

1.给出你安装Ubuntu的完整步骤，并分别给出登陆界面（输入用户名和密码）和进入系统后进入终端窗口（显示登录用户路径及其提示符）界面两张截图。

在虚拟机中安装Ubuntu

截图

2.给出Ubuntu安装后根目录下的一级目录的名称及存放的信息类别。

3.什么是linux的挂载，实现挂载的步骤是什么？

挂载

挂载步骤

4.linux的文件系统是怎样的？在你的登录用户目录下创建一个文本文件输入“hello，你的名字！”。然后用ls -dil命令显示访问该文件的文件系统工作过程，并配上文字说明。

文件系统

创建文件

5.linux的文件类型有哪些？一个文件的属性有哪些？修改文件的属性的命令有哪些？

文件类型及对应包含的属性

修改文件的属性的命令

6.linux提供的参数配置方式有哪些？各自有什么特点？

软件参数配置

网络参数配置

硬件参数配置

7.linux的swap目录和proc目录是在什么时候和由谁建立的？swap和proc下的文件有哪些其内容是什么？

swap目录在系统创建时被创建也可以由用户创建

proc目录是操作系统安装时由系统建立的

8.linux进程设计了几种状态，它们之间的转换条件和转换时机是什么？

9.linux的管道的定义、特点及应用（举例）。

管带的定义

管道具有以下特点

管道的应用

10.给出20个常用的shell命令及常用参数，并简述其作用。

## 1.给出你安装Ubuntu的完整步骤，并分别给出登陆界面（输入用户名和密码）和进入系统后进入终端窗口（显示登录用户路径及其提示符）界面两张截图。

### 在虚拟机中安装Ubuntu

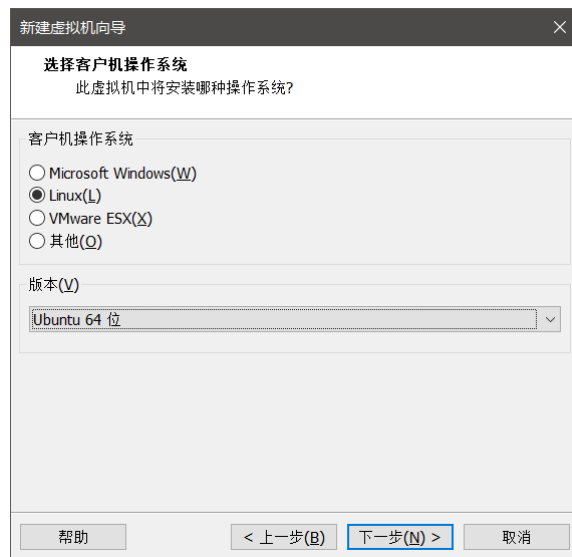
需要使用的工具有 `VMware workstation 15 Pro` , `ubuntu` 对应版本的镜像

(1).首先下载镜像，最好在官网下载，更加安全可靠，而且版本很全，我下载的是18.04版本

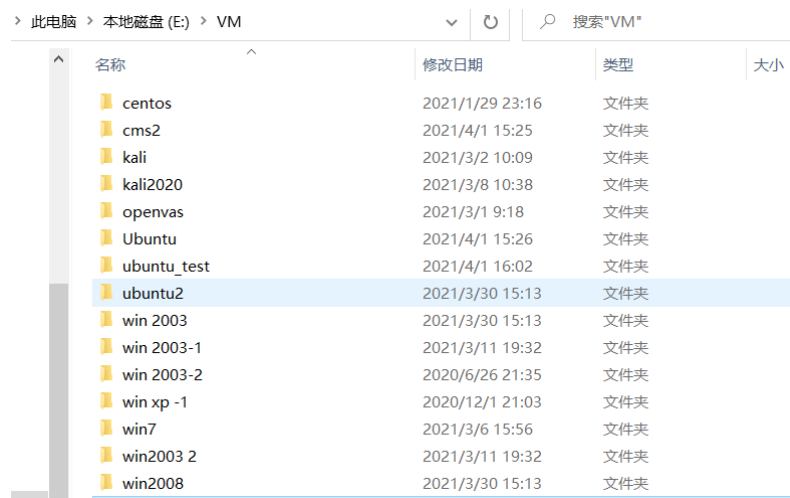
(2).打开VMware,选择"创建新的虚拟机"，选择"自定义（高级）"，再点击下一步

(3).到达"安装客户机操作系统"页面，选择"稍后安装操作系统"，点击下一步，"选择客户机操作系统"，选择"linux","ubuntu6

4",这里由你所下载的镜像决定



(4)."命名虚拟机", 这一步是给你创建的虚拟机取名字, 可以随便取, 并选择虚拟机安装的位置, 这里建议专门创建一个文件, 存放所有的虚拟机, 我选择的位置为 `E:\VM\ubtuntu_test`

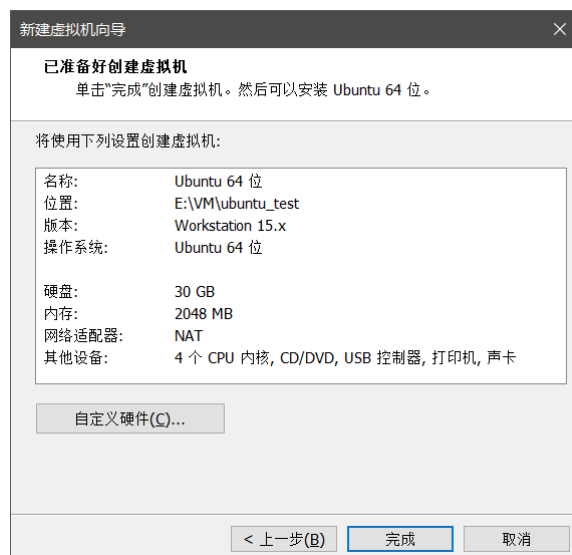


(5).配置虚拟机的处理器和内存, 这里可以根据自己电脑的性能适当的改变选项, 电脑的性能好, 则可以多加几个处理器, 内存也可以分配多一点, 这样虚拟机运行起来不会那么卡。

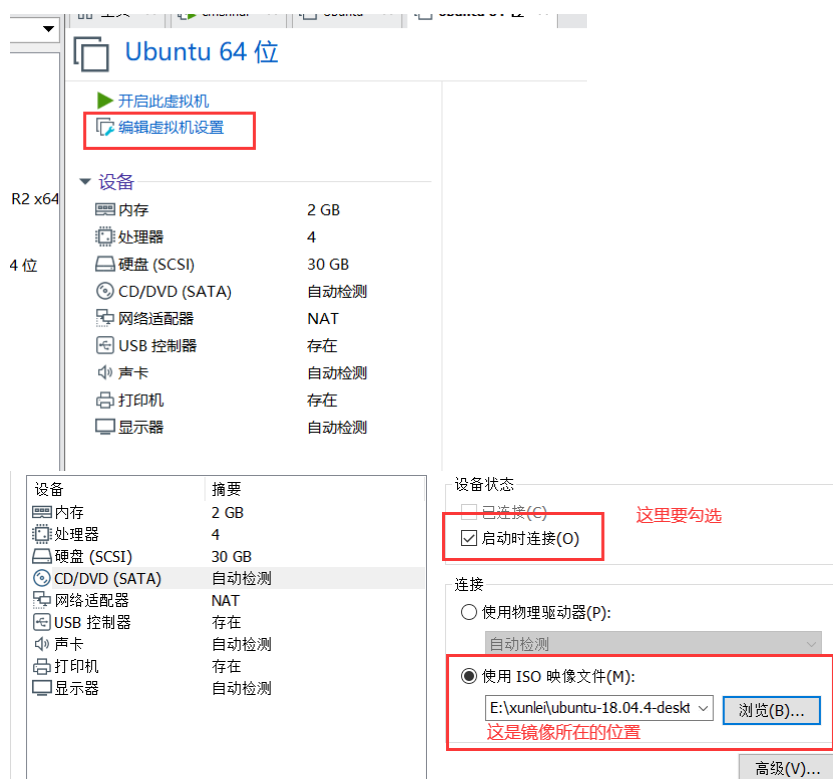
(6)."网络类型",选择默认的NAT模式, 这样你的虚拟机不但可以与物理机通信, 而且可以正常上网

(7)."选择I/O控制器类型", 这里默认, 点击下一步; "创建磁盘", 默认即可, 点击下一步; "选择磁盘", 默认即可, 点击下一步

(8)."指定磁盘容量", 这里也可以根据实际情况分配, 但是不能过小, 容量过小的可能安装失败, 选择"将虚拟磁盘存储为单个文件", 这样对于新手来说更简单一些, 点击下一步; 默认即可, 再点击下一步, 点击完成



(9).进入首页，查看刚刚创建的虚拟机，此时需要点击"编辑虚拟机设置"，为该虚拟机选择镜像文件，点击确定

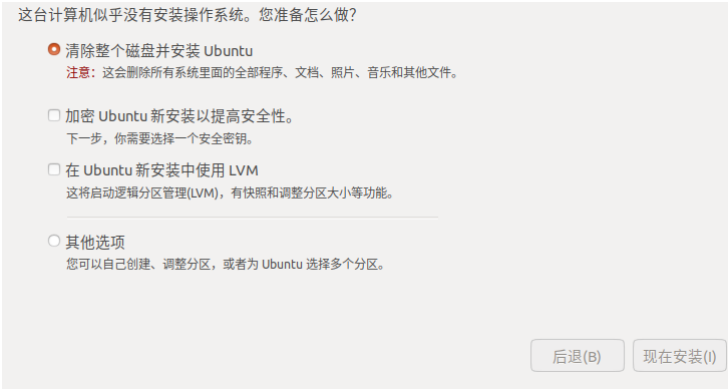


(10).回到首页，开启虚拟机，开始安装系统，首先选择语言，这里选择中文简体，点击"试用Ubuntu"



(11)."最小安装",这里可以不更新系统,节省时间

(12)."安装类型",默认即可,点击现在安装



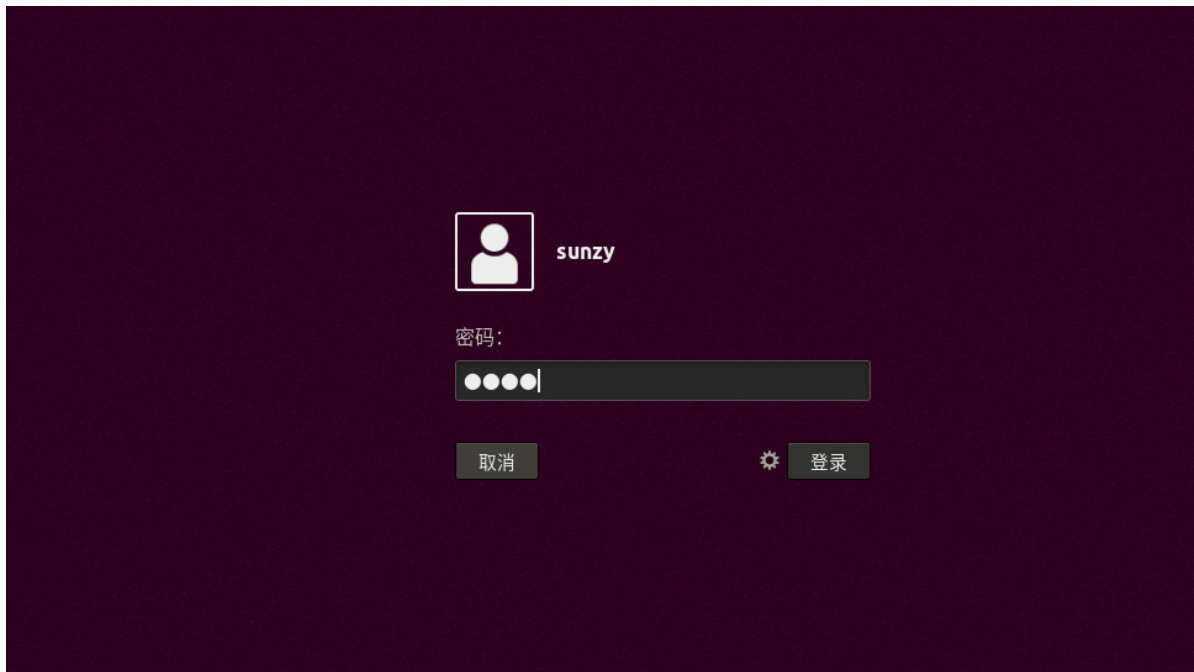
(13).选择时区,输入"shanghai",确定

(14).添加用户,点击继续,之后开始安装系统,等待即可

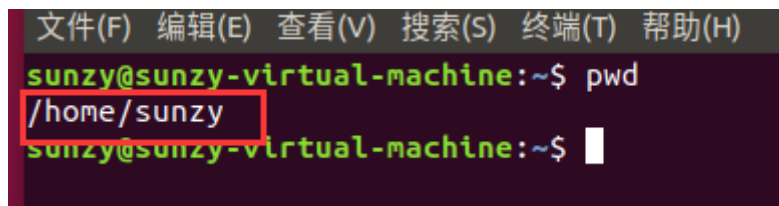
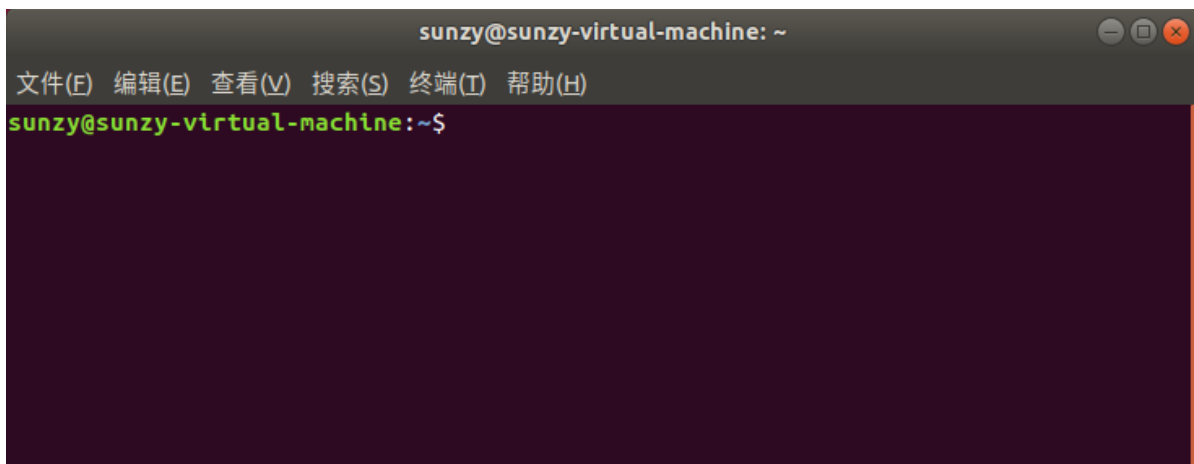


## 截图

用户登录, 输入密码



显示登录用户路径和提示符，可以看到该用户的提示符为 `$`，也就是普通用户



## 2.给出Ubuntu安装后根目录下的一级目录的名称及存放的信息类别。

- 1 /bin 该目录中存放Linux的常用命令。
- 2 /boot 该目录默认下存放的是Linux的启动文件和内核。
- 3
- 4 /cdrom 该目录在刚安装系统时是空的，你可以将光驱文件系统挂在这个目录下。
- 5
- 6 /dev 该目录包含了Linux系统中使用的所有外部设备，它实际上是访问这些外部设备的端口，你可以访问这些外部设备，与访问一个文件或一个目录没有区别。例如在系统中键入“`cd /dev/cdrom`”，就可以看到光驱中的文件；键入“`cd /dev/mouse`”即可看鼠标的相关文件。
- 7 /etc 该目录存放系统管理时要用到的各种配置文件和子目录，例如网络配置文件、文件系统、X系统配置文件、设备配置信息、设置用户信息等。
- 8

9 /home 如果建立一个名为“xx”的用户，那么在/home目录下就有一个对应的“/home/xx”路径，用来存放该用户的主目录。比如上面的sunzy用户。

10

11 /lib 该目录用来存放系统动态链接共享库，几乎所有的应用程序都会用到该目录下的共享库。

12

13

14 /media: 提供挂载（mounting）和自动挂载设备的标准位置，如远程文件系统和可移动介质（目录名为cdrecorder、floppy等）。

15

16 /mnt 临时将别的文件系统挂在该目录下。

17

18 /opt 第三方软件在安装时默认会找这个目录，所以你没有安装此类软件时它是空的，但如果你一旦把它删除了，以后在安装此类软件时就有可能碰到麻烦。

19

20 /proc 可以在该目录下获取系统信息，这些信息是在内存中由系统自己产生的，该目录的内容不在硬盘上而在内存里。

21

22 /root 如果你是以超级用户的身份登录的，这个就是超级用户的主目录。

23

24 /sbin 该目录用来存放系统管理员使用的管理程序。初始化的内存盘。在linux内核启动前，boot loader会将存储介质（一般是硬盘）中的

25

26 /sys: 一个类似与/proc的文件系统，在Linux2.6内核中最新出现的，包含的文件用于获得硬件状态并反映内核看到的系统设备树。它使用了/proc中的很多帮助。

27

28 /tmp 用来存放不同程序执行时产生的临时文件，该目录会被系统自动清理干净。

29

30 /usr 用户的应用程序和文件几乎都存放在该目录下。

31 /var 该目录存放那些经常被修改的文件，包括各种日志、数据文件。

32

33 /initrd 它的英文含义是boot loader initialized RAM disk,就是由boot loader,文件加载到内存，内核启动时会在访问真正的根文件系统前先访问该内存中的initrd文件系统。

### 3.什么是linux的挂载，实现挂载的步骤是什么？

#### 挂载

指的就是将设备文件中的顶级目录连接到 Linux 根目录下的某一目录（最好是空目录），**访问此目录就等同于访问设备文件。**

Linux 系统中“一切皆文件”，所有文件都放置在以根目录为树根的树形目录结构中。在linux 看来，任何硬件设备也都是文件，它们各有自己的一套文件系统（文件目录结构）。

因此产生的问题是，当在 Linux 系统中使用这些硬件设备时，只有将Linux本身的文件目录与硬件设备的文件目录合二为一，硬件设备才能为我们所用。合二为一的过程称为“挂载”。

如果不挂载，通过Linux系统中的图形界面系统可以查看找到硬件设备，但命令行方式无法找到。

并不是根目录下任何一个目录都可以作为挂载点，由于挂载操作会使得原有目录中文件被隐藏，因此**根目录以及系统原有目录都不要作为挂载点，会造成系统异常甚至崩溃，挂载点最好是新建的空目录。**

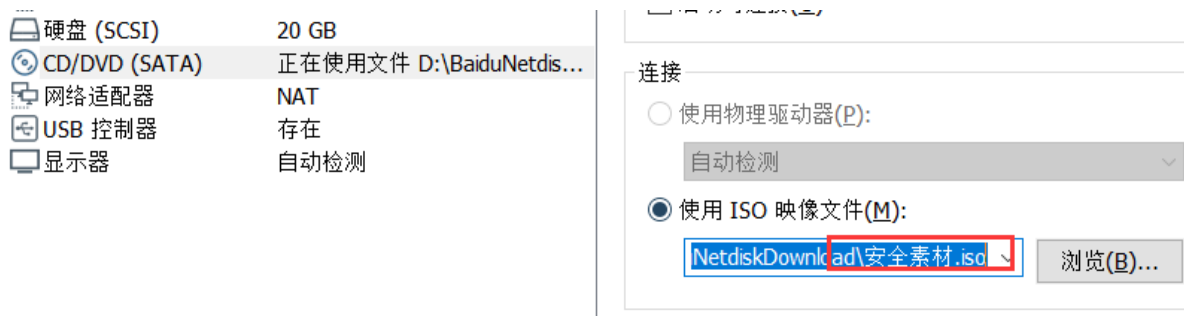
## 挂载步骤

### (1).创建挂载点

```
1 | sudo mkdir /mnt/cdrom
2 | //先提升权限 在 /mnt 目录下创建一个空文件夹cdrom作为光盘的挂载点（任何一个空目录都可以作为挂载点）
```

### (2).打开光驱，放入光盘

如果用的是VMware中的虚拟机，进入虚拟机设置，选择“CD/DVD”硬件，勾选“已连接”和“启动时连接”，在连接处，选择“使用ISO映像文件”，浏览选择本地电脑中下载好的ISO光盘镜像文件，确定即可。



### (3).执行挂载命令

```
1 | sudo mount /dev/sr0 /mnt/cdrom
```

此时光盘挂载已经成功，进入 /mnt/cdrom，查看即可

### (4).解除挂载

```
1 | cd 其他目录
2 | sudo umount /dev/sr0
```

## 4.linux的文件系统是怎样的？在你的登录用户目录下创建一个文本文件输入“hello，你的名字！”。然后用ls -dil命令显示访问该文件的文件系统工作过程，并配上文字说明。

### 文件系统

文件系统是一种对物理空间的组织方式，通常在格式化硬盘时创建。Linux有自己的文件系统,如ext2fs文件系统，ext3fs文件系统，ext4fs文件系统，ReiserFS文件系统。

- ext2fs

硬盘分区首先被分割为一个一个的“Block”，每个Block就是实际用来存储数据的单元，大小相同，Block按照0，1，2，3的顺序进行编号，第一个Block的编号为0。对于ext2文件系统来说，支持的Block的大小有1024字节/2048字节/4096字节，Block的大小在创建文件系统的时候可以通过参数指定，如果不指定，则会从/etc/mke2fs.conf文件中读取对应的值。原则上，Block的大小与数量在格式化后就不能够发生改变，每个Block内只会存放至多一个文件的数据（即不会出现两个文件的数据被放入同一个Block的情况），如果文件超过一个Block，则会占用多个Block来存放文件，如果小于一个Block大小，则这个Block剩余的空间就浪费掉了。

- ext3fs



•Ext3文件系统是直接Ext2文件系统发展而来，目前ext3文件系统已经非常稳定可靠。它完全兼容ext2文件系统。用户可以平滑地过渡到一个日志功能健全的文件系统中来。

•Ext3有多种日志模式，一种工作模式是对所有的文件数据及metadata（定义文件系统中数据的数据,即数据的数据）进行日志记录（data=journal模式）；另一种工作模式则是只对metadata记录日志，而不对数据进行日志记录，也即所谓data=ordered或者data=writeback模式。系统管理人员可以根据系统的实际工作要求，在系统的工作速度与文件数据的一致性之间作出选择。

- ext4fs

与 Ext3 兼容，更大的文件系统和更大的文件，无限数量的子目录。自动优化

- Resierfs

- 先进的日志机制——ReiserFS有先进的日志(journaling/logging)功能 机制。日志机制保证了在每个实际数据修改之前，相应的日志已经写入硬盘。文件与数据的安全性有了很大提高。
- 高效的磁盘空间利用——Reiserfs对一些文件不分配inode。而是将这些文件打包，存放在同一个磁盘分块中。而其它文件系统则为每个小文件分别放置到一个磁盘分块中。这意味着：如果有10000个小文件，就要占用10000个分块。想想看这多浪费磁盘空间。
- 独特的搜寻方式——ReiserFS基于快速平衡树(balanced tree)搜索，平衡树在性能上非常卓越，这是一种非常高效的算法。
- 支持海量磁盘——可轻松管理上百G的文件系统，ReiserFS文件系统最大支持的文件系统尺寸为16TB。这非常适合企业级应用中。

## 创建文件

```
sunzy@sunzy-virtual-machine:~$ echo hello,sunzy\!>hello
sunzy@sunzy-virtual-machine:~$ cat hello
hello,sunzy!
sunzy@sunzy-virtual-machine:~$ ls -dil hello
797940 -rw-r--r-- 1 sunzy sunzy 13 4月 1 22:04 hello
sunzy@sunzy-virtual-machine:~$
```

1	向文件中写内容
2	<code>echo hello,sunzy\!&gt;hello</code>
3	
4	查看文件
5	<code>ls -dil hello</code>
6	显示的内容
7	<code>797940 -rw-r--r-- 1 sunzy sunzy 13 4月 1 22:01 hello</code>
8	
9	797940是文件的节点
10	<code>-rw-r--r--</code>
11	hello是一个文件，所有者是sunzy,权限为rw,所有用户组为sunzy,权限为r,其他用户权限为r
12	创建时间是4月1日22:01

使用 `--help` 参数查看ls命令各参数的含义

1	<code>-d, --directory</code>	list directories themselves, not their contents
2	<code>-i, --inode</code>	print the index number of each file
3	<code>-l</code>	使用较长格式列出信息

由上面的帮助信息可知 `ls -idl`



是列出该文件自身，如果是文件夹则只列出该文件夹，不显示其中的内容，显示该文件的节点信息，并以横列表的格式显示。

### 工作过程如下

Linux文件系统为每个文件都分配两个数据结构，索引节点（index node）和目录项（directory entry）。

索引节点，简称为inode，用来记录文件的元数据，比如inode编号、文件大小、访问权限、修改日期、数据的位置等。索引节点和文件一一对应，它跟文件内容一样，都会被持久化存储到磁盘中。所以记住，索引节点同样占用磁盘空间。

目录项，简称为dentry，用来记录文件的名称、索引节点指针以及与其他目录项的关联关系。多个关联的目录项，就构成了文件系统的目录结构。不过，不同于索引节点，目录项是由内核维护的一个内存数据结构，所以通常也被叫做目录项缓存。

索引节点是每个文件的唯一标志，而目录项维护的正是文件系统的树状结构。目录项和索引节点的关系是多对一，可以理解为，一个文件可以有多个别名。

## 5.linux的文件类型有哪些？一个文件的属性有哪些？修改文件的属性的命令有哪些？

### 文件类型及对应包含的属性

- 普通文件类型  
Linux中最多种文件类型, 包括 纯文本文件(ASCII); 二进制文件(binary); 数据格式的文件(data);各种压缩文件.第一个属性为 [-]
- 目录文件  
就是目录，能用 # cd 命令进入的。第一个属性为 [d]，例如 [drwxrwxrwx]
- 块设备文件  
块设备文件：就是存储数据以供系统存取的接口设备，简单而言就是硬盘。例如一号硬盘的代码是 /dev/hda1等文件。第一个属性为 [b]
- 字符设备  
字符设备文件：即串行端口的接口设备，例如键盘、鼠标等等。第一个属性为 [c]
- 套接字文件  
这类文件通常用在网络数据连接。可以启动一个程序来监听客户端的要求，客户端就可以通过套接字来进行数据通信。第一个属性为 [s]，最常在 /var/run目录中看到这种文件类型
- 管道文件  
FIFO也是一种特殊的文件类型，它主要的目的是，解决多个程序同时存取一个文件所造成的错误。FIFO是first-in-first-out(先进先出)的缩写。第一个属性为 [p]
- 链接文件  
类似Windows下面的快捷方式。第一个属性为 [l]，例如 [lrwxrwxrwx]

### 修改文件的属性的命令

每个文件的属性由左边第一部分的 10 个字符来确定

文件 类型	属主 权限			属组 权限			其他用户 权限		
0	1	2	3	4	5	6	7	8	9
<b>d</b>	<b>rwX</b>			<b>r-X</b>			<b>r-X</b>		
目录 文件	读	写	执行	读	写	执行	读	写	执行

- 更改文件用户组：chgrp

```
1 chgrp [-R] 属性组 文件名
2 // -R: 递归更改文件属组，就是在更改某个目录文件的属组时，如果加上-R的参数，那么该目录下的所有文件的属组都会更改。
```

- 更改文件属主，也可以同时更改文件属组：chown

```
1 chown [-R] 属主名 文件名
2 chown [-R] 属主名:属组名 文件名
```

- 更改文件9个属性: chmod

Linux文件属性有两种设置方法，一种是数字，一种是符号。

Linux 文件的基本权限就有九个，分别是 **owner/group/others(拥有者/组/其他)** 三种身份各有自己的 **read/write/execute** 权限。

先复习一下刚刚上面提到的数据：文件的权限字符为：**-rwxrwxrwx**，这九个权限是三个三个一组的！其中，我们可以使用数字来代表各个权限，各权限的分数对照表如下：

- r:4
- w:2
- x:1

每种身份(owner/group/others)各自的三个权限(r/w/x)分数是需要累加的，例如当权限为：**-rwxrwx---** 分数则是：

- owner = rwx = 4+2+1 = 7
- group = rwx = 4+2+1 = 7
- others = --- = 0+0+0 = 0

所以等一下我们设定权限的变更时，该文件的权限数字就是 **770**。变更权限的指令 chmod 的语法是这样的：

```
1 chmod [-R] 属性值 文件或目录
```

## 6.linux提供的参数配置方式有哪些？各自有什么特点？

### 软件参数配置

- .deb软件包的管理：dpkg

- 1 `-i`选项 实现`install` 安装
- 2 `-l`选项实现`list`查看
- 3 `-s`选项实现`search`查找
- 4 `-r`选项实现`remove`卸载

### • 安装和卸载软件包命令 `apt-get`

- 1 `apt-get update` 在修改`/etc/apt/sources.list`或`/etc/apt/preferences`之后运行该命令。需要定期运行这一命令以确保您的软件包列表是最新的。
- 2 `apt-get install packagename` 安装一个新软件包（与下文的`aptitude`功能类似）
- 3 `apt-get remove packagename` 卸载一个已安装的软件包（保留配置文档）
- 4 `apt-get remove --purge packagename` 卸载一个已安装的软件包（删除配置文档），注意“`--`”符号必不可少
- 5 `apt-get autoremove packagename` 删除包及其依赖的软件包
- 6 `apt-get autoremove --purge packagename` 删除包及其依赖的软件包+配置文件，比上面的要删除的彻底一点

### • 还有图形化APT

命令行的特点是更加的方便，快速，对于经常使用linux系统的，熟悉linux命令的人很友好便捷，但是对于新手来说很困难

而图形化的APT对于新手来说很友好，如同在windows上操作，无缝对接。

## 网络参数配置

### • 图形化方式网络配置

该方法很简单，就像在windows上一样操作即可



### • 命令行配置

`ifconfig`命令

直接在命令行中输入`ifconfig`，可以看到本机的网络配置信息

```

root@sunzy-virtual-machine:/# ifconfig
br-3154a4f3e7de: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.18.0.1 netmask 255.255.0.0 broadcast 172.18.255.255
    ether 02:42:b6:e2:1d:92 txqueuelen 0 (以太网)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

br-d5a00104618c: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.19.0.1 netmask 255.255.0.0 broadcast 172.19.255.255

```

```

1 | ifconfig 网络端口 IP地址 hw <HW> MAC地址 netmask 掩码地址 broadcast 广播地址
   | [up/down]
2 |
3 | ifconfig eth0 down;关闭网卡;
4 | ifconfig eth0 192.168.1.99 broadcast 192.168.1.255 netmask 255.255.255.0 配
   | 置网卡IP地址、广播地址和掩码;
5 | ifconfig eth0 up启动激活网卡
6 | ifconfig eth0 查看网卡配置。

```

- **直接编辑网络配置文件**

编辑/etc/host文件，设置主机名和IP地址间的映射

编辑/etc/sysconfig/network文件，启动/关闭网络，设置主机名和网关IP地址。

编辑/etc/resolv.conf文件，设置主/次域名服务器和IP地址间的映射

编辑/etc/sysconfig/network-scripts/ifcfg-eth0文件，设置网卡参数

## 硬件参数配置

- **查看机器所有硬件信息:**

```
1 | dmidecode | more
```

```

root@sunzy-virtual-machine:/# dmidecode |more
# dmidecode 3.1
Getting SMBIOS data from sysfs.
SMBIOS 2.7 present.
620 structures occupying 29060 bytes.
Table at 0x000E0010.
...
Handle 0x0000, DMI type 0, 24 bytes
BIOS Information
    Vendor: Phoenix Technologies LTD
    Version: 6.00
    Release Date: 07/29/2019
    Address: 0xEA480

```

```
1 | dmesg | more
```

```

root@sunzy-virtual-machine:/# dmesg | more
[ 0.000000] Linux version 5.4.0-66-generic (buildd@lgw01-amd64-016) (gcc version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04))
P Fri Feb 5 11:17:31 UTC 2021 (Ubuntu 5.4.0-66.74~18.04.2-generic 5.4.86)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-5.4.0-66-generic root=UUID=8b7c7fe7-0f3f-4878-850d-66a9380c
[ 0.000000] KERNEL supported cpus:
[ 0.000000] Intel GenuineIntel
[ 0.000000] AMD AuthenticAMD
[ 0.000000] Hygon HygonGenuine

```

- 查看CPU信息

方法一:

Linux下CPU相关的参数保存在 /proc/cpuinfo 文件里

```
1 | cat /proc/cpuinfo | more
```

```

root@sunzy-virtual-machine:/# cat /proc/cpuinfo | more
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 158
model name     : Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz
stepping       : 10
microcode      : 0xde
cpu MHz        : 2303.999
cache size     : 8192 KB
physical id    : 0
siblings       : 1

```

本机的cpu

方法二:

可以查看到相关CPU的启动信息

```
1 | dmesg | grep CPU
```

- 查看磁盘信息

系统上的磁盘(包括U盘)的分区以及大小相关信息

```
1 | fdisk -l
```

```

root@sunzy-virtual-machine:/# fdisk -l
Disk /dev/loop0: 2.5 MiB, 2600960 字节, 5080 个扇区
单元: 扇区 / 1 * 512 = 512 字节
扇区大小(逻辑/物理): 512 字节 / 512 字节
I/O 大小(最小/最佳): 512 字节 / 512 字节

Disk /dev/loop1: 303.1 MiB, 317796352 字节, 620696 个扇区
单元: 扇区 / 1 * 512 = 512 字节
扇区大小(逻辑/物理): 512 字节 / 512 字节
I/O 大小(最小/最佳): 512 字节 / 512 字节

Disk /dev/loop2: 217.9 MiB, 228478976 字节, 446248 个扇区
单元: 扇区 / 1 * 512 = 512 字节
扇区大小(逻辑/物理): 512 字节 / 512 字节
I/O 大小(最小/最佳): 512 字节 / 512 字节

```

直接查看

```
1 | cat /proc/partitions
```

```

root@sunzy-virtual-machine:/# cat /proc/partitions
major minor #blocks name

    7         0         2540 loop0
    7         1        310348 loop1
    7         2        223124 loop2
    7         3        101588 loop3
    7         4        165288 loop4
    7         5          276 loop5
    7         6         2544 loop6
    7         7        224248 loop7
   11         0        2076704 sda

```

## 7. linux的swap目录和proc目录是在什么时候和由谁建立的？ swap和proc下的文件有哪些其内容是什么？

### swap目录在系统创建时被创建也可以由用户创建

创建命令

```

1  1. 创建要作为swap分区的文件:增加1GB大小的交换分区, 则命令写法如下, 其中的count等于想要的
   块的数量 (bs*count=文件大小)。
2  dd if=/dev/zero of=/root/swapfile bs=1M count=1024
3
4  2. 格式化为交换分区文件:
5  mkswap /root/swapfile #建立swap的文件系统
6
7  3. 启用交换分区文件:
8  swapon /root/swapfile #启用swap文件
9
10 4. 使系统开机时自启用, 在文件/etc/fstab中添加一行:
11 /root/swapfile swap swap defaults 0 0

```

swap 分区通常被称为交换分区, 这是一块特殊的硬盘空间, 即当实际内存不够用的时候, 操作系统会从内存中取出一部分暂时不用的数据, 放在交换分区中, 从而为当前运行的程序腾出足够的内存空间。

也就是说, 当内存不够用时, 我们使用 swap 分区来临时顶替。这种“拆东墙, 补西墙”的方式应用于几乎所有的操作系统中

### proc目录是操作系统安装时由系统建立的

```

1  /proc 目录下的文件
2
3  /proc/cpuinfo      CPU的信息 (型号、家族、缓存大小等)
4
5  /proc/meminfo      物理内存、交换空间
6
7  /proc/mounts        已加载的文件系统的列表
8
9  /proc/devices      可用设备的列表
10
11 /proc/filesystems    被支持的文件系统

```

```
12
13 /proc/modules 已加载的模块
14
15 /proc/virsion 内核版本
16
17 /proc/cmdline 系统启动时输入的内核命令行参数
18
19 /proc/xxx xxx是指以进程PID（数字编号）命名的目录，每一个目录表示一个进程（即线程组）。
20
21 /proc/swaps 要获知swap空间的使用情况
22
23 /proc/uptime 获取系统的正常运行时间
24
25 /proc/fs/nfsd/exports 列出由NFS共享的文件系统
26
27 /proc/kmsg 该文件被作为内核日志信息源，它可以被作为一个系统信息调用的接口使用
28
29 /proc/self -- 到当前进程/proc目录的符号链接，通过这个目录可以获取当前运行进程的信息。
30
31 /proc/pci -- 挂载在PCI总线上的设备
32
33 /proc/tty/driver/serial --串口配置、统计信息
34
35 /proc/version -- 系统版本信息
36
37 /proc/sys/kernel/ostype
38
39 /proc/sys/kernel/osrelease
40
41 /proc/sys/kernel/version
42
43 /proc/sys/kernel/hostname -- 主机名
44
45 /proc/sys/kernel/domainname -- 域名
46
47 /proc/partitions -- 硬盘设备分区信息
48
49 /proc/sys/dev/cdrom/info -- CDROM信息
50
51 /proc/locks -- 当前系统中所有的文件锁
52
53 /proc/loadavg -- 系统负荷信息
54
55 /proc/uptime -- 系统启动后的运行时间
```



cpuinfo	0 文件	2021/04/07 15:05	-r--r--r--	root/root
crypto	0 文件	2021/04/07 15:05	-r--r--r--	root/root
devices	0 文件	2021/04/07 15:01	-r--r--r--	root/root
diskstats	0 文件	2021/04/07 15:05	-r--r--r--	root/root
dma	0 文件	2021/04/07 15:05	-r--r--r--	root/root
execdomains	0 文件	2021/04/07 15:05	-r--r--r--	root/root
fb	0 文件	2021/04/07 15:05	-r--r--r--	root/root
filesystems	0 文件	2021/04/07 15:05	-r--r--r--	root/root
interrupts	0 文件	2021/04/07 15:01	-r--r--r--	root/root
iomem	0 文件	2021/04/07 15:05	-r--r--r--	root/root
ioports	0 文件	2021/04/07 15:05	-r--r--r--	root/root
kallsyms	0 文件	2021/04/07 15:05	-r--r--r--	root/root
kcore	128 TB 文件	2021/04/07 15:05	-r-----	root/root
keys	0 文件	2021/04/07 15:05	-r--r--r--	root/root
key-users	0 文件	2021/04/07 15:05	-r--r--r--	root/root
kmsg	0 文件	2021/04/03 10:16	-r-----	root/root
kpagecgroup	0 文件	2021/04/07 15:05	-r-----	root/root
kpagecount	0 文件	2021/04/07 15:05	-r-----	root/root
kpageflags	0 文件	2021/04/07 15:05	-r-----	root/root
loadavg	0 文件	2021/04/05 19:26	-r--r--r--	root/root
locks	0 文件	2021/04/07 15:05	-r--r--r--	root/root
mdstat	0 文件	2021/04/07 15:05	-r--r--r--	root/root
meminfo	0 文件	2021/04/05 19:26	-r--r--r--	root/root
misc	0 文件	2021/04/07 15:05	-r--r--r--	root/root
modules	0 文件	2021/04/07 15:05	-r--r--r--	root/root
mounts	0 文件	2021/04/07 15:06	-r--r--r--	root/root
mtrr	0 文件	2021/04/03 10:16	-rw-r--r--	root/root
pagetypeinfo	0 文件	2021/04/07 15:05	-r-----	root/root
partitions	0 文件	2021/04/07 15:05	-r--r--r--	root/root
sched_debug	0 文件	2021/04/07 15:05	-r--r--r--	root/root
schedstat	0 文件	2021/04/07 15:05	-r--r--r--	root/root

## 8.linux进程设计了几种状态，它们之间的转换条件和转换时机是什么？

- 五种状态及转换时机
  - 运行：正在运行或在运行队列中等待。
  - 中断：休眠中，受阻，在等待某个条件的形成或接受到信号。
  - 不可中断：收到信号不唤醒和不可运行，进程必须等待直到有中断发生。
  - 僵死：进程已终止，但进程描述符存在，直到父进程调用wait4()系统调用后释放。
  - 停止：进程收到SIGSTOP，SIGSTP，SIGTIN，SIGTOU信号后停止运行运行。

## 9.linux的管道的定义、特点及应用（举例）。

### 管带的定义

管道是一种两个进程间进行单向通信的机制。因为管道传递数据的单向性，管道又称为半双工管道。管道的这一特点决定了其使用的局限性。

## 管道具有以下特点

(1).数据只能由一个进程流向另一个进程（其中一个读管道，一个写管道）；如果要进行双工通信，需要建立两个管道。

(2).管道只能用于父子进程或者兄弟进程间通信。，也就是说管道只能用于具有亲缘关系的进程间通信。

**注意：**从管道读数据是一次性操作，数据一旦被读，它就从管道中被抛弃，释放空间以便写更多的数据。

(3).管道式半双工的，一端读，另一端只能写。

(4).双方使用同一个管道(即一方读，另一方写)，一方打开管道之后，需要等待另一方打开，才能继续执行。

(5).读端读取管道时，如果管道内没有数据，read就会阻塞，一直等着。

(6).如果管道写端关闭，读端read时,会返回0.

(7).如果读端关闭，写端还继续向管道写数据，则写端程序会直接崩溃。

## 管道的应用

- 使用竖线（|）来把输出用管道导入到命令中

比如查看 /etc 下的目录

```
1 ls -al /etc | less
2 //
3 现在，你就可以一个屏幕一个屏幕地查看 /etc 目录的内容了。要向前移动一个屏幕，按
  [Space] 键；要向后移动一个屏幕，按[Enter]键；要退出，按 [q] 键
4
5 less 来搜索文本文件的输出，按 [/] ，然后键入你想在文件内搜索的内容 效果如下/bash
6 带有bash的文件就会高亮显示
```

```
drwxr-xr-x  3 root root  4096 2月  4 2020 avahi
-rw-r--r--  1 root root  2319 4月  5 2018 bash.bashrc
-rw-r--r--  1 root root    45 4月  2 2018 bash_completion
drwxr-xr-x  2 root root  4096 3月 11 10:09 bash_completion.d
-rw-r--r--  1 root root   367 1月 27 2016 bindresvport.blacklist
drwxr-xr-x  2 root root  4096 4月 21 2018 binfmt.d
drwxr-xr-x  2 root root  4096 11月 20 21:39 bluetooth
```

- 统计当前目录下有多少个文件

```
1 ls | wc -l
```

```
root@sunzy-virtual-machine:~# ls /etc | wc -l
222
```

- 查看当前目录下的前n-1个文件

```
1 ls -l | head -3
```

```

root@sunzy-virtual-machine:~/test# ls -l | head -3
总用量 4
-rw-r--r-- 1 root root 0 4月  6 23:05 1
-rw-r--r-- 1 root root 0 4月  6 23:05 2
root@sunzy-virtual-machine:~/test# ls -l | head -4
总用量 4
-rw-r--r-- 1 root root 0 4月  6 23:05 1
-rw-r--r-- 1 root root 0 4月  6 23:05 2
-rw-r--r-- 1 root root 6 4月  6 23:05 3

```

- 查看文件包含某个字符的字符行

```
1 | cat info.php | grep name
```

```

?>root@sunzy-virtual-machine:/www/src/public# cat info.php | grep name
$name1='张三';
$name2='李四';
$name3='王五';
$name4='alice';
    'name1' => $name1,
    'name2' => $name2,
    'name3' => $name3,
    'name4' => $name4,

```

- 查看内存使用情况，并读取某一行

```
1 | free -m | grep 交换
```

```

root@sunzy-virtual-machine:/# free -m
              总计          已用          空闲          共享          缓冲/缓存          可用
内存:         1895         1354           78           10           461          363
交换:           947           682          264
root@sunzy-virtual-machine:/# free -m | grep 交换
交换:           947           682          264

```

- 向一个文件里面添加字符

```
1 | echo "Hahahahhh" | cat >hello.txt
```

```

root@sunzy-virtual-machine:~/test# echo "Hahahahhh" | cat >hello.txt
root@sunzy-virtual-machine:~/test# cat hello.txt
Hahahahhh

```

## 10. 给出20个常用的shell命令及常用参数，并简述其作用。

- 1.cd 用于切换当前目录，它的参数是要切换到的目录的路径，可以是绝对路径，也可以是相对路径

```

1 cd ../
2 // 回到当前目录的前一个目录
3 cd /
4 //到根目录
5 cd ./test
6 //进入当前目录的test目录中

```

- 2.ls

#### 查看文件与目录的命令

```

1 -l : 列出长数据串, 包含文件的属性与权限数据等
2 -a : 列出全部的文件, 连同隐藏文件(开头为.的文件)一起列出来(常用)
3 -d : 仅列出目录本身, 而不是列出目录的文件数据
4 -h : 将文件容量以较易读的方式(GB, kB等)列出来
5 -R : 连同子目录的内容一起列出(递归列出), 等于该目录下的所有文件都会显示出来

```

注: 这些参数也可以组合使用, 下面举两个例子:

```

1 ls -l #以长数据串的形式列出当前目录下的数据文件和目录
2 ls -lR #以长数据串的形式列出当前目录下的所有文件

```

```

root@sunzy-virtual-machine:/# ls -la
总用量 1015792
drwxr-xr-x 28 root root 4096 4月 6 23:03 .
drwxr-xr-x 28 root root 4096 4月 6 23:03 ..
drwxr-xr-x 2 root root 4096 3月 11 10:08 bin
drwxr-xr-x 3 root root 4096 3月 12 16:46 boot
drwxrwxr-x 2 root root 4096 5月 23 2020 cdrom
drwxr-xr-x 3 root root 4096 3月 12 16:48 compose_lnpm
drwxr-xr-x 18 root root 4340 4月 7 17:23 dev
drwxr-xr-x 128 root root 12288 4月 3 10:16 etc
drwxr-xr-x 5 root root 4096 3月 11 16:44 home
lrwxrwxrwx 1 root root 32 3月 11 10:11 initrd.img -> boot/initrd.img
lrwxrwxrwx 1 root root 32 3月 11 10:11 initrd.img.old -> boot/initrd.img
drwxr-xr-x 22 root root 4096 11月 14 16:12 lib
drwxr-xr-x 2 root root 4096 11月 20 21:52 lib64
drwx----- 2 root root 16384 5月 23 2020 lost+found
drwxr-xr-x 4 root root 4096 11月 14 16:10 media
drwxr-xr-x 3 root root 4096 5月 24 2020 mnt

```

```

root@sunzy-virtual-machine:/home# ls -lR www/
www/:
总用量 4
-rw-r--r-- 1 root root 20 3月 10 23:20 index.php

```

- 3.cat 该命令用于查看文本文件的内容, 后接要查看的文件名, 通常可用管道与more和less一起使用

```

1 cat text | less # 查看text文件中的内容
2 # 注: 这条命令也可以使用less text来代替
3 # cat text | more

```

```

root@sunzy-virtual-machine:/www/src/application/admin/controller# cat Config.php | more
<?php
// +-----+
// | Yzncms [ 御宅男工作室 ]
// +-----+
// | Copyright (c) 2018 http://yzncms.com All rights reserved.
// +-----+
// | Licensed ( http://www.apache.org/licenses/LICENSE-2.0 )
// +-----+
// | Author: 御宅男 <530765310@qq.com>
// +-----+

// +-----+
// | 系统配置
// +-----+
namespace app\admin\controller;

use app\admin\model\Config as ConfigModel;
use app\common\controller\Adminbase;
use think\Db;
--更多--

```

- 4.more 读取文件内容 一页一页翻动

```

1 more 文件
2
3 //可以进行的操作如下
4 空白键 (space): 代表向下翻一页;
5 Enter       : 代表向下翻『一行』;
6 /字串       : 代表在这个显示的内容当中, 向下搜寻『字串』这个关键字;
7 :f          : 立刻显示出档名以及目前显示的行数;
8 q           : 代表立刻离开 more , 不再显示该文件内容。
9 b 或 [ctrl]-b : 代表来回翻页, 不过这动作只对文件有用, 对管线无用。

```

- 5.cp 用于复制文件, copy之意, 它还可以把多个文件一次性地复制到一个目录下

```

1 -a : 将文件的特性一起复制
2 -p : 连同文件的属性一起复制, 而非使用默认方式, 与-a相似, 常用于备份
3 -i : 若目标文件已经存在时, 在覆盖时会先询问操作的进行
4 -r : 递归持续复制, 用于目录的复制行为
5 -u : 目标文件与源文件有差异时才会复制

```

```

1 cp -a file1 file2 #连同文件的所有特性把文件file1复制成文件file2
2 cp file1 file2 file3 dir #把文件file1、file2、file3复制到目录dir中

```

```

root@sunzy-virtual-machine:~/test# mkdir test1
root@sunzy-virtual-machine:~/test# cp 1 ./test1/
root@sunzy-virtual-machine:~/test# cd test1/
root@sunzy-virtual-machine:~/test/test1# ls
1

```

```

root@sunzy-virtual-machine:~/test# cp -a 2 3 ./test1/
root@sunzy-virtual-machine:~/test# cd test1/
root@sunzy-virtual-machine:~/test/test1# ls
1 2 3

```

- 6.rm 用于删除文件或目录, remove之意

```
1 -f : 就是force的意思, 忽略不存在的文件, 不会出现警告消息
2 -i : 互动模式, 在删除前会询问用户是否操作
3 -r : 递归删除, 最常用于目录删除, 它是一个非常危险的参数
4
5 rm -i file # 删除文件file, 在删除之前会询问是否进行该操作
6 rm -fr dir # 强制删除目录dir中的所有文件
```

```
root@sunzy-virtual-machine:~/test/test1# rm -i 1
rm: 是否删除普通空文件 '1'?
root@sunzy-virtual-machine:~/test/test1# ls
1 2 3
root@sunzy-virtual-machine:~/test/test1# rm -i 1
rm: 是否删除普通空文件 '1'? yes
root@sunzy-virtual-machine:~/test/test1# ls
2 3
```

```
root@sunzy-virtual-machine:~/test/test1# ls
2 3
root@sunzy-virtual-machine:~/test/test1# rm -f ./*
root@sunzy-virtual-machine:~/test/test1# ls
```

- 7.mv 该命令用于移动文件、目录或更名, move之意, 该命令跟cp命令很相似

```
1 -f : force强制的意思, 如果目标文件已经存在, 不会询问而直接覆盖
2 -i : 若目标文件已经存在, 就会询问是否覆盖
3 -u : 若目标文件已经存在, 且比目标文件新, 才会更新
```

注: 该命令可以把一个文件或多个文件一次移动一个文件夹中, 但是最后一个目标文件一定要是“目录”

```
1 mv file1 file2 file3 dir # 把文件file1、file2、file3移动到目录dir中
2 mv file1 file2 # 把文件file1重命名为file2
```

- 8.file 该命令用于判断接在file命令后的文件的基本数据

```
1 file filename
2 #例如:
3 file ./test
```

```
root@sunzy-virtual-machine:/www/src# file build.php
build.php: PHP script, UTF-8 Unicode text
```

- 9.tar 该命令用于对文件进行打包, 默认情况并不会压缩, 如果指定了相应的参数, 它还会调用相应的压缩程序(如gzip和bzip等)进行压缩和解压

```
1 -c : 新建打包文件
2 -t : 查看打包文件的内容含有哪些文件名
3 -x : 解打包或解压缩的功能, 可以搭配-C (大写) 指定解压的目录, 注意-c, -t, -x不能同时出现在同一条命令中
4 -j : 通过bzip2的支持进行压缩/解压缩
5 -z : 通过gzip的支持进行压缩/解压缩
6 -v : 在压缩/解压缩过程中, 将正在处理的文件名显示出来
7 -f filename : filename为要处理的文件
8 -C dir : 指定压缩/解压缩的目录dir
```

- 10.chmod 该命令用于改变文件的权限

```
1 chmod [-R] xyz 文件或目录
2 -R: 进行递归的持续更改, 即连同子目录下的所有文件都会更改
```

同时, chmod还可以使用u (user)、g (group)、o (other)、a (all) 和+ (加入)、- (删除)、= (设置) 跟rwx搭配来对文件的权限进行更改。

例如:

```
1 chmod 0755 file # 把file的文件权限改变为-rxwr-xr-x
2 chmod g+w file # 向file的文件权限中加入用户组可写权限
```

- 11.mkdir 创建文件夹

```
1 -m : 配置文件的权限喔! 直接配置, 不需要看默认权限 (umask) 的脸色~
2 -p : 帮助你直接将所需要的目录(包含上一级目录)递归创建起来!
```

- 12.rmdir 删除一个空的文件夹

```
1 -p : 连同上一级『空的』目录也一起删除
```

- 13.pwd Print Working Directory 的缩写, 也就是显示目前所在目录的命令

```
1 pwd [-P]
2 // -P 显示出确实的路径, 而非使用连结 (link) 路径
```

```
root@sunzy-virtual-machine:~/test/test1# pwd -P
/root/test/test1
```

- 14.find 是一个基于查找的功能非常强大的命令, 相对而言, 它的使用也相对较为复杂, 参数也比较多

```
1 find [PATH] [option] [action]
2
3 # 与时间有关的参数:
4 -mtime n : n为数字, 意思为在n天之前的“一天内”被更改过的文件;
5 -mtime +n : 列出在n日之前 (不含n天本身) 被更改过的文件名;
```



```

6  -mtime -n : 列出在n天之内（含n天本身）被更改过的文件名；
7
8  # 例如：
9  find /root -mtime 0 # 在当前目录下查找今天之内有改动的文件
10
11 # 与用户或用户组名有关的参数：
12 -user name : 列出文件所有者为name的文件
13 -group name : 列出文件所属用户组为name的文件
14 -uid n : 列出文件所有者为用户ID为n的文件
15 -gid n : 列出文件所属用户组为用户组ID为n的文件
16 # 例如：
17 find /home/ljianhui -user ljianhui # 在目录/home/ljianhui中找出所有者为
    ljianhui的文件
18
19 # 与文件权限及名称有关的参数：
20 -name filename : 找出文件名为filename的文件
21 -size [+ -]SIZE : 找出比SIZE还要大（+）或小（-）的文件
22 -type TYPE : 查找文件的类型为TYPE的文件，TYPE的值主要有：一般文件（f）、设备文件
    （b、c）、
23             目录（d）、连接文件（l）、socket（s）、FIFO管道文件（p）；
24 -perm mode : 查找文件权限刚好等于mode的文件，mode用数字表示，如0755；
25 -perm -mode : 查找文件权限必须要全部包括mode权限的文件，mode用数字表示
26 -perm +mode : 查找文件权限包含任一mode的权限的文件，mode用数字表示
27 # 例如：
28 find / -name passwd # 查找文件名为passwd的文件
29 find . -perm 0755 # 查找当前目录中文件权限的0755的文件
30 find . -size +12k # 查找当前目录中大于12KB的文件，注意c表示byte

```

```

root@sunzy-virtual-machine:/# find /root/ -mtime
/root/.config/nautilus
/root/.config/nautilus/desktop-metadata
/root/1
/root/.local/share
/root/.local/share/recently-used.xbel
/root/test/1
/root/test/2
/root/test/test1

```

- 15.ifconfig可设置网络设备的状态，或是显示目前的设置语法

```

1 ifconfig [网络设备][down up -allmulti -arp -promisc][add<地址>][del<地址>]
  [<hw<网络设备类型><硬件地址>][io_addr<I/O地址>][irq<IRQ地址>][media<网络媒介类
  型>][mem_start<内存地址>][metric<数目>][mtu<字节>][netmask<子网掩码>]
  [tunnel<地址>][-broadcast<地址>][-pointopoint<地址>][IP地址]
2
3
4 //参数
5 add<地址> 设置网络设备IPv6的IP地址。
6 del<地址> 删除网络设备IPv6的IP地址。
7 down 关闭指定的网络设备。
8 <hw<网络设备类型><硬件地址> 设置网络设备的类型与硬件地址。
9 netmask<子网掩码> 设置网络设备的子网掩码。
10 tunnel<地址> 建立IPv4与IPv6之间的隧道通信地址。
11 up 启动指定的网络设备。
12 [IP地址] 指定网络设备的IP地址。
13 [网络设备] 指定网络设备的名称。

```

- 16.ping 命令用于检测主机

```

1 ping [-dfnqrRV][-c<完成次数>][-i<间隔秒数>][-I<网络界面>][-l<前置载入>][-p<范
  本样式>][-s<数据包大小>][-t<存活数值>][主机名称或IP地址]
2
3 常用参数
4
5 -c <完成次数> 设置完成要求回应的次数。
6 -f 极限检测。
7 -i<间隔秒数> 指定收发信息的间隔时间。
8 -I<网络界面> 使用指定的网络接口送出数据包。
9 -l<前置载入> 设置在送出要求信息之前，先行发出的数据包。
10 -n 只输出数值。
11 -p<范本样式> 设置填满数据包的范本样式。

```

```

root@sunzy-virtual-machine:/# ping -c 4 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.020 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.037 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.033 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.030 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3
078ms
rtt min/avg/max/mdev = 0.020/0.030/0.037/0.006 ms

```

- 17.sudo 命令以系统管理者的身份执行指令

```

1 sudo [-v][-h][-l]
2 //
3 -v 显示版本编号
4 -h 会显示版本编号及指令的使用方式说明
5 -l 显示出自己（执行 sudo 的使用者）的权限
6 -v 因为 sudo 在第一次执行时或是在 N 分钟内没有执行（N 预设为五）会问密码，这个参数是
  重新做一次确认，如果超过 N 分钟，也会问密码
7 -k 将会强迫使用者在下次执行 sudo 时问密码（不论有没有超过 N 分钟）
8 -b 将要执行的指令放在背景执行

```

- 18.netstat 命令用于显示网络状态

```

1 netstat [-a][-apu][-s][-l]
2
3 //
4 -a 显示详细的网络状况
5 -apu 显示UDP端口号的使用情况
6 -s 显示UDP端口号的使用情况
7 -l 显示监听的套接口
8

```

```

激活Internet连接（服务器和已建立连接的）
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:ssh             0.0.0.0:*               LISTEN
tcp        0      0 localhost:ipp           0.0.0.0:*               LISTEN
tcp        0      0 localhost:34435         0.0.0.0:*               LISTEN
tcp        1      0 sunzy-virtual-mac:40232 10.2.5.251:801          CLOSE_WAIT
tcp        1      0 sunzy-virtual-mac:47094 10.2.5.251:http         CLOSE_WAIT
tcp        0      36 sunzy-virtual-machi:ssh 192.168.164.1:8472      ESTABLISHED
tcp        1      0 sunzy-virtual-mac:47098 10.2.5.251:http         CLOSE_WAIT
tcp        1      0 sunzy-virtual-mac:47090 10.2.5.251:http         CLOSE_WAIT
tcp        1      0 sunzy-virtual-mac:40234 10.2.5.251:801          CLOSE_WAIT
tcp        1      0 sunzy-virtual-mac:47092 10.2.5.251:http         CLOSE_WAIT
tcp        1      0 sunzy-virtual-mac:47088 10.2.5.251:http         CLOSE_WAIT
tcp        1      0 sunzy-virtual-mac:40220 10.2.5.251:801          CLOSE_WAIT
tcp        0      0 sunzy-virtual-machi:ssh 192.168.164.1:8846      ESTABLISHED
tcp        1      0 sunzy-virtual-mac:47096 10.2.5.251:http         CLOSE_WAIT
tcp        1      0 sunzy-virtual-mac:40236 10.2.5.251:801          CLOSE_WAIT

```

- 19. grep 用于分析一行的信息，若当中有我们所需要的信息，就将该行显示出来，该命令通常与管道命令一起使用，用于对一些命令的输出进行筛选加工等等

```

1 grep [-cinv] [--color=auto] '查找字符串' filename
2
3 //
4 -c : 计算找到‘查找字符串’的次数
5 -i : 忽略大小写的区别，即把大小写视为相同
6 -v : 反向选择，即显示出没有‘查找字符串’内容的那一行
7 # 例如：
8 # 取出文件/etc/man.config中包含MANPATH的行，并把找到的关键字加上颜色
9 grep --color=auto 'MANPATH' /etc/man.config
10 # 把ls -l的输出中包含字母file（不区分大小写）的内容输出
11 ls -l | grep -i file

```

```

root@sunzy-virtual-machine:~/test# grep -ci 'friend' hello.txt
2

```

```

root@sunzy-virtual-machine:~/test# grep -i 'friend' hello.txt
true friend is someone who reaches for your hand and touches your heart.
How many people actually have 8 true friends?Hardly anyone I know.But some of us have all right friends
and good friends.

```

- 20.ps 用于将某个时间点的进程运行情况选取下来并输出，process之意

```

1 ps [-A][-a][-u][-x][-l]
2
3 ///
4 -A : 所有的进程均显示出来
5 -a : 不与terminal有关的所有进程
6 -u : 有效用户的相关进程

```

```
7  -x : 一般与a参数一起使用，可列出较完整的信息
8  -l : 较长，较详细地将PID的信息列出
9  组合参数
10 //
11 ps aux # 查看系统所有的进程数据
12 ps ax # 查看不与terminal有关的所有进程
13 ps -lA # 查看系统所有的进程数据
14 ps axjf # 查看连同一部分进程树状态
```

- 21.httpd 命令是Apache HTTP服务器程序

- 22.shutdown

shutdown用于关闭计算机，而shutdown -r用于重启计算机相当于reboot

- 23 .whereis 命令用于查找文件

```
1  whereis [-bfmsu][-B <目录>...][-M <目录>...][-S <目录>...][文件...]
2
3  //
4  -b    只查找二进制文件。
5  -B<目录> 只在设置的目录下查找二进制文件。
6  -f    不显示文件名前的路径名称。
7  -m    只查找说明文件。
8  -M<目录> 只在设置的目录下查找说明文件。
9  -s    只查找原始代码文件。
10 -S<目录> 只在设置的目录下查找原始代码文件。
11 -u    查找不包含指定类型的文件。
```