目录。

(2)使用root用户登录时，使用useradd -d 指定目录 用户名 的方式可以添加用户并指定其家目录的位置。

(3)添加用户后，使用passwd 用户名 的方式可以设置该用户的密码。

(4)使用新添加的用户登录后，打开终端，则默认所在路径是其家目录下。此时可以使用pwd命令显示当前所在路径。

14.删除用户

(1)删除用户，但是保留其家目录。在root用户下使用userdel 用户名 的方式删除。

(2)删除用户以及其家目录，在root用户下使用userdel -r 用户名 的方式删除。

15.查询用户信息

命令：id 用户名

16.查询首次登录的用户

命令：who am i

如果首次登录的用户是A，然后使用su命令切换到用户B，那么使用who am i时，仍

然显示的是用户A的信息。

17.用户组

(1)用户组类似于角色，系统可以对有共性/权限的多个用户分到一个组。

(2)新增组：groupadd 组名

(3)删除组：groupdel 组名

(4)添加用户时指定用户组：useradd -g 用户组 用户名

(5)切换用户组：usermod -g 用户组 用户名

(6)用户和组相关文件

/etc/passwd文件：用户的配置文件，记录用户的各种信息

每行的含义：用户名:口令:用户标识号:组标识号:注释性描述:主目录:登录Shell

/etc/shadow文件：口令的配置文件

每行的含义：登录名:加密口令:最后一次修改时间:最小时间间隔:最大时间间隔:警

告时间:不活动时间:失效时间:标志

/etc/group 文件：组的配置文件，记录Linux包含的组的信息

每行的含义：组名:口令:组标识号:组内用户列表

18.指定运行级别

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【只有命令界面，无图形界面】

4：系统未使用，保留给用户

5：图形界面

6：系统重启

常用的是3和5，可以切换运行级别，使用init[0123456] 来切换。

指定默认的运行级别（需要重启）：

(1)在centos7以前，在/etc/inittab文件中可以指定默认的运行级别。

(2)之后的版本中，指定的方式发生了变化。

multi-user.target 表示运行级别3

graphical.target表示运行级别5

查询默认的运行级别（非当前的运行级别）：systemctl get-default

修改默认的运行级别（需要重启）：

例如：修改默认的运行级别为3：systemctl set-default multi-user.target

19.找回root密码

视频地址：

<https://www.bilibili.com/video/BV1Sv411r7vd?spm_id_from=333.788.player.switch&vd_source=11e4dce1b79b451398ab2c230943e3e6&p=26>

本地：E:\video\linux\centos7.6找回root密码.mp4

20.帮助指令

(1)基本语法：man [命令或配置文件]。例如：man ls

(2)基本语法：help 命令。例如：help cd

21.pwd指令

基本语法：pwd

作用：显示当前工作目录的绝对路径

22.ls指令

基本语法：ls [选项] [目录]

作用：显示当前目录下所有的文件和目录

常用选项：

-a：显示当前目录下所有的文件和目录，包括隐藏的

-l：以列表的方式显示信息

-h：将文件大小以KB、MB、GB显示，而非字节数

例如：ls -al /boot

23.cd指令

基本语法：cd [参数]

作用：切换到指定目录

举例：

(1)cd ~ 或 cd：切换到当前用户的家目录

(2)cd .. 或 cd ../：回到当前目录的上一级目录

24.mkdir指令

基本语法：mkdir [选项] 要创建的目录

作用：创建目录

常用选项：

-p：创建多级目录

例如：mkdir -p /home/animal/tiger

25.rmdir指令

基本语法：rmdir [选项] 要删除的空目录

作用：删除空目录

例如：rmdir /home/animal/tiger

提示：如果要删除非空目录，使用rm -rf要删除的目录

26.touch指令

基本语法：touch 文件名称

作用：创建空文件

例如：touch hello.txt

27.cp指令

基本语法：cp [选项] 被拷贝的文件路径 拷贝到的文件路径

作用：拷贝文件到指定目录

常用选项：

-r：递归复制整个文件夹

例如：

(1)cp hello.txt /home/bbb ：将hello.txt拷贝到/home/bbb下

(2)cp -r /home/bbb /opt ：将/home/bbb拷贝到/opt下

(3)\cp -r /home/bbb /opt ：将/home/bbb拷贝到/opt下,如果有重名的文件，直接覆

盖，不再一个文件一个文件的提示是否覆盖。

28.rm指令

基本语法：rm [选项] 要删除的文件或目录

作用：删除文件或目录

常用选项：

-r：递归删除整个文件夹

-f：强制删除，不提示

例如：

(1)rm /home/hello.txt ：删除home目录下的hello.txt

(2)rm -rf /home/bbb ：递归删除home目录下的bbb目录，且不提示。

29.mv指令

基本语法：

(1)mv 旧文件/目录名称 新文件/目录名称：重命名

(2)mv 被移动的路径 移动到的路径：移动

作用：移动文件/目录，或者重命名文件/目录

例如：

(1) mv hello.txt he.txt ：将hello.txt重命名为he.txt

(2) mv he.txt /root ：将he.txt移动到root目录下

(3) mv he.txt /root/hh.txt ：将he.txt移动到root目录下并重命名为hh.txt

(4) mv bbb /home/ ：将bbb目录移动到home目录下

30.cat指令

基本语法：cat [选项] 要查看的文件

作用：查看文件内容

常用选项：

-n：显示行号

例如：

(1)cat -n /etc/profile，查看etc下profile文件中的内容，同时显示行号

注意：cat只能浏览文件，不能修改文件。为了浏览方便，一般会带上管道命令，

例如cat -n /etc/profile | more。

31.more指令

作用：more指令是一个基于VI编辑器的文本过滤器，它以全屏幕的方式按页显示文本

内容，more指令中内置了若干快捷键(交互指令)，详细操作如下：

空格键(space)：向下翻一页

Enter：向下翻一行

q：立刻离开more，不再显示文本内容

Ctrl+F：向下滚动一屏

Ctrl+B：返回上一屏

=：输出当前行的行号

:f：输出文件名和当前行的行号

例如：

(1)cat -n /etc/profile | more，浏览文本，之后使用more的内置快捷键操作。

(2)more /etc/profile，浏览文本，之后使用more的内置快捷键操作。

32.less指令

基本语法：less 要查看的文件

作用：与more指令类似，但是比more指令更加强大，支持各种显示终端，且是按需

加载文本内容，这对于显示大文本具有较高的效率。less指令中内置了若干快捷键

(交互指令)，详细操作如下：

空格键(space)：向下翻一页

[pagedown]：向下翻一页

[pageup]：向上翻一页

/字串：向下搜寻字串的功能；n：向下查找；N：向上查找

?字串：向上搜寻字串的功能；n：向上查找；N：向下查找

q：立刻离开less，不再显示文本内容

例如：

(1)cat -n /etc/profile | less，浏览文本，之后使用less的内置快捷键操作。

(2)less /etc/profile，浏览文本，之后使用less的内置快捷键操作。

33.echo指令

基本语法：echo [选项] [输出内容]

作用：输出内容到控制台

例如：

(1)echo $HOSTNAME， 输出环境变量

(2)echo "11"， 输出11

34.head指令

基本语法：head 文件

作用：显示文件开头部分内容

例如：

(1) head /etc/profile，显示profile文本头10行内容

(2) head -n 5 /etc/profile，显示profile文本头5行内容

35.tail指令

基本语法：tail 文件

作用：显示文件尾部部分内容

例如：

(1) tail /etc/profile，显示profile文本末尾10行内容

(2) tail -n 5 /etc/profile，显示profile文本末尾5行内容

(3) tail -f 文件，实时监控文本变化

36.>指令和>>指令

作用：>输出重定向；>>追加

基本语法：

(1)ls -l > 文件：将列表的内容写入文件中(覆盖写)

(2)ls -al >> 文件：将列表的内容写入文件中(追加写到末尾)

(3)cat 文件1 > 文件2：将文件1的内容覆盖到文件2

(4)echo “内容” >> 文件：将内容写入文件中(追加写到末尾)

例如：

(1) ls -l /home > /home/info.txt，将home目录下的文件列表覆盖写入到info.txt中，

如果info.txt不存在，则自动创建

(2) echo “ok” >> /home/info.txt，将ok追加写入到info.txt的末尾

(3)cal >> /home/info.txt，将cal指令打印出的日历追加写入到info.txt的末尾

37.ln指令(ln中的l是小写的L)

基本语法：ln -s [原文件或目录] [软连接名]

作用：给原文件/目录创建一个软链接，其类似于windows里的快捷方式。

例如：

(1)ln -s /root/ /home/myroot，在home目录下创建一个软链接myroot，该连接指向

root目录

(2)rm /home/myroot，删除软链接myroot

38.history指令

基本语法：history

作用：查看已经执行过的历史指令，也可以指定历史指令

例如：

(1) history，显示所有历史指令

(2) history 10，显示最近的10个历史指令

(3) !382，执行历史编号为382的历史指令

39.date指令

基本语法：

(1) date，显示当前时间

(2) date +%Y，显示当前年份

(3) date +%m，显示当前月份

(4) date +%d，显示当前天数

(5) date "+%Y-%m-%d %H:%M:%S"，按格式显示当前时间

(6) date -s 字符串时间，设置系统当前时间

例如：date -s "2020-11-03 20:02:10"，设置系统时间

40.cal指令

基本语法：cal [选项]

作用：显示日历

例如：

(1)cal，显示当前日历

(2)cal 2020，显示2020年的日历

41.find指令

基本语法：find [搜索范围] [选项]

作用：从指定目录向下递归地遍历其各个子目录，将满足条件的文件或目录显示在终端

选项：

-name<文件名>：按照指定的文件名查找文件

-user<用户名>：查找属于指定用户的所有文件

-size<文件大小>：按照指定的文件大小查找文件

案例：

(1)查找/home目录下的hello.txt文件

find /home -name hello.txt

通配符可以使用：find /home -name \*.txt

(2)查找/opt目录下，属于用户名为nobody的文件

find /opt -user nobody

也可以配合管道命令：find /opt -user nobody | more

(3)查找linux系统下大于200M的文件(+n大于 -n小于 n等于，单位有k,M,G)

find / -size +200M

42.locate指令

基本语法：locate 文件名

作用：快速定位文件路径。由于该指令基于数据库进行查询，因此第一次运行前，必须

使用updatedb指令创建locate数据库

43.which指令

基本语法：which 指令名

作用：查询指令在哪个目录下

例如：

(1)查询ls指令在哪个目录下

which ls

44.grep指令

基本语法：grep [选项] 查找内容 源文件

常用选项：

-n：显示匹配行及行号

-i：忽略字母大小写

作用：过滤查找

案例：在hello.txt文件中，查找yes所在行并显示行号

cat /home/hello.txt | grep -n "yes"

或者 grep -n "yes" /home/hello.txt

补充：管道符|表示将前一个命令的处理结果输出传递给后面的命令处理

45.gzip/gunzip指令

基本语法：

(1)gzip 文件

(2)gunzip 文件.gz

作用：gzip用于压缩文件，gunzip用于解压文件

案例：

(1)压缩hello.txt：gzip /home/hello.txt

(2)解压hello.txt：gunzip /home/hello.txt.gz

补充：压缩 或者 解压后，原文件就消失了，只剩下压缩或解压后的文件

46.zip/unzip指令

基本语法：

(1)zip [选项] xxx.zip 将要压缩的内容

(2)unzip [选项] xxx.zip

作用：zip用于压缩文件，unzip用于解压文件，在项目打包发布时很有用

zip常用选项：

-r：递归压缩

unzip常用选项：

-d 存放目录：指定解压后文件的存放目录

案例：

(1)将/home自己和其下的文件及目录压缩：zip -r myhome.zip /home/

(2)将myhome.zip解压到/opt/tmp目录下：unzip -d /opt/tmp /home/myhome.zip

47.tar指令

基本语法：tar [选项] xxx.tar.gz 打包的内容

选项：

-c：产生.tar打包文件

-C：指定解压目录

-v：显示详细信息

-f：指定压缩后的文件名

-z：用gzip对存档压缩或解压

-x：解包.tar文件

作用：打包指令，产生后缀为.tar.gz的文件

案例：

(1)将/home/pig.txt和/home/cat.txt压缩成pc.tar.gz

tar -zcvf pc.tar.gz /home/pig.txt /home/cat.txt

(2)将/home的文件夹压缩成myhome.tar.gz

tar -zcvf myhome.tar.gz /home/

(3)将pc.tar.gz解压到当前目录

tar -zxvf pc.tar.gz

(4)将myhome.tar.gz解压到/opt/tem2目录下

tar -zxvf /home/myhome.tar.gz -C /opt/tem2

# linux组

1.介绍

在linux中的每个用户必须属于一个组，不能独立于组外。在linux中每个文件有所有者、

所在组、其他组的概念。一般为谁创建了文件，就自然的成为该文件的所有者，文件所

属于的组就是这个所有者所在的组。

2.查看文件的所有者和文件所在组

指令：ls -ahl。a是显示所有文件；h是显示文件大小具体；l是竖着显示详细信息。

例如：-rw-r--r--. 1 root app 0 5月 9 09:42 apple.txt。其中root是所有者，app

是所在组。

3.修改文件所有者

指令：chown 用户名 文件名

例如：chown tom apple.txt

4.创建组

指令：groupadd 组名

例如：创建一个组monster，创建一个用户fox并放入到monster组中

groupadd monster

useradd -g monster fox

5.修改文件所在组

指令：chgrp 组名 文件名

例如：chgrp app apple.txt

6.修改用户所在组

指令：

(1)usermod -g 新组名 用户名

修改用户所在组

(2)usermod -d 目录名 用户名

修改该用户登录的初始目录，前提是用户有进入该目录的权限

7.权限介绍

ls -l指令中的内容如下：

drwx------. 16 tom tom 4096 5月 1 14:13 tom

0-9位说明：

第0位确定文件类型（d,-,l,c,b）

-是普通文件

l是链接，相当于windows的快捷方式

d是目录，相当于windows的文件夹

c是字符设备文件，如鼠标，键盘

b是块设备文件，如硬盘

1～3位确定所有者拥有文件的权限

4～6位确定所在组拥有该文件的权限

7～9位确定其他用户拥有该文件的权限

其他说明：

16 ：如果是文件，表示硬连接数；如果是目录，表示子目录数量

tom：用户名

tom：组名

4096：文件大小（字节）

5月 1 14:13：最后一次的修改时间

tom：文件名

rwx权限详解：

rwx作用在文件：

r代表可读：可以读取，查看

w代表可写：可以修改，但是不代表可以删除该文件，删除一个文件的前提条

件是对该文件所在的目录有写权限，才能删除该文件

x代表可执行：可以被执行

rwx作用在目录：

r代表可读：可以读取，ls查看目录内容

w代表可写：可以修改，对目录内创建+删除+重命名目录

x代表可执行：可以进入该目录，执行目录内的文件必须可以进入该目录

补充：rwx可以用数字表示：r=4，w=2，x=1

8.修改权限

说明：chmod指令，可以修改文件或者目录权限。

方式一：+、-、=变更权限

u:所有者 g:所在组 o:其他组 a:所有人

chmod u=rwx,g=rx,o=x 文件/目录名

chmod o+w 文件/目录名

chmod a-x 文件/目录名

方式二：通过数字变更权限

r=4 w=2 x=1

chmod u=rwx,g=rx,o=x 文件目录名

相当于 chmod 751 文件目录名

9.修改文件所有者

chown 用户名 文件/目录：改变所有者

chown 用户名:组名 文件/目录：改变所有者和所在组

-R 如果是目录 则使其下的所有子文件或目录递归生效

例如：将/home/test目录下所有文件和目录的所有者都改为tom

chown -R tom /home/test

10.修改文件所在组：

chgrp 组名 文件/目录：改变所在组

-R 如果是目录，则使其下的所有子文件或目录递归生效

11.权限实战经验

(1)假设有一个目录结构/jack/tom.txt，用户A对tom.txt有读和写的权限，想修改tom.txt

中的内容，则用户A必须对jack目录有能够进入的权限，否则无法修改tom.txt的内

容。如果用户A被赋予jack目录x权限时仍然无法进入该目录，远程连接时可以重

新登录试试。

# crond任务调度

任务调度：是指系统在某个时间执行的特定的命令或程序。

分类：

(1)周而复始的系统工作，如病毒扫描等。

(2)个别用户工作，如mysql数据库的备份。

语法：crontab [选项]

常用选项：

-e:编辑crontab定时任务

-l:查询crontab任务

-r:删除当前用户所有的crontab任务

service crond restart 重启任务调度

设置任务调度规则：

\*/1 \* \* \* \* ls -l /etc/ > /tmp/to.txt：表示每小时每分钟执行一次ls -l /etc/ > /tmp/to.txt

占位符说明：

第一个‘\*’，一小时当中的第几分钟，0-59

第二个‘\*’，一天中的第几小时，0-23

第三个‘\*’，一个月当中的第几天，1-31

第四个‘\*’，一年中的第几月，1-12

第五个‘\*’，一周当中的星期几，0-7(0=7)

特殊符号说明：

\*代表任何时间，比如第一个\*就代表每分钟都执行一次

,表示不连续的时间，“0 8,12 \* \* \*”代表每天的8点0分，12点0分都执行一次命令

-代表连续的时间范围，比如0 5 \* \* 1-6代表在周一到周六的凌晨5点0分执行命令

\*/n代表每隔多久执行一次，比如“\*/10 \* \* \* \*”代表每隔十分钟就执行一次命令

案例：

(1)每隔一分钟，将当前的日期信息追加到/tmp/mydate文件中

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

(2)每隔一分钟，将当前日期和日历都追加到/home/mycal文件中

vim /home/my.sh 并写入内容 date >> /home/mycal 和 cal >> /home/mycal

给my.sh增加执行权限 chmod u+x /home/my.sh

crontab -e 并写入 \*/1 \* \* \* \* /home/my.sh

(3)每天凌晨2:00将mysql数据库testdb备份到文件中

crontab -e 并写入 0 2 \* \* \* mysqldump -uroot -p123456 testdb > /home/db.bak

# at定时任务

基本介绍：

1.at命令是一次性定时计划任务，at的守护进程atd会以后台模式运行，检查作业队列

的运行。

2.默认情况下，atd守护进程每60秒检查作业队列，有作业时，会检查作业队列运行时间，

如果时间与当前时间匹配，则运行此作业。

3.at命令是一次性定时计划任务，执行完一个任务后不再执行此任务了。

4.使用at命令时，一定要保证atd进程的启动，可以使用相关指令来查看。

ps -ef 可以查看当前所有运行的进程

ps -ef | grep atd 检查atd是否在运行

语法：at [选项] [时间]

Ctrl + D 结束at命令的输入

atq：查看系统中有没有执行的工作任务

atrm 编号：删除对应编号的工作任务

常用选项：

-m 当指定的任务被完成后，将给用户发送邮件，即使没有标准输出

-I(大i) atq的别名

-d atrm的别名

-v 显示任务被执行的时间

-c 打印任务的内容到标准输出

-V显示版本信息

-q <队列> 使用指定的队列

-f <文件> 从指定文件读入任务而不是从标准输入读入

-t <时间参数> 以时间参数的形式提交要运行的任务

at时间定义：

1.接受在当天hh:mm（小时:分钟）式的时间指定，假如时间已经过去，那么就放在

第二天执行。例如：04:00

2.使用midnight（深夜），noon（中午），teatime（一般是下午4点）等比较模糊的词语来指

定时间

3.采用12小时制，即在时间后面加上AM或PM来说明是上午还是下午。例如：12pm

4.指定命令执行的具体日期，指定格式为month day（月 日）或mm/dd/yy（月/日/年）或

dd.mm.yy（日.月.年），指定日期必须在指定时间的后面。例如：04:00 2021-13-1

5.使用相对计时法，指定格式为：now + count time-units，now就是当前时间，time-units是

时间单位，这里能够是minutes，hours，days，weeks，count是时间的数量，几天，几小时。例如：now + 5 minutes

6.直接使用today，tomorrow来指定完成命令的时间

# linux分区

原理介绍：

1.Linux无论有几个分区，分给哪个目录使用，它归根结底就只有一个根目录，一个独立且

唯一的文件结构，Linux中每个分区都是用来组成整个文件系统的一部分。

2.Linux采用了一种叫“载入”的处理方法，它的整个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使它的存储空间在一个目录下获得。

lsblk 或 lsblk -f 查看所有设备的挂载情况

硬盘说明：

1.Linux硬盘分IDE硬盘和SCSI硬盘，目前基本上是SCSI硬盘

2.对于IDE硬盘，驱动器标识符为“hd~”，其中“hd”表明分区所在设备的类型，这里是指

IDE硬盘。“x”为盘号（a为基本盘，b为基本从属盘，c为辅助主盘，d为辅助从属盘），“~”代表分区，前四个分区用数字1-4表示，它们是主分区或扩展分区，从5开始就是逻辑分区。例：hda3表示为第一个IDE硬盘上的第三个主分区或扩展区，hdb2表示为第二个IDE硬盘上的第二个主分区或扩展分区。

3.对于SCSI硬盘则标识为“sdx~”，SCSI硬盘是用“sd”来表示分区所在设备的类型的，其

余则和IDE硬盘的表示方法一样

增加磁盘实例：

1. 右击左边列表中你的虚拟机名称->设置->添加->选择硬盘->选择SCSI(S)（推荐）->创建新

虚拟磁盘(V)->填写磁盘大小->点击完成，然后重启系统。如果重启后要你安装系统，是

因为虚拟机读取磁盘顺序不对，要在BIOS界面调整。

2.分区命令 fdisk /dev/sdb

开始对/sdb分区

m 显示命令列表

p 显示磁盘分区 同 fdisk -l

n 新增分区

d 删除分区

w 写入并退出

q 不保存退出

说明：开始分区后输入n，新增分区，然后选择p，分区类型为主分区，两次回车默认

剩余全部空间，最后输入w写入分区并退出，若不保存退出输入q。

3.格式化磁盘

mkfs -t ext4 /dev/sdb1

其中ext4是分区类型

4.挂载：将一个分区与一个目录联系起来

mount 设备名称 挂载目录：挂载

例如：mount /dev/sdb1 /newdisk

umount 设备名称 或 挂载目录：消除挂载

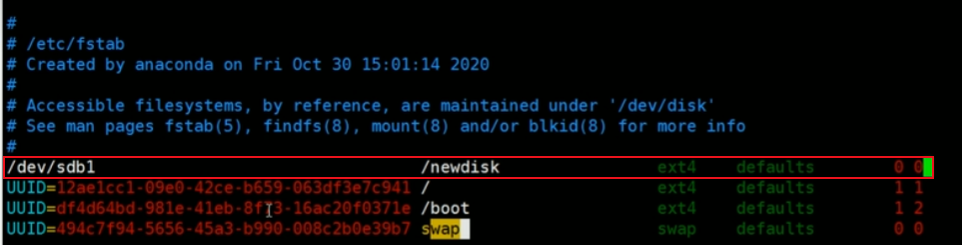
例如：umount /dev/sdb1 或 umount /newdisk

ps:用命令行挂载，重启后会失效

5.永久挂载

修改/etc/fstab实现挂载

添加完成后 执行mount -a 即刻生效



红框中的内容为示例修改，最右侧的0 0，第一个0表示不做dump备份，第二个0表示不

检查硬盘扇区。

# 磁盘

1.查询系统整体磁盘使用情况：df -h

2.查询指定目录的磁盘占用情况：du -h /目录

常见选项：

-s 指定目录占用大小汇总

-h 带计量单位

-a 含文件

--max-depth=1 子目录深度

-c 列出明细的同时，增加汇总值

例如：查询/opt目录的磁盘占用情况，深度为1

du -h --max-depth=1 /opt

3.磁盘情况-工作使用指令

(1)统计/opt目录下文件的数量

ls -l /opt | grep "^-" | wc -l

(2)统计/opt目录下目录的数量

ls -l /opt | grep "^d" | wc -l

(3)统计/opt文件夹下文件的个数，包括子文件夹里的

ls -lR /opt | grep "^-" | wc -l

(4)统计/opt文件夹下目录的个数，包括子文件夹里的

ls -lR /opt | grep "^d" | wc -l

(5)以树状显示目录结构

tree 目录

ps：默认tree未安装，yum install tree

# 网络

1.自动获取ip（默认）

查看位置：应用程序->系统工具->设置->网络->点击 有线连接右下方的齿轮->点击 Ipv4

2.指定ip

说明：直接修改配置文件（/etc/sysconfig/network-scripts/ifcfg-ens33）来指定ip，并可以连

接到外网。

ifcfg-ens33文件说明：

DEVICE=eth0 #接口名（设备，网卡）

HWADDR=00:0c:2x:6x:0x:xx #MAC地址

TYPE=Ethernet #网络类型（通常是Ethenet）

UUID= #随机ID

#系统启动的时候网络接口是否有效(yes/no)

ONBOOT=yes

ifcfg-ens33文件需要修改或添加的内容：

#IP的配置方法[none|static|bootp|dhcp](引导时不使用协议|静态分配IP|BOOTP协议

|DHCP协议)

BOOTPROTO=static

#IP地址

IPADDR=192.168.200.130

#网关

GATEWAY=192.168.200.2

#域名解析器

DNS1=192.168.200.2

重启网络服务或者重启系统生效

service network restart、reboot

补充：查看网关 ip route show

3.设置主机名和hosts映射

(1)查看主机名：hostname

(2)修改主机名，需要修改文件/etc/hostname中的内容

(3)重启生效

(4)设置hosts映射，也就是通过主机名能找到（如ping）某个linux系统

a.如果是windows，在C:\Windows\System32\drivers\etc\hosts文件中指定即可。

如：192.168.200.130 hspedu100

b.而在linux中，在/etc/hosts文件中指定

如：192.168.200.1 hspedu100

4.主机名解析过程

(1)hosts是一个文本文件，用来记录IP和Hostname（主机名）的映射关系

(2)DNS是互联网上作为域名和IP地址互相映射的一个分布式数据库

(3)当用户在浏览器输入了[www.baidu.com](http://www.baidu.com)并查询时：

a.浏览器先检查浏览器缓存中有没有该域名解析IP地址，有就完成解析。如果没有，就

检查DNS解析器缓存，如果有就完成解析。这两个缓存可以理解为本地解析器缓存。

b.一般来说，当电脑第一次成功访问某一网站后，在一定时间内，浏览器或操作系统会

缓存网站的IP地址（DNS解析记录）

ipconfig /displaydns //DNS域名解析缓存

ipconfig /flushdns //手动清理dns缓存

c.如果本地解析器缓存没有找到对应映射，则检查系统中hosts文件中有无配置对应的

域名IP映射，如果有则完成解析并返回。

d.如果本地解析器缓存和系统中hosts文件都没有找到对应的IP，则到域名服务DNS进

行解析。

# 进程管理

1.基本介绍

(1)在linux中，每个执行的程序都被称为一个进程，每一个进程都分配一个ID号（pid,进程

号）

(2)每一进程都可能以两种方式存在的。前台与后台，所谓前台进程就是用户目前的屏幕上

可以进行操作的。后台进程则是实际在操作，但由于屏幕上无法看到的进程，通常使用后台方式执行

(3)一般系统的服务都是以后台进程的方式存在，而且都会常驻在系统中，直到关机才结束

2.进程指令

ps命令是用来查看目前系统中，有哪些正在执行，以及他们执行的状况。可以不加任何参

数。

ps显示的信息选项：

PID 进程识别号

TTY 终端机号

TIME 此进程所消耗CPU时间

CMD 正在执行的名称或进程名

常用参数

-a:显示当前终端的所有进程信息

-u:以用户的格式显示进程信息

-x:显示后台进程运行的参数

-e:显示所有进程

-l:长格式显示更加详细的信息

-f:全格式，以全格式显示当前所有进程，可以查看进程的父进程

常用组合：

ps -aux | more 可以查看系统中所有的进程

ps -aux | grep xxx 查看是否有xxx服务

ps -le 可以查看系统中所有的进程，而且还能看到进程的父进程的 PID 和进程优先级

ps -ef 同le相似

ps -l 只能看到当前 Shell 产生的进程

ps -aux | more 显示的信息选项：

USER:用户名

PID:进程识别号

%CPU:进程占用CPU的百分比

%MEM:进程占用物理内存的百分比

VSZ:进程占用的虚拟内存大小（KB）

RSS:进程占用的物理内存大小（KB）

TTY:终端机号

STAT:运行状态，S-休眠，s-该进程是会话的先导进程，N-进程拥有比普通优先级更低的

优先级，R-正在运行，D-短期等待，Z-僵死进程，T-被跟踪或者被停止等等

START:进程的启动时间

TIME:此进程所消耗CPU时间

COMMAND:正在执行的命令和参数，过长会截断显示

ps -ef | more：以全格式显示当前所有进程。信息选项：

UID:用户ID

PID:进程ID

PPID:父进程ID

C:CPU用于计算执行优先级的因子。数值越大，表示进程是CPU密集型运算，执行优先

级会降低；数值越小，表明进程是I/O密集型运算，执行优先级会提高

STIME:进程的启动时间

TTY:终端机号

TIME:此进程所消耗CPU时间

CMD:正在执行的命令和参数

3.终止进程

kill [选项] 进程号（通过进程号杀死进程）

killall 进程名称（通过进程名杀死进程，也支持通配符，这在系统因负载过大而变得很慢时

很有用）

常用选项：

-9：表示强迫进程立即停止

重启sshd（远程SSL登录服务）：systemctl start sshd

4.查看进程树：pstree [选项]（可以更加直观的查看进程信息）

常用选项：

-p:显示进程

-u:显示进程的所属用户

# 服务管理

1.基本介绍

服务本质就是进程，但是是运行在后台的，通常都会监听某个端口，等待其他程序的请求，

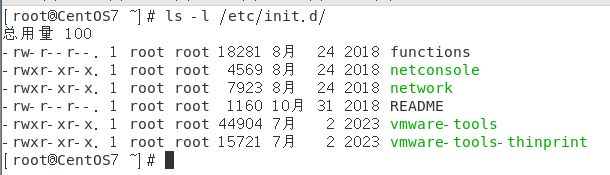
比如mysql、sshd、防火墙等，因此又称守护进程。

2.service管理指令

(1)service 服务名 [start|stop|restart|reload|status]

(2)在CentOS7.0后，很多服务不再使用service，而是systemctl

(3)service指令管理的服务在/etc/init.d查看



补充：如图所示，其中绿色的表示仍然使用service指令。

案例：停止，然后重新打开网络服务

service network stop

service network start

3.查看服务的方式

(1)使用指令 ls -l /etc/init.d/ 能够看到简单的一些服务

(2)使用指令setup可以看到全部服务。

步骤：

a.输入setup然后回车

b.选择系统服务，然后使用tab键选择运行工具，然后回车，此时展示的就是全部

服务

c.左边带\*号的服务表示随系统启动而自动启动，可以使用空格键将\*号去掉，这样

就不会再随系统而自动启动了，最后使用tab键确定或取消

4.开机流程说明

开机 => BIOS => /boot => systemd进程1 => 运行级别 => 运行级别对应的服务

5.chkconfig指令

介绍：通过chkconfig可以给服务的各个运行级别设置自启动/关闭。

范围：chkconfig能够管理的服务在/etc/init.d查看

语法：

(1) chkconfig --list [| grep 服务名] 查看服务在7个运行级别下自启动/关闭状态

(2) chkconfig 服务名 --list 查看服务在7个运行级别下自启动/关闭状态

(3) chkconfig --level 5 服务名 on/off 设置某个服务在某个运行级别下的自启动/关闭

状态，这里示范的运行级别是5

示例：对network服务在运行级别3下关闭自启动

chkconfig --level 3 network off

补充：修改自启动/关闭后，需要重启机器生效，可以使用reboot指令重启。

6.systemctl指令

基本语法：systemctl [start|stop|restart|status] 服务名

重启网络：systemctl restart network

范围：systemctl能够管理的服务在/usr/lib/systemd/system中查看

systemctl设置服务的自启动状态：

systemctl list-unit-files [| grep 服务名] （查看服务开机启动状态）

systemctl enable 服务名 （设置服务开机启动）

systemctl disable 服务名 （关闭服务开机启动）

systemctl is-enabled 服务名 （查询某个服务是否是自启动的）

补充：CentOS7.0后，以上设置控制的是3,5级别下开机是否自启动

案例：

(1)查看防火墙自启动状态

systemctl list-unit-files | grep firewalld 或者 systemctl is-enabled firewalld

(2)查看防火墙状态

systemctl status firewalld

(3)关闭防火墙

systemctl stop firewalld

(4)重启防火墙

systemctl start firewalld

补充：案例中的关闭/重启防火墙后会立即生效，但这只是临时生效，重启系统后，还是会

回归默认的设置。如果想永久生效，要使用指令systemctl enable/disable 服务名

7. firewall指令

作用：打开/关闭端口

语法：

firewall-cmd --permanent --add-port=端口/协议 打开端口

firewall-cmd --permanent --remove-port=端口/协议 关闭端口

firewall-cmd --reload 重新载入

打开或关闭端口后，必须重新载入，否则不生效

firewall-cmd --query-port=端口/协议 查询端口是否开放

firewall-cmd --list-all 查询开放端口

补充：

windows的telnet指令可用于测试某个ip下的某个端口是否开放，如果没安装telnet

可以按以下步骤安装。

(1)按 Windows + R 键，在弹出的运行对话框中输入 control，点击回车，打开控制面板。

(2)在控制面板中，选择 程序，点击 程序和功能。

(3)在左侧菜单中，点击 启用或关闭Windows功能。

(4)在弹出的窗口中，向下滚动，找到Telnet客户端，勾选该选项，点击“确定”进行

安装。

(5)系统会自动下载并安装Telnet客户端，安装完成后，即可在命令行中使用 telnet 命

令。

例子：使用windows的telnet指令测试linux的111端口是否开放

telnet 192.168.231.128 111

如果没提示连接失败则表示111端口是开放的

8. 动态监控进程

介绍：top和ps命令相似。他们都用来显示正在执行的进程。但是top在执行一段时间可以

更新正在运行的进程。

语法：top [选项]

常用选项：

-d 秒数：指定top命令每隔几秒更新，默认是3秒

例：top -d 5 每隔5秒刷新

-i:使top不显示任何限制或者僵死进程

-p:通过指定监控进程ID来仅仅监控某个进程的状态

动态监控交互操作：

P:以CPU使用率排序，默认即使此项

M:以内存的使用率排序

N:以PID排序

q:退出top

u:监控指定用户

输入u后，再输入用户名，然后回车

k:终止指定进程

输入k后，再输入进程的pid号，然后回车，此时需要输入信号，一般15即可，

如果仍然无法终止进程，就把信号重新改为9（强制），然后回车即可。

9.网络监控

语法：netstat [选项]

选项说明：

-an 按一定顺序排列输出

-p 显示哪个进程在调用

一般使用：netstat -anp | more 更方便输出详细信息

案例：查看服务名为sshd的服务信息

netstat -anp | grep sshd

补充：windows和linux都可以使用ping ip地址 的方式来检测本机对某个ip是否网络通畅。

在linux下使用此命令后，可以用Ctrl+C的方式退出。

# rpm

1.介绍

rpm用于互联网下载包的打包及安装工具，它包含在某些Linux分发版中。它生成具有.RPM

扩展名的文件。RPM是RedHat Package Manager（RedHat软件包管理工具）的缩写，类

似windows的setup.exe，这一文件格式名称虽然打上了RedHat的标签，但理念是通用

的。

2.根据名称查询已安装的rpm：rpm -qa | grep xx

查询已安装的火狐浏览器：rpm -qa | grep firefox

3.rpm包名基本格式

以rpm -qa | grep firefox指令的输出为例：firefox-60.2.2-1.el7.centos.x86\_64

名称：firefox

版本号：60.2.2-1

适用操作系统：el7.centos.x86\_64，表示centos7.x的64位系统。

如果是i686,i386表示32位系统，noarch表示通用

4.详细指令

(1)查询所有安装的rpm软件包：

rpm -qa 或 rpm -qa | more

(2)查询某个安装的rpm软件包

rpm -q xx 或 rpm -qa | grep xx

(3)查询软件包信息：

rpm -qi xx

(4)查询软件包中的文件：

rpm -ql xx

(5)查询文件所属软件包：

rpm -qf 文件全路径

例：rpm -qf /root/install.log

(6)卸载rpm包：

rpm -e xx

补充：如果删除的软件包被其他软件包所依赖，卸载是会产生错误信息，如果确定

要卸载，增加参数--nodeps，但是不推荐。

例：rpm -e --nodeps foo

(7)安装rpm包：

rpm -ivh rpm包全路径名称

参数说明：

i：install 安装

v：verbose 提示

h：hash 进度条

5.yum

介绍：yum是一个shell前端软件包管理器。基于rpm包管理，能够从指定的服务器自动下

载rpm包并且安装，可以自动处理依赖关系，并且一次安装所有依赖的软件包。

查询yum服务器是否有需要安装的软件：

yum list | grep xx

安装指定的yum包：

yum install xx

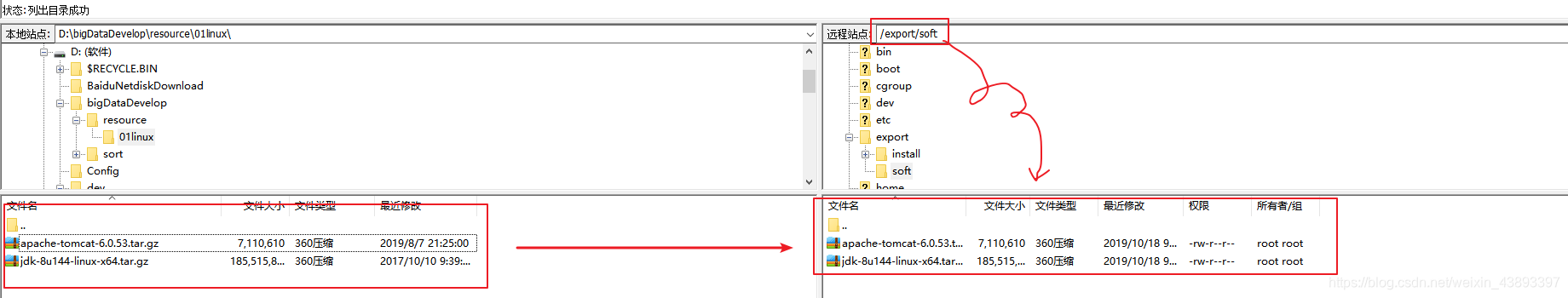
# 安装jdk1.8

1.创建两个文件夹

创建sort mkdir -p /export/sort (此目录用来存放安装包)

创建install mkdir -p /export/install

2.使用工具上传至Linux服务器



3.解压到install文件夹中

tar -zxvf /export/sort/jdk-8u144-linux-x64.tar.gz -C /export/install/

4.配置环境变量

(1)编辑文件

vim /etc/profile.d/java.sh

(2)将内容复制到文件中

export JAVA\_HOME=/export/install/jdk1.8.0\_144

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

(3)检查是否配置完成

输入jps，显示未成功！

在这里插入图片描述

(4)更新profile文件

source /etc/profile

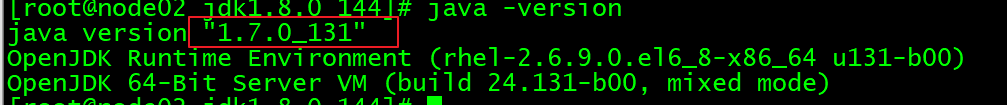
(5)重新检查

如果出现数字表示安装成功!

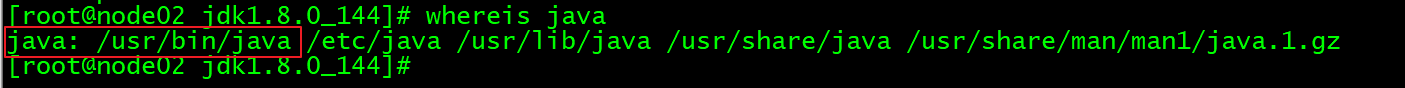
在这里插入图片描述

5.切换版本（本地已有jdk时操作，没有请忽略！）

先查看一下版本：java -version



whereis java



查看这个目录：ll /usr/bin/java

在这里插入图片描述

你会发现是一个软链接(类似于Windows的快捷方式)，再查看一下：ll /etc/alternatives/java

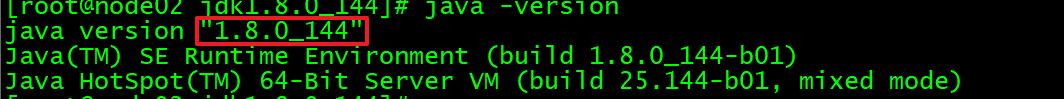
在这里插入图片描述

你会看到指向的jdk1.7的bin里的java，我们将这个软连接删除掉 再重新创建一个新的

rm -rf /etc/alternatives/java

ln -s /export/install/jdk1.8.0\_144/bin/java /etc/alternatives/java

再来查看一下版本：java -version



此时成功切换版本！

# 安装tomcat

1.下载linux版本的tomcat

下载地址：<http://archive.apache.org/dist/tomcat/tomcat-8/v8.5.59/bin/>

注意：是下载 apache-tomcat-8.5.59.tar.gz

2.创建两个文件夹（如果没有）

创建sort mkdir -p /export/sort (此目录用来存放安装包)

创建install mkdir -p /export/install

3.解压到install文件夹中

tar -zxvf /export/sort/apache-tomcat-8.5.59.tar.gz -C /export/install/

4.启动tomcat

(1)进入tomcat的bin目录

cd /export/install/apache-tomcat-8.5.59/bin

(2)使用命令startup

./startup.sh

(3)开放端口8080并重新加载

firewall-cmd --permanent --add-port=8080/tcp

firewall-cmd --reload

(4)查询8080端口是否已开放

firewall-cmd --query-port=8080/tcp

(5)校验

使用外部windows的浏览器访问<http://192.168.231.128:8080/>，如果显示tomcat页面，

则表示成功启动。192.168.231.128是linux的ip地址，可以使用ifconfig指令查看。

(6)测试html文件

cd /export/install/apache-tomcat-8.5.59/webapps/ROOT/

vim hsp.html

输入<h1>hspedu.ok</h1>并保存退出

使用外部windows的浏览器访问http://192.168.231.128:8080/hsp.html

# 安装mysql

1.创建三个文件夹（如果没有）

创建sort mkdir -p /export/sort (此目录用来存放安装包)

创建install mkdir -p /export/install

创建mysql5.7 mkdir -p /export/install/mysql5.7

2.下载linux版本的mysql5.7

(1)首先进入/export/sort：cd /export/sort

(2)使用命令直接下载mysql5.7：

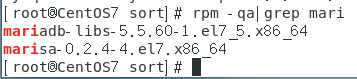
wget http://dev.mysql.com/get/mysql-5.7.26-1.el7.x86\_64.rpm-bundle.tar

3.解压到mysql5.7文件夹中

tar -xvf /export/sort/mysql-5.7.26-1.el7.x86\_64.rpm-bundle.tar -C /export/install/mysql5.7/

4.centos7.6自带的数据库是mariadb，会跟mysql冲突，要先删除

(1)使用rpm -qa|grep mari查询mariadb



(2)卸载mariadb

rpm -e --nodeps mariadb-libs

5.开始安装mysql

cd /export/install/mysql5.7

rpm -ivh mysql-community-common-5.7.26-1.el7.x86\_64.rpm

rpm -ivh mysql-community-libs-5.7.26-1.el7.x86\_64.rpm

rpm -ivh mysql-community-client-5.7.26-1.el7.x86\_64.rpm

rpm -ivh mysql-community-server-5.7.26-1.el7.x86\_64.rpm

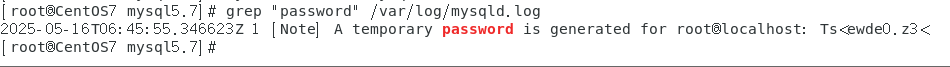
6.启动mysql

systemctl start mysqld.service

7.设置root用户密码

(1)查询随机初始密码

grep "password" /var/log/mysqld.log



如图所示，初始密码是Ts<ewde0.z3<

(2)使用初始密码登录

mysql -uroot -p

然后输入Ts<ewde0.z3< 并回车

(3)设置密码策略

set global validate\_password\_policy=0 ，然后回车，然后输入;并回车

注意：mysql的密码策略有3种，分别是0，1，2。其中0是最简单的，实际开发中可选1

或2，默认是1。

(4)修改root用户的密码为li123456

set password for 'root'@'localhost' =password('li123456');

(5)刷新

flush privileges;

8.退出并使用新密码登录

quit

mysql -uroot -p

然后输入li123456并回车，成功登录

# shell编程

1.介绍

Shell是一个命令行解释器，它为用户提供了一个向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。

2.格式要求

(1) 脚本以#!/bin/bash开头

(2) 脚本需要有可执行权限

3.执行方式

(1) 赋予脚本x权限，直接使用路径执行

(2) 使用 sh + 脚本执行，这种方式不用赋予脚本x权限

4.简单案例

(1)创建shell文件

mkdir /root/shellcode

cd /root/shellcode

vim hello.sh

输入

#!/bin/bash

echo "hello,world~"

并保存退出

(2)增加hello.sh的执行权限

chmod u+x hello.sh

(3)执行hello.sh

./hello.sh

或

/root/shellcode/hello.sh

或

sh hello.sh（这种方式不需要有hello.sh的执行权限）

或

sh /root/shellcode/hello.sh（这种方式不需要有hello.sh的执行权限）

5.Shell变量

介绍：Linux Shell中变量分为系统变量和用户自定义变量

系统变量：$HOME、$PWD等。例：echo $HOME 可以输出HOME表示的值。

显示当前Shell中所有的变量：set|more

shell变量的定义：

定义变量：变量=值

撤销变量：unset 变量

声明静态变量：readonly 变量，注意：不能unset

案例：

(1)创建shell文件

vim var.sh

(2)在var.sh中输入内容并保存退出

#!/bin/bash

#案例1：定义变量A

A=100

#输出变量需要加上$

echo $A

echo A=$A

echo "A=$A"

#案例2：撤销变量A

unset A

echo "A=$A"

#案例3：声名静态变量B=2，无法unset此变量

readonly B=2

echo "B=$B"

(3)赋予执行权限

chmod u+x var.sh

(4)执行

./var.sh

定义变量的规则：

变量名称可以由字母、数组、下划线组成，但是不能以数字开头

等号两侧不能有空格

变量名称一般习惯大写

将命令的返回值赋给变量：

A=`date`反引号，运行命令，并把结果返回给变量A

A=$(date)等价于反引号

6.环境变量(/etc/profile)

语法：

export 变量名=变量值（将shell变量输出为环境变量/全局变量）

环境变量/全局变量可以被多个shell文件共享

source 配置文件（让修改后的配置文件立即生效）

echo $变量名（查询环境变量的值）

补充：shell脚本多行注释 :<<!内容!

简单案例：在/etc/profile文件中定义TOMCAT\_HOME环境变量

(1)vim /etc/profile，输入：

#定义一个环境变量

export TOMCAT\_HOME=/export/install/apache-tomcat-8.5.59

保存并退出

(2)source /etc/profile（刷新配置）

(3)echo $TOMCAT\_HOME（输出TOMCAT\_HOME表示的值）

在其他的shell文件中，只需要用$TOMCAT\_HOME便可以引用其值。

7.位置参数变量

介绍：当我们执行一个shell脚本时，如果希望获取到命令行的参数信息，就可以使用到位

置参数变量。比如：./myshell.sh 100 200，这个就是一个执行shell的命令行，可以在

myshell脚本中获取到参数信息。

基本语法：

$n(n为数字，$0代表命令本身，$1~$9代表第一个到第九个参数，十以上的参数需要

用大括号包含，如${10})

$\*(这个变量代表命令行中的所有参数，$\*把所有的参数看成一个整体)

$@(这个变量也代表命令行中的所有参数，不过$@把每个参数区分对待)

$#(这个变量代表命令行中所有参数的个数)

补充：$\*和$@不加引号时，二者作用相同。加了引号，就有整体和个体的区分。

案例：编写一个shell脚本position.sh，在脚本中获取到命令行的各个参数信息

vim position.sh

然后输入以下内容：

#!/bin/bash

echo "0=$0 1=$1 2=$2"

echo "所有的参数=$\*"

echo "$@"

echo "参数的个数=$#"

然后保存退出

chmod u+x position.sh

./position.sh 100 200

输出结果为：

0=./position.sh 1=100 2=200

所有的参数=100 200

100 200

参数的个数=2

8.预定义变量

介绍：就是shell设计者事先定义好的变量，可以直接在shell脚本中使用。

基本语法：

$$(当前进程的进程号(PID))

$!(后台运行的最后一个进程的进程号(PID))

$?(最后一次执行的命令的返回状态。如果这个变量的值为0，证明上一个命令正确执行；

如果这个变量的值为非0，则证明上一个命令执行不正确了)

案例：在一个shell脚本中简单使用下预定义变量

vim preVar.sh

然后输入以下内容：

#!/bin/bash

echo "当前执行的进程id=$$"

#以后台的方式运行一个脚本，并获取它的进程号

/root/shcode/position.sh &

echo "最后一个后台方式运行的进程id=$!"

echo "执行的结果=$?"

然后保存退出

chmod u+x preVar.sh

./preVar.sh

输出结果为：

当前执行的进程id=16392

最后一个后台方式运行的进程id=16393

执行的结果=0

[root@hspEdu100 shcode]# 0=/root/shcode/position.sh 1= 2=

所有的参数=

参数的个数=0

补充：后台运行脚本的方式：执行命令 &

9.运算符

基本语法：

$((运算式))或$[运算式]或expr m + n

注意expr运算符间要有空格，如果希望将expr 的结果赋给某个变量，使用``(反引号)

expr m - n

expr \\*,/,% 乘,除,取余

案例：

(1)计算(2+3)X4的值

#!/bin/bash

#第一种方式

RES1=$(((2+3)\*4))

echo "res1=$RES1"

#第二种方式

RES2=$[(2+3)\*4]

echo "res2=$RES2"

#第三种方式，注意加空格，如果写成了2+3，那么输出时会把+看作连接符而不是加号

TEMP=`expr 2 + 3`

RES4=`expr $TEMP \\* 4`

echo "res4=$ RES4"

(2)求出命令行的两个参数[整数]的和

#!/bin/bash

SUM=$[$1+$2]

echo "sum=$SUM"

调用时加上参数即可，例如：./oper.sh 20 50

10.条件判断

基本语法：

[ condition ] (condition前后要有空格)

非空返回true,可以使用$?验证(0为true，>1为false)

案例：

[ hsp ] 返回true

[ ] 返回false

[ conditon ] && echo OK || echo notok 条件满足时执行后面的语句

常用判断条件：

= 字符串比较

两个整数比较：

-lt 小于

-le 小于等于

-eq 等于

-gt 大于

-ge 大于等于

-ne 不等于

按照文件权限进行判断：

-r 有读的权限

-w 有写的权限

-x 有执行的权限

按照文件类型判断：

-f 文件存在并且是一个常规文件

-e 文件存在

-d 文件存在并且是一个目录

案例：

(1)“ok”是否等于“ok”

#!/bin/bash

if [ "ok" = "ok" ]

then

echo "equal"

fi

(2)23是否大于等于22

#!/bin/bash

if [ 23 -ge 22 ]

then

echo "大于"

fi

(3)/root/shcode/aaa.txt目录中的文件是否存在

#!/bin/bash

if [ -f /root/shcode/aaa.txt ]

then

echo "存在"

fi

11.流程控制

(1)if语句：

if [ 条件判断式 ]

then

程序

fi

或者

if [ 条件判断式 ]

then

程序

elif [ 条件判断式 ]

then

程序

fi

案例：编写一个shell，如果输入的参数大于等于60则输出“及格了”，如果小于60，输出“不

及格”

#!/bin/bash

if [ $1 -ge 60 ]

then

echo "及格了"

elif [ $1 -lt 60 ]

then

echo "不及格"

fi

(2)case语句：

case $变量名 in

"值1")

如果变量值等于值1，则执行程序1

;;

"值2")

如果变量值等于值2，则执行程序2

;;

...省略其他分支...

\*)

如果变量值都不是以上值，则执行此程序

;;

esac

案例：当命令行参数是1时，输出“周一”，是2时输出“周二”，其他情况输出“other”

#!/bin/bash

case $1 in

"1")

echo "周一"

;;

"2")

echo "周二"

;;

\*)

echo "other"

;;

esac

(3)for循环：

语法一：

for 变量 in 值1 值2 值3 ...

do

程序

done

语法二：

for((初始值;循环控制条件;变量变化))

do

程序

done

案例：打印命令行输入的参数

#!/bin/bash

for i in "$\*"

do

echo "num is $i"

done

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

for j in "$@"

do

echo "num is $j"

done

从1加到100的值输出显示

#!/bin/bash

SUM=0

for(( i=1; i<=100; i++ ))

do

SUM=$[$SUM+$i]

done

echo "SUM=$SUM"

补充：100可以用$1来替换，这样在执行shell时，加上参数100即可，这样更灵活。

(4)while循环

while [ 条件判断式 ]

do

程序

done

案例：从命令行输入一个数n，统计从1+...+n的值是多少

#!/bin/bash

SUM=0

i=0

while [ $i -le $1 ]

do

SUM=$[$SUM+$i]

i=$[$i+1]

done

echo "结果=$SUM"

12.read读取控制台输入

基本语法：

read (选项) (参数)

选项：

-p:指定读取值时的提示符

-t:指定读取值时等待的时间（秒），如果没有在指定时间内输入，就不再等待。

参数：

变量：指定读取值时的变量名

案例：

(1)读取控制台输入一个NUM1值

#!/bin/bash

read -p "请输入一个数NUM1=" NUM1

echo "输入的NUM1=$NUM1"

(2)读取控制台输入一个NUM1值，在10s内输入

#!/bin/bash

read -t 10 -p "请输入一个数NUM2=" NUM2

echo "输入的NUM2=$NUM2"

13.系统函数

介绍：shell编程和其他编程语言一样，有系统函数，也可以自定义函数。

系统函数：

(1) basename

语法：

basename [pathname] [suffix] 返回完整路径最后/的部分，常用于获取文件名（不带/）。

suffix为后缀，如果指定了suffix，则pathname中的suffix会去掉

案例：返回/home/aaa/test.txt的”text.txt”部分

basename /home/aaa/test.txt 输出结果：test.txt

basename /home/aaa/test.txt .txt 输出结果：test

(2) dirname

语法：

dirname 文件绝对路径 返回完整路径最后/的前面部分，常用于返回路径部分。

案例：返回/home/aaa/test.txt的/home/aaa

dirname /home/aaa/test.txt

14.自定义函数

语法：

[ function ] funname[()]

{

Action;

[return int;]

}

调用时直接写函数名：function [值]

案例：创建函数计算输入两个参数的和

#!/bin/bash

function getSum() {

SUM=$[$n1+$n2]

echo "和是=$SUM"

}

read -p "请输入一个数n1=" n1

read -p "请输入一个数n2=" n2

getSum $n1 $n2

15.使用shell定时备份mysql数据库

要求：

(1)每天凌晨2:30备份（使用crond来调度）数据库hspedu到/data/backup/db；

(2)备份开始和备份结束能够给出相应的提示信息；

(3)备份后的文件要求以备份时间为文件名，并打包成.tar.gz的形式，比如：

2021-03-12\_230201.tar.gz；

(4)在备份的同时，检查是否有10天前备份的数据库文件，如果有就将其删除；

步骤：

(1)在/usr/sbin目录下创建mysql\_db\_backup.sh

vim mysql\_db\_backup.sh

(2)填写内容

#!/bin/bash

#备份目录

BACKUP=/data/backup/db

#获取当前时间

DATETIME=$(date +%Y-%m-%d\_%H%M%S)

echo $DATETIME

#数据库的地址

HOST=localhost

#数据库用户名

DB\_USER=root

#数据库密码

DB\_PW=li123456

#备份的数据库名

DATABASE=hspedu

#创建备份目录，如果不存在就创建

[ ! -d "${BACKUP}/${DATETIME}" ] && mkdir -p "${BACKUP}/${DATETIME}"

#备份数据库

mysqldump -u${DB\_USER} -p${DB\_PW} --host=${HOST} -q -R --databases ${DATABASE} | gzip > ${BACKUP}/${DATETIME}/$DATETIME.sql.gz

#将文件处理成tar.gz

cd ${BACKUP}

tar -zcvf $DATETIME.tar.gz ${DATETIME}

#删除对应的备份目录

rm -rf ${BACKUP}/${DATETIME}

#删除10天前的备份文件

find ${BACKUP} -atime +10 -name "\*.tar.gz" -exec rm -rf {} \;

echo "备份数据库 ${DATETIME} 成功-"

(3)赋予执行权限

chmod u+x mysql\_db\_backup.sh

(4)设置定时

crontab -e

然后输入：

30 2 \* \* \* /usr/sbin/mysql\_db\_backup.sh

# Ubuntu

1.下载地址

<https://cn.ubuntu.com/download/>

说明：Centos停止更新后，Ubuntu LTS（长期支持版本，一般为5年）作为其可替代的选择，

优点是完全免费。当然，Rocky Linux、AlmaLinux 这二者也是可以考虑的选择。

2.安装

视频地址：

<https://www.bilibili.com/video/BV1Sv411r7vd?spm_id_from=333.788.player.switch&vd_source=11e4dce1b79b451398ab2c230943e3e6&p=108>

本地：E:\video\linux\Ubuntn安装.mp4

3.设置中文

视频地址：

https://www.bilibili.com/video/BV1Sv411r7vd?spm\_id\_from=333.788.videopod.episodes&vd\_source=11e4dce1b79b451398ab2c230943e3e6&p=109

本地：E:\video\linux\Ubuntn设置中文.mp4

4.设置root用户初始密码

视频地址：

https://www.bilibili.com/video/BV1Sv411r7vd?spm\_id\_from=333.788.videopod.episodes&vd\_source=11e4dce1b79b451398ab2c230943e3e6&p=110

本地：E:\video\linux\Ubuntn设置root用户初始密码.mp4

5.SSH

SSH用于远程登录使用。Ubuntu不会默认安装SSH，或者只默认安装SSH的客户端，没有安

装SSH的服务端，比如24版本，此时需要手动安装SSH。

(1)检查 SSH 服务状态

sudo systemctl status ssh

注意：如果是centos，那么命令是sudo systemctl status sshd

如果显示 active (running)，说明 SSH 服务器已安装并运行。

如果提示 Unit ssh.service could not be found，则表示未安装。

(2)检查 SSH 客户端

ssh -V

如果显示版本信息（如 OpenSSH\_9.6p1），说明客户端已安装。

(3)如果需要安装 SSH 服务器

sudo apt update

sudo apt install openssh-server

安装后，SSH 服务会自动启动，并默认监听 22 端口。

(4)防火墙，如果启用了ufw，需允许 SSH 端口

sudo ufw allow ssh

(5)ssh客户端与服务端的区别

对比项 SSH（客户端） SSHD（服务端）

角色 主动发起连接 被动等待连接

主要功能 登录远程服务器/传输文件 接受连接并提供安全Shell环境

进程名 ssh（命令行工具） sshd（后台守护进程）

配置文件 ~/.ssh/config /etc/ssh/sshd\_config

默认端口 动态（临时端口） 22（可修改）

(6)ssh访问方式

说明：用一台linux访问另一台linux

语法：

ssh 用户名@IP，例如 ssh [hsp@192.168.200.130](mailto:hsp@192.168.200.130)

使用ssh访问，如访问出现错误，可查看是否有该文件~/.ssh/known\_ssh，尝试删除该文件

解决。

登出命令：exit 或 logout

# 日志管理

1.介绍

日志文件是重要的系统信息文件，其中记录了许多重要的系统事件，包括用户的登录信息、

系统的启动信息、系统的安全信息、邮件相关信息、各种服务相关信息等。日志对于安

全来说很重要，它记录了系统每天发生的各种事件，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。可以理解为日志是用来记录重大事件的工具。

2.位置

日志文件的位置是/var/log/目录下，展示的文件以及含义如下：

/var/log/boot.log：系统启动日志

/var/log/cron：记录与系统定时任务相关的日志

/var/log/cups：记录打印信息的日志

/var/log/dmesg：记录系统在开机时内核自检的信息，也可以使用dmesg命令直接查看

内核自检信息

/var/log/btmp：记录错误登录的日志，这个文件是二进制文件，不能直接用vi查看，

而要直接使用lastb命令查看，例如：lastb

/var/log/lastlog：记录系统中所有用户最后一次登录时间的日志，这个文件是二进制文

件，要使用lastlog命令查看

/var/log/mailog：记录邮件信息的日志

/var/log/message：记录系统最重要的日志，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，首先检查这个日志文件

/var/log/secure：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，甚至添加用户和修改密码

都会记录在这个日志文件中

/var/log/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、

关机事件。是二进制文件，直接使用last命令查看，例如：last

/var/run/utmp：记录当前已经登录的用户信息，这个文件会随着用户的登录和注销而不

断变化，只记录当前用户的信息。这个文件不能用vi查看，而要使用w，who,users

等命令查看，例如：who

3.日志服务rsyslogd

查询Linux中的rsyslogd服务是否启动：

ps aux | grep "rsyslog" | grep -v "grep"

查询rsyslogd服务自启动状态：

systemctl list-unit-files | grep rsyslog

/etc/rsyslog.conf 中记录了rsyslogd管理的日志文件

该文件内容详解：

编辑文件格式为：\*.\* 存放日志文件的绝对路径

第一个\*代表日志类型，第二个代表\*代表日志级别

例如：local7.\* /var/log/boot.log

日志类型：

auth ##pam产生的日志

authpriv ##ssh、ftp等登录信息的验证信息

cron ##时间任务相关

kern ##内核

lpr ##打印

mail ##邮件

mark(syslog)-rsyslog ##服务内部的信息，时间标识

news ##新闻组

user ##用户程序产生的相关信息

uucp ##unix to nuix copy主机之间相关的通信

local 1-7 ##自定义的日志设备

日志级别：

debug ##有调试信息的，日志通信最多

info ##一般信息日志，最常用

notice ##最具有重要性的普通条件的信息

warning ##警告级别

err ##错误级别，阻止某个功能或模块不能正常工作的信息

crit ##严重级别，阻止整个系统或整个软件不能正常工作的信息

alert ##需要立刻修改的信息

emerg ##内核崩溃等重要信息

none ##什么都不记录

注意：从上到下，级别从低到高，记录信息越来越少

一般日志文件的内容的格式包含4列：

1.事件产生的时间

2.产生事件的服务器的主机名

3.产生事件的服务名或程序名

4.事件的具体信息

例如查看日志文件/var/log/secure中的部分内容：

May 22 10:29:16 CentOS7 sshd[7797]: Server listening on 0.0.0.0 port 22.

May 22 10:29:16 CentOS7 sshd[7797]: Server listening on :: port 22.

4.自定义日志

说明：根据上面的内容，我们知道了linux7.6的日志服务是rsyslogd，这个服务的配置文件

是/etc/rsyslog.conf，然后在这个配置文件中根据日志的类型和级别定义了一些日志及其

存放的路径，那么我们就可以自己在/etc/rsyslog.conf中添加一些内容来设置自己的日志

文件。

案例：在/etc/rsyslog.conf中添加一个日志文件/var/log/hsp.log，该日志的日志级别和类型都

设置为\*，也就是记录所有行为。

(1)打开/etc/rsyslog.conf

vim /etc/rsyslog.conf

(2)添加内容并保存退出

#增加自定义的日志

\*.\* /var/log/hsp.log

(3)创建文件/var/log/hsp.log

> /var/log/hsp.log

(4)重启日志服务

sudo systemctl restart rsyslog

(5)使用远程连接重新登录到linux

(6)查看/var/log/hsp.log中关于sshd（远程登录）的内容

cat /var/log/hsp.log | grep sshd

5.日志轮替

说明：日志轮替就是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。

介绍：

(1)Centos7使用logrotate进行日志轮替管理，要想改变日志轮替文件名字，通过

/etc/logrotate.conf配置

(2)如果配置文件中有“dateext”参数，那么日志会用日期来作为日志文件的后缀，例如

“secure-20201010”。这样日志文件名不会重叠，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即可。

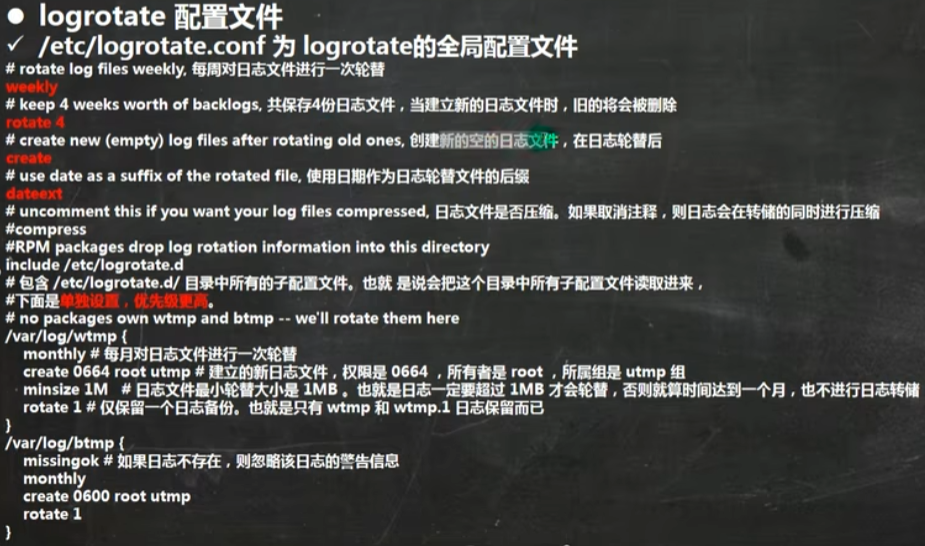
(3)如果配置文件中没有“dateext”参数，日志文件就需要进行改名了。当第一次进行日志

轮替时，当前的“secure”日志会自动改名为“secure.1”，然后新建“secure”日志，

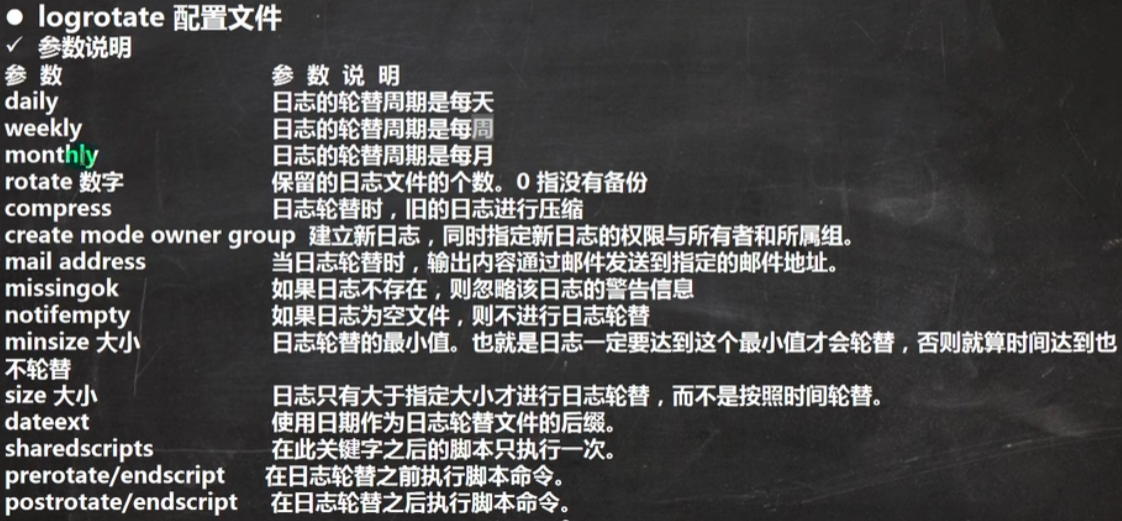
用来保存新的日志。当第二次进行日志轮替时，“secure.1”会自动改名为“secure.2”，

当前的“secure”日志会自动改名为“secure.1”，然后也会新建“secure”日志，用来

保存新的日志，以此类推。



参数说明：



把自己的日志加入日志轮替：

第一种方法是直接在/etc/logrotate.conf配置文件种写入该日志的轮替策略；

第二种方法是在/etc/logrotate.d/目录中新建立该日志的轮替文件，在该轮替文件中写入

正确的轮替策略，因为该目录中的文件都会被包含到主配置文件中，所以也可以把

日志加入轮替；

推荐使用第二种方法，因为系统中需要轮替的日志非常多，如果全部直接写入

/etc/logrotate.conf中，那么这个文件的可管理性就会非常差，不利于此文件的维护。

案例：在/etc/logrotate.d/下创建文件hsplog轮替自定义日志hsp.log

(1)进入目录

cd /etc/logrotate.d/

(2)创建文件并进入文件

vim hsplog

(3)插入内容并保存退出

/var/log/hsp.log

{

missingok

daily

copytruncate

rotate 7

notifempty

}

(4)生效

根据日志轮替机制原理，默认每天运行一次，因此新配置会在下次计划任务执行时自动加

载。若需立即验证配置是否正确，可手动运行以下命令：

sudo logrotate -d /etc/logrotate.conf # 调试模式（仅模拟，不实际执行）

sudo logrotate -f /etc/logrotate.conf # 强制立即执行轮替

日志轮替机制原理：

日志轮替之所以可以在指定的时间备份文件，是依赖系统的定时任务。在/etc/cron.daily/目录，其中有一个logrotate文件，logrotate通过这个文件依赖定时任务执行。该任务默认每天运行一次，因此新配置会在下次计划任务执行时自动加载。

6.内存日志

说明：默认存储在/run/log/journal/（内存中），重启后丢失。

相关命令：

journalctl 查看全部内存日志

journalctl -n 3 查看最新3条

journalctl --since 19:00 --until 19:10:10 查看起始时间到结束时间的日志，可加日期

journalctl -p err 报错日志

journalctl -o verbose 日志详细内容

journalctl \_PID=1245 \_COMM=sshd或journalctl | grep sshd 查看包含这些参数的日志

# 高级部分

1.linux启动流程

(1)首先Linux要通过自检，检查硬件设备有没有故障

(2)如果有多块启动盘的话，需要在BIOS中选择启动磁盘

(3)启动MBR中的BootLoader引导程序

(4)加载内核文件

(5)执行所有进程的父进程，老祖宗systemd

(6)欢迎界面

在Linux的启动流程中，加载内核文件时关键文件：

kernel文件：vmlinuz-3.10.0-957.el7.x86\_64

initrd文件：initramfs-3.10.0-957.el7.x86\_64.img

2.备份

Linux的备份和恢复有两种方式：

(1)把需要的文件（或分区）用TAR打包，下次需要恢复时，再解压覆盖即可。

(2)使用dump和restore命令。

如果linux上没有dump和restore命令，需要先安装这两个命令

yum -y install dump

yum -y install restore

本机使用这俩命令安装未成功，提示未找到镜像！！！

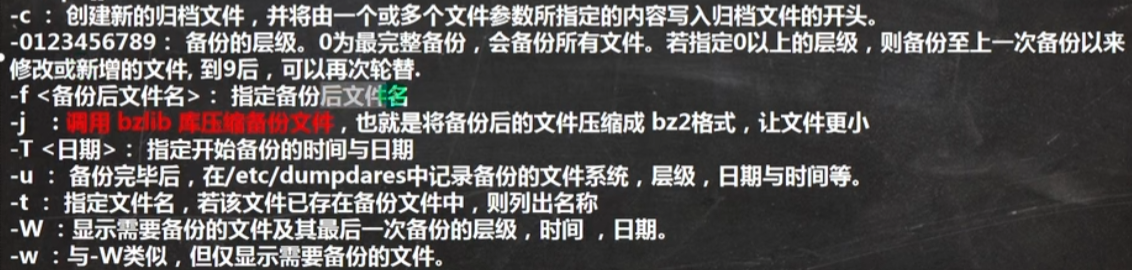
使用dump完成备份：

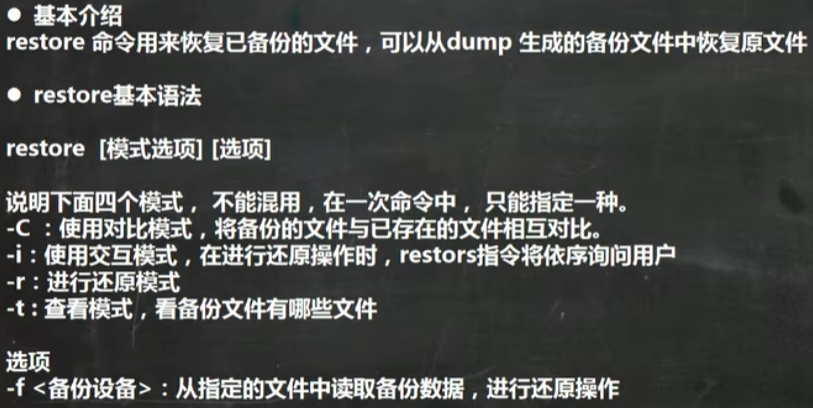
dump支持分卷备份和增量备份（所谓增量备份是指备份上一次备份后 修改/增加

过的文件，也称差异备份）

dump [-cu] [-123456789] [-f <备份后文件名>] [-T <日期>] [目录和文件系统]

dump [-wW]





3.linux可视化管理工具

国内的linux可视化管理工具主要有宝塔、Webmin等。