

Linux的常用指令

一、指令与选项

1. 概述

- 终端中输入的内容为指令
- 指令格式: #指令 [选项] [操作对象]
- 一个指令可以包含多个选项、操作对象也可以是多个

二、基础指令

1. **ls**指令: (list) 列出所在的目录下的所有目录和文件的名称

- #ls 路径
- 路径可分为**相对路径**和**绝对路径**，相对路径需要一个参照物，一般为单签的工作路径，绝对路径不需要参照物，直接从“/”(表示根目录下)开始寻找对应路径
- **相对路径**: 在相对路径中通常用到“.”/“(表示当前目录下)”和“../”(表示上一级目录下)
- #ls 选项 路径
- #ls -l 路径: 表示以详细列表的形式进行展示

```
ysw@ysw--Linux:~$ ls -l
总用量 52
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 公共的
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 模板
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 视频
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 图片
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 文档
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 下载
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 音乐
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 桌面
```

- #ls -la 路径: 表示显示所有文件/文件夹(包含隐藏文件)

```
ysw@ysw--Linux:~$ ls -la
总用量 108
drwxr-xr-x 17 ysw ysw 4096 3月 16 12:42 .
drwxr-xr-x 3 root root 4096 3月 4 11:35 ..
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 公共的
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 模板
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 视频
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 图片
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 文档
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 下载
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 音乐
drwxr-xr-x 2 ysw ysw 4096 3月 4 12:33 桌面
-rw-r--r-- 1 ysw ysw 945 3月 16 09:15 .bash_history
-rw-r--r-- 1 ysw ysw 220 3月 4 11:35 .bash_logout
-rw-r--r-- 1 ysw ysw 3771 3月 4 11:35 .bashrc
drwxr-xr-x 17 ysw ysw 4096 3月 5 13:24 .cache
```

- 其中以“-”开头的表示对应文件类型为**文件**，“d”开头的表示文件类型为**文件夹**，以“.”开头的为隐藏文件

- o #ls -lh 路径：列出指定路径下的所有文件/文件夹的名称，以列表的形式并且显示文档的大小以可读性较高的形式显示

```
ysw@ysw--Linux:~$ ls -lh
总用量 52K
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 公共的
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 模板
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 视频
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 图片
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 文档
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 下载
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 音乐
drwxr-xr-x 2 ysw ysw 4.0K 3月 4 12:33 桌面
-rw-r--r-- 1 ysw ysw 8.8K 3月 4 11:35 examples.desktop
```

2. **pwd**指令：打印当前所在的路径
3. **cd**(change directory)命令：用于切换当前工作目录
4. **mkdir**(make directory)：创建目录。**语法**：mkdir 路径(可以是文件名，也可以是包含文件名的完整路径)
mkdir -p 路径：可以一次创建多层目录
5. **touch**指令：创建文件。**语法**：touch 文件名或包含文件名的完整路径
6. **cp**指令：复制文件或复制文件夹到指定位置。**语法**：cp 被复制的文件路径 文件被复制到的路径
注意：使用cp命令进行文件夹复制时需要添加参数“-r”，否则目录将被略过
7. **mv**指令：移动或剪切(可以给文件重命名)。**语法**：mv 需要移动的文件路径 需要保存的文件路径
8. **rm**命令：删除文档。**语法**：rm 选项 需要删除的文件路径
注意：rm -f 需要删除的文件路径 —>强制删除，不提示是否删除
rm -r 需要删除的目录 —>删除目录
rm -f Linux* —>使用通配符*，则会删除以Linux开头的所有文件
9. **vim**命令：vim是文本编辑器。**语法**：vim 文件路径 —>具体使用后序介绍

10. 输出重定向

一般命令输出会显示在终端，有些时候需要将一些命令的执行结果保存到文件中进行后序的分析，则需要使用输出重定向技术

语法：正常执行的指令 >/>> 文件路径 —>文件可以不存在(>：表示覆盖输出，会覆盖掉原先的文件内容；>>：覆盖输出，不会覆盖原始文件内容，会在原始内容末尾继续追加)

11. **cat**指令：可以直接打开一个文件。**语法**：cat 文件名

注意：cat命令可以配合**输出重定向**对文件进行合并。**语法**：cat 文件1 文件2 > 文件名

三、进阶指令

1. **df**命令：查看磁盘空间。**语法**：df -h —>会显示磁盘空间的详细信息

```
ysw@ysw-Linux:~$ df -h
文件系统      容量  已用  可用  已用% 挂载点
udev          960M    0    960M    0% /dev
tmpfs         197M    1.6M   196M    1% /run
/dev/sda1      20G    7.9G   11G   43% /
tmpfs          984M    0    984M    0% /dev/shm
```

2. **free**命令：查看内存使用情况。**语法**：free —>-m表示以mb为单位输出

	总计	已用	空闲	共享	缓冲/缓存	可用
内存:	2013324	949656	87400	2952	976268	874512
交换:	969960	3852	966108			

	总计	已用	空闲	共享	缓冲/缓存	可用
内存:	1966	927	85	2	953	854
交换:	947	3	943			

注意：交换表示临时内存，当系统真实内存不够用时会临时使用磁盘充当内存

- head**命令：查看一个文件的前n行，如果不指定则默认为前10行。**语法：**head -n 文件路径(n表示数字)
- tail**指令：查看一个文件的后n行语法和head类似，也可以查看文件的动态变化过程(变化内容不能是手动增加的)。**语法：**tail -f 文件路径
- less**指令：以较少的内容进行输出，按下辅助功能键查看更多(数字+回车、空格、上下方向键)。**语法：**less 需要查看的文件路径
- wc**指令：统计文件内容信息(包含行数、单词数、字节数)
 - l: 表示lines, 行数
 - w: 表示words, 单词数
 - c: 表示bytes, 字节数

- date**指令：表示操作时间和日期。**语法：**如下图

```

ysw@ysw-Linux:~$ date
2020年 04月 02日 星期四 10:56:15 CST
ysw@ysw-Linux:~$ date +%F
2020-04-02
ysw@ysw-Linux:~$ date +%F %T
2020-04-02 10:56:21
ysw@ysw-Linux:~$
  
```

注意：date -d "-1 day" "+%Y-%m-%d %H:%M:%S" 获取之前的某个时间，"-1 day"可以进行改变

- cal**指令：操作日历

```

ysw@ysw-Linux:~$ cal
      四月 2020
日 一 二 三 四 五 六
           1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30

ysw@ysw-Linux:~$ cal -3
      三月 2020      四月 2020      五月 2020
日 一 二 三 四 五 六 日 一 二 三 四 五 六 日 一 二 三 四 五 六
 1  2  3  4  5  6  7           1  2  3  4           1  2
 8  9 10 11 12 13 14       5  6  7  8  9 10 11       3  4  5  6  7  8  9
15 16 17 18 19 20 21     12 13 14 15 16 17 18     10 11 12 13 14 15 16
22 23 24 25 26 27 28     19 20 21 22 23 24 25     17 18 19 20 21 22 23
29 30 31                26 27 28 29 30                24 25 26 27 28 29 30
                                     31
  
```

注意：-3表示输出当前月和上一个月和下个月的日历

-y 年份 输出某一年的日历

- clear/ctrl+L**指令：清楚终端中已经存在的结果
- 管道：**| 作用："过滤"、"特殊"、"扩展处理"。**语法：**配合前面的指令使用

```
ysw@ysw-Linux:/$ ls / | grep s
lost+found
sbin
snap
srv
swapfile
sys
usr
ysw@ysw-Linux:/$
```

注意: grep的作用主要用于过滤, 管道前面的输出作为管道后面的输入, 即过滤出文件名含有s的文件

四、高级指令

1. **hostname**指令: 操作服务器的主机名(读取, 设置)
2. **id**指令: 查看一个用户的基本信息(包含用户id、用户组id.....), 该指令如果不指定用户则默认当前用户

```
ysw@ysw-Linux:/$ id
uid=1000(ysw) gid=1000(ysw) 组=1000(ysw),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),118(lpadmin),129(sambashare)
ysw@ysw-Linux:/$ id root
uid=0(root) gid=0(root) 组=0(root)
ysw@ysw-Linux:/$
```

3. **whoami**命令: 显式当前登录的用户, 用于shell脚本
4. **ps -ef**指令: 用于查看当前服务器进程信息。-e: 表示列出全部进程, -f: 表示显式全部的列

```
ysw@ysw-Linux:/$ ps -ef
UID          PID     PPID  C  STIME TTY          TIME CMD
root          1         0  0   10:01 ?        00:00:03 /sbin/init splash
root          2         0  0   10:01 ?        00:00:00 [kthreadd]
root          3         2  0   10:01 ?        00:00:00 [rcu_gp]
root          4         2  0   10:01 ?        00:00:00 [rcu_par_gp]
root          6         2  0   10:01 ?        00:00:00 [kworker/0:0H-kblockd]
```

注意: UID: 用户id; PID: 进程id; PPID: 父进程id; C: CPU的占用率; STIME: 进程的启动时间; TTY: 发起该进程的设备识别符号; TIME: 进程的执行时间; CMD: 该进程的名称或路径

当找不到父进程时表示此进程为僵尸进程

5. **top**命令: 查看服务器进程占用的资源

```
ysw@ysw-Linux:/$ top
top - 12:16:46 up 2:15, 1 user, load average: 0.25, 0.25, 0.20
任务: 279 total, 2 running, 277 sleeping, 0 stopped, 0 zombie
%Cpu(s): 4.2 us, 3.4 sy, 0.0 ni, 92.3 id, 0.2 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1966.1 total, 98.2 free, 935.1 used, 932.8 buff/cache
MiB Swap: 947.2 total, 942.5 free, 4.8 used. 848.3 avail Mem

 进程号 USER      PR  NI   VIRT   RES   SHR  %CPU  %MEM   TIME+ COMMAND
    634 root        20   0 1147536 31556 13920 S   13.0   1.6 18:21.07 snapd
    260 root        20   0      0      0      0 S    0.0   0.0  0:00:00 ibd2 (sd
```

表头含义: PR: 优先级, VIRT: 虚拟内存(程序申请的内存), RES: 常驻内存(实际使用的内存), SHR: 共享内存; 实际使用内存=常驻内存(RES)-共享内存(SHR)。S: 表示进程状态(S表示睡眠, R表示运行); MEM: 表示内存占用百分比; COMMAND: 表示进程名称或路径

6. **du -sh**指令: 查看目录的真实大小。语法: du -sh 目录路径

```
ysw@ysw-Linux:/$ sudo du -sh /home/
[sudo] ysw 的密码:
504M    /home/
ysw@ysw-Linux:/$
```

7. **find**指令：用于查找文件。**语法**：find 路径范围 [选项] 选项值 (选项：-name：按照文档名进行搜索(支持模糊搜索)，-type：按照文档类型进行搜索)
8. **service**指令：控制一些软件服务启动/停止/重启。**语法**：service 服务器名 start/stop/restart
9. **kill**命令：杀死进程。**语法**：kill 进程PID, killall 进程名称
10. **ifconfig**命令：用于操作网卡
11. **reboot**命令：重启计算机
12. **shutdown**命令：关机指令。**语法**：shutdown -h now 或者 shutdown -h [时间]
13. **uptime**命令：输出计算机的在线时间

```
ysw@ysw-Linux:~/桌面$ uptime
14:19:52 up 4:18, 1 user, load average: 0.47, 0.61, 0.51
```

14. **uname**命令：获取系统的类型。**语法**：uname -a 表示获取系统的全部信息
15. **netstat -tnlp**指令：查看网络连接状态

```
root@ysw-Linux:/home/ysw/桌面# netstat -tnlp
激活Internet连接 (仅服务器)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0 127.0.0.53:53          0.0.0.0:*               LISTEN
520/systemd-resolve
tcp        0      0 127.0.0.1:631          0.0.0.0:*               LISTEN
683/cupsd
tcp6       0      0 :::1:631               :::*                    LISTEN
683/cupsd
```

说明：-t：表示列出tcp协议的连接

-n：表示将地址从字母组合转换成ip地址，将协议转为端口号显式

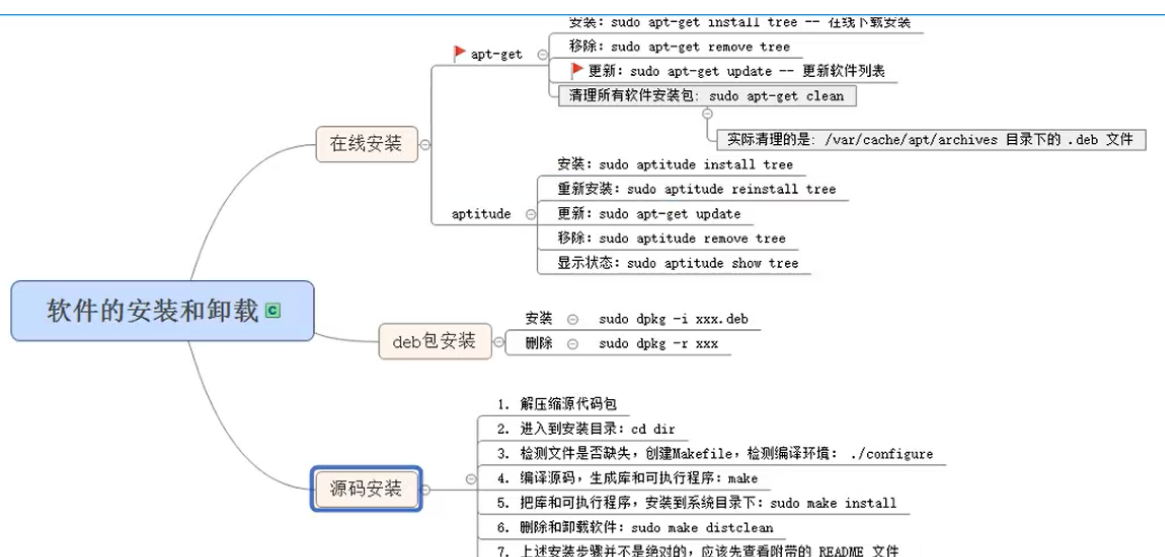
-l：表示过滤出“state”列中值为LISTEN(监听)的连接

-p：表示显式发起连接的进程pid和进程名称

16. **man**指令：手册(包含了Linux中全部命令手册)。**语法**：man 指令名(按q键退出)

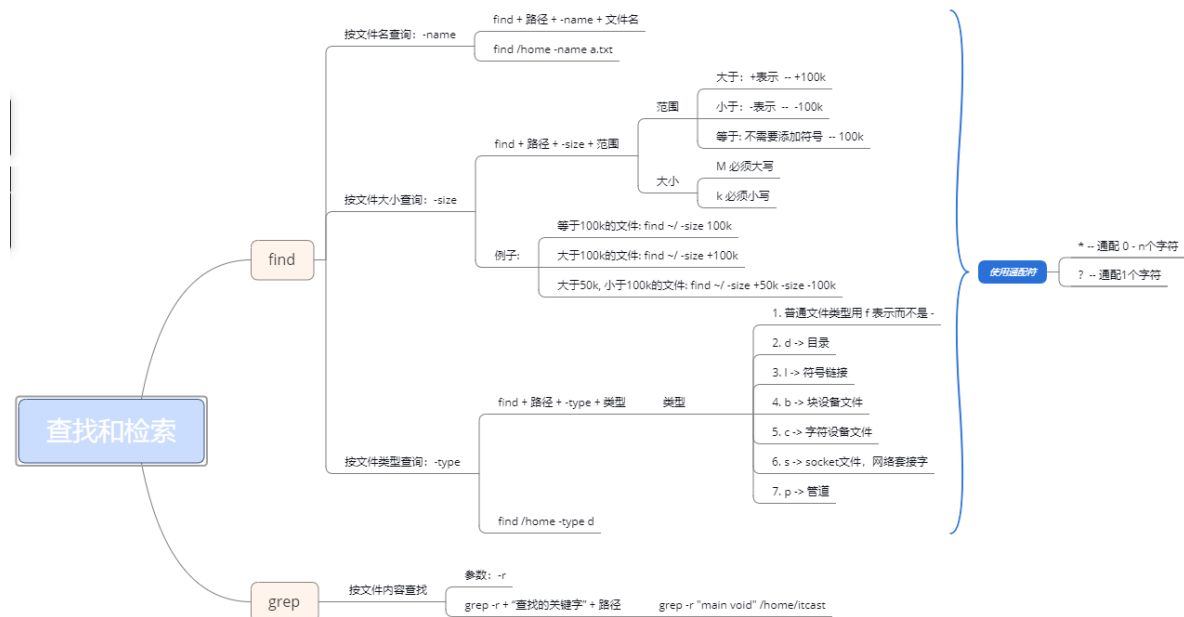
五、其他

1. 软件安装



2. 文件检索

- `grep -r "查找的关键字" + 路径`
- 例如: `grep -r "main void" /home/ysw`



3. U盘挂载

- 系统默认挂载的目录: `/media`
- 手动挂载: `/mnt`
- 挂载方式: `mount+设备名+挂载目录`:
 - 设备名: `sudo fdisk -l`
 - 挂载目录: 默认手动挂载到: `/mnt`
- 卸载: `sudo umount /mnt`
- 查找设备名

设备	启动	起点	末尾	扇区	大小	Id	类型
<code>/dev/sdb1</code>	*	614400	62463999	61849600	29.5G	7	HPFS/NTFS/exFAT
<code>/dev/sdb2</code>		62464000	63074302	610303	298M	6	FAT16

- 挂载

```
ysw@ysw-virtual-machine:/$ sudo mount /dev/sdb2 /mnt/
ysw@ysw-virtual-machine:/$ tree /mnt/
```

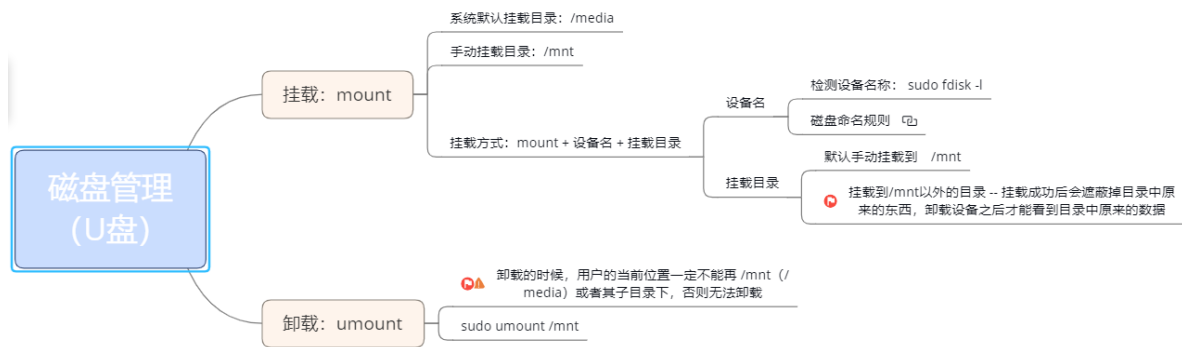
```
/mnt/
├── EFI
│   └── BOOT
```

- 卸载

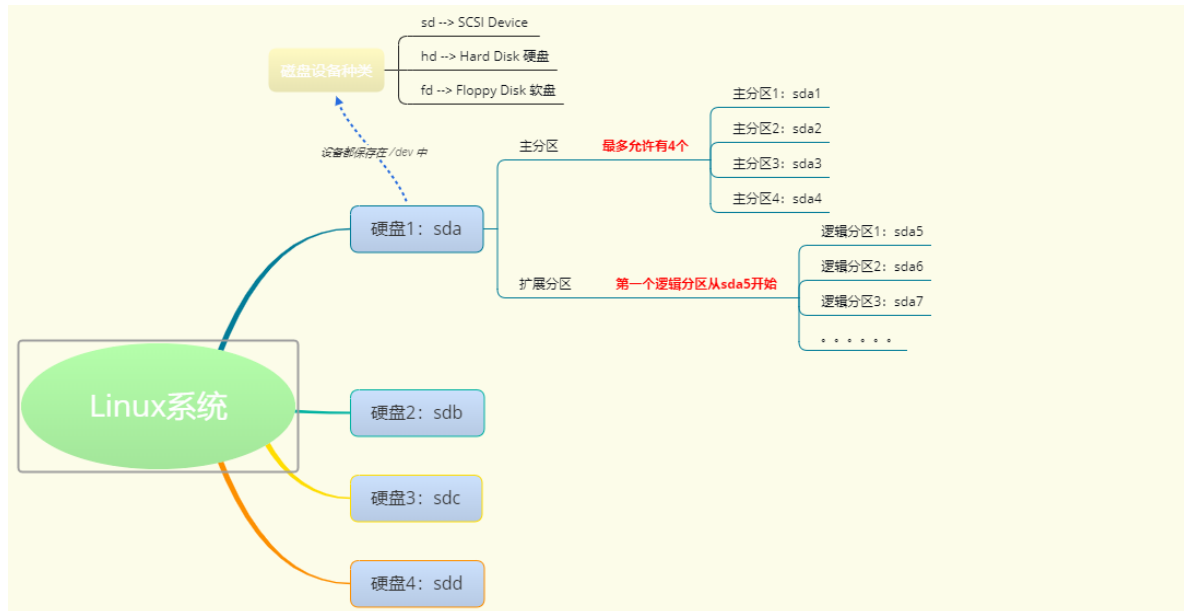
```
ysw@ysw-virtual-machine:/$ sudo umount /mnt
ysw@ysw-virtual-machine:/$ tree /mnt/
/mnt/
└── hgfs
```

1 directory, 0 files

- 思维导图



• 磁盘命名



4. 压缩包管理

1. gzip和gunzip:

- 压缩和解压, 不会保留源文件, 也不会打包, 只会在原文件基础上进行压缩

2. tar: 不使用z/j参数则该命令只能打包

- **参数:** c--创建; x--释放; v--显式提示信息; f--指定压缩文件的名字; z--使用gzip的方式压缩文件后缀为.gz; j--使用bzip2方式压缩后缀为.bz2
- **压缩:** tar zcvf 生成的压缩包名字(xxx.tar.gz) 要压缩的文件或目录
tar jcvf 生成的压缩包名字(xxx.tar.bz2) 要压缩的文件或目录
- **解压:** 若文件后缀为.gz: tar zxvf 压缩包名 -C 解压后的位置

```

ysw@ysw-virtual-machine:~$ tar zcvf study.tar.gz studySell/
studySell/
studySell/for.sh
studySell/out.txt
studySell/fun.sh
studySell/file2
studySell/while.sh
studySell/test.sh
studySell/test
studySell/file1
ysw@ysw-virtual-machine:~$ ls
公共的  视频  文档  音乐  snap  examples.desktop  python  studySell
模板    图片  下载  桌面
ysw@ysw-virtual-machine:~$ tar zxvf study.tar.gz
studySell/
studySell/for.sh
studySell/out.txt
studySell/fun.sh
studySell/file2
studySell/while.sh
studySell/test.sh
studySell/test
studySell/file1
  
```

3. rar(必须安装该软件):

- **参数**: 压缩: a; 解压缩: x
- **压缩**: rar a 生成的压缩文件名(temp) 压缩和文件或目录
- **解压缩**: rar x 压缩文件名 (解压缩目录)

4. zip:

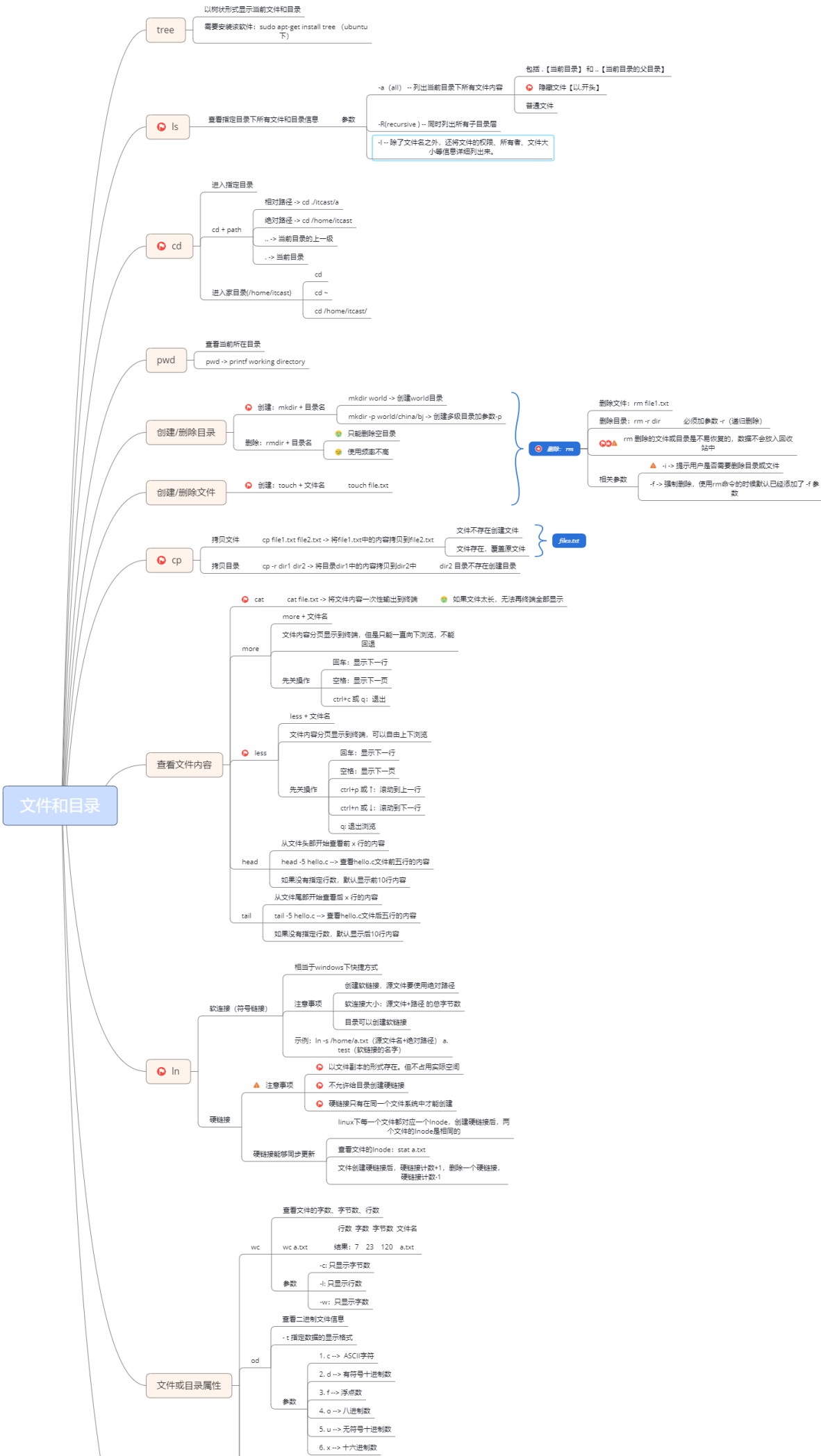
- **压缩**: zip 生成的名字 压缩的文件或目录
- **解压缩**: unzip 压缩包名字 -d 解压目录

5. 其他

- 其他命令可查看手册[Linux命令手册](#)

六、思维导图

- 文件和目录操作





• 压缩包管理

