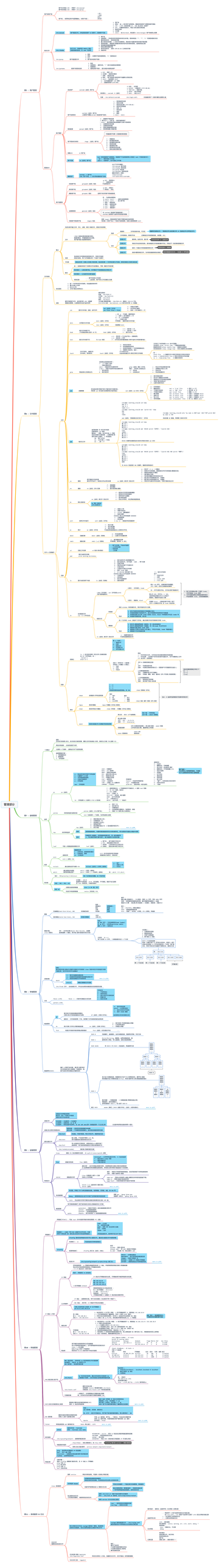

管理部分

章5 ：用户管理

- 用户和用户组
  - 用户名与相应 UID ：存储于 /etc/passwd
  - 用户组与相应 GID ：存储于 /etc/group
  - 用户组 ：相同特征用户的逻辑集合，与用户关系 ：
    - 一对一
    - 一对多
    - 多对一
    - 多对多
- 相关文件
  - **/etc/passwd** — 用户配置文件，亦包括系统用户 & 伪用户，任意用户可读
    1. 用户名
    2. 密码 ：由 x 表示用户设有密码，删除则导致用户无需密码即可登陆
    3. UID ：1~499 为系统保留 ID ，500 及以后为一般用户
    4. GID ：所属的初始组ID，再加入其它组则为附加组
    5. 描述性信息
    6. 主目录
    7. shell ：如/bin/bash，若设置为 /sbin/nologin 用户则被禁止登录
  - **/etc/shadow** — 影子文件，存储采用 SHA512 算法加密的真实密码，仅 root 可读写
    1. 用户名
    2. 加密密码 ：手动修改时将导致密码无法识别，密码串前加 "!"、"*"、"x" 可使密码暂时失效
    3. 最后修改时间 ：修改密码的时间
    4. 最小修改时间间隔 ：0 表示可任意修改；其余表示与最后修改时间的天数间隔
    5. 密码有效期 ：强制用户最后修改时间多长时间修改密码，或者密码会过期
    6. 密码需变更提前提醒天数
    7. 密码过期后宽限天数
    8. 账号失效时间 ：使用 1970-01-01 以来的日天数
    9. 保留字段
  - /etc/group — 用户组配置文件
    1. 组名
    2. 密码 ：主要用于指定组管理员，"x" 为密码标识
    3. GID
    4. 用户组中的用户列表
  - /etc/gshadow — 组用户的密码信息
    1. 组名
    2. 加密密码 ：通常为空，"!" 表示无组密码及管理员
    3. 组管理员
    4. 组附加用户列表 ：显示该组中的附加用户
- 管理命令
  - 用户管理
    - 添加用户 — useradd [选项] 用户名
      - -u UID ：指定 UID
      - -d 主目录 ：指定绝对路径
      - -c 说明 ：描述性内容
      - -g 组名 ：用户初始组(指定组后将不创建默认初始组)
      - -s shell ：指定登录 shell
      - -o ：允许创建的用户 UID 相同
      - -r ：创建系统用户(UID<500)
      - 修改默认 ：useradd -D [选项]
        - -b HOME ：主目录
        - -e EXPIRE ：过期时间 (YYYY-MM-DD)
        - -f INACTIVE ：宽限天数
        - -g GROUP ：默认初始组
        - -s SHELL ：默认shell
      - 位置 ：/etc/default/useradd　/etc/login.defs　对创建的用户一些基本属性设置默认值
    - 用户信息修改 — usermod [选项] 用户名 (主要用于调整useradd 添加的用户信息)
      - -c ：修改描述性信息
      - -d ：修改主目录
      - -e ：修改失效日期 (YYYY-MM-DD)
      - -g ：修改所在初始组
      - -u ：修改 ID
      - -l ：修改名称
      - -L ：锁定
      - -U ：解锁
      - -s shell ：shell
    - 密码配置 — passwd [选项] 用户名
      - -S ：查看用户密码信息
      - -l 用户名 & -u 用户名 ：锁定及解锁用户
      - --stdin ：通过管道将输出的数据作为用户初始密码
      - -i ：设置密码失效日期
      - -x ：设置密码有效期
      - -w ：密码过期前警告天数
      - -n ：修改密码最小间隔天数
    - 用户密码状态修改 — chage [选项] 用户名
      - 可强制用户在第一次登录时修改密码
      - -l ：详细状态
      - -d ：修改最后修改时间
      - -m ：密码最短保留天数
      - -M ：密码有效期
      - -W ：提前警告天数
      - -E ：修改账号失效日期
      - -i ：密码过期后宽限天数
    - 查看UID — id 用户名
    - 用户切换 — **su [选项] 用户名**
      - root 切至普通用户无需密码，普通用户切换至 root (环境变量未切)
      - - 或-l ：同时切换工作环境 & 用户
      - -p或-m ：切换用户，但不切换工作环境
    - 删除用户 — **userdel -r 用户名** root 用户可用，-r 表示同时删除用户目录
      - 从以下文件中删除：
        - /etc/passwd 基本信息
        - /etc/shadow 密码信息
        - /etc/group 组信息
        - /etc/gshadow 组密码信息
        - 删除主目录 & 邮箱 ：/home/username　/var/spool/mail/username
  - 用户组管理
    - 添加用户组 — groupadd [选项] 组名
      - -g GID ：指定组ID
      - -r ：创建系统组
      - 不加参数亦可
    - 修改用户组 — groupmod [选项] 组名
      - -g GID ：改组ID
      - -n 新组名 ：改名
    - 删除用户组 — groupdel 组名　适用于非任何用户初始组的组
    - 设置管理员 — gpasswd [选项] 组名
      - 空 ：为该组设置密码
      - -A user1, user2... ：设置为该组管理员
      - -M user1, user2... ：加入用户组
      - -r 组名 ：移除密码
      - -R 组名 ：使密码失效
      - -a user ：添加用户
      - -d user ：移除用户
      - 与 usermod 添加用户至组的区别：usermod 会将用户从原本所在的组中移除
    - 切换用户初始用户组 — newgrp 组名
      - 使用 newgrp 切换组使用户获取不同资源加组对应的权限
      - 每切换一次用户均以一个新的子进程登录，回复父进程需使用 exit

章6 ：文件管理

- 文件系统 — 负责为用户建立文件、存入、读取、修改、转储文件，控制文件的存取
  - 概念
    - 文件 — LINUX 系统处理信息的基本单位；数据的集合，存储数据的命名对象；用户眼中最小的逻辑存储单元
      - 组成
        - 数据项 — 文件的实际内容，可为空 — 根据有无结构分为 ：有结构文件(如目录文件) & 无结构文件(字符流文件)
        - 文件控制块，即索引节点 — 记录相应文件的控制信息，如权限、大小
      - 划分
        - 普通文件 — 最常用，包括文档、图片等 — 分为文本文件 & 二进制文件
        - 目录文件 — 存放文件名及相关信息，是内核组织文件系统的基本节点，可包含子目录或普通文件
        - 链接文件 — 指向一个真实文件的链接的特殊文件 — 包括软连接 & 硬链接
        - 设备文件 — 系统中最特殊的文件，为外部设备提供同意的标准接口 — 据访问数据方式分为 ：块设备 & 字符设备
    - 目录 — 包含在很多文件控制项目的特殊文件，支持文件系统的层次结构，每个目录都记在一个或多个目录中
    - 子目录 — 被包含在另一目录(父目录)中的目录，除去根目录 / 外所有目录均有父目录且为其自身
    - 路径
      - 由 / 连接起来的多个目录和文件名的集合，可唯一确定文件或目录
      - 绝对路径 ：以根目录开始，任何情况下可使用其指定所需文件
      - 相对路径 ：以当前所在目录为起始
      - 特殊目录
        - . ：用户所在的工作目录
        - .. ：上级目录，用户所在工作目录的父目录
  - 命名规则
    1. 除"/"外所有字符均可使用，但应避免特殊字符
    2. 最大长度255字符
    3. 区分大小写
    4. 文件扩展名无特殊含义
  - 设备 — 硬件亦被视作文件，由内核中的 udev 设备管理器自动管理，以进程的形式运行监听信息
    - 硬件设备名
    - 软驱 ：dev/fd[0-1]
    - 打印机 ：dev/lp[0-15]
    - 光驱 ：dev/cdrom
    - 鼠标 ：dev/mouse
    - IDE设备(硬盘驱动器 ：/dev/hd[a-d]，硬盘以 /dev/sd 开头
    - SCSI/SATA/U盘 ：dev/sd[a-p]，a-p 表示主机最多可有 16 块硬盘
- 文件 & 目录操作
  - 文件
    - cat — 显示文本内容、连接、合并文件 — **cat [选项] 文件名　cat file1 file2 > file3**
      - -n ：对输出行编号
      - -b ：仅对非空行编号
      - -s ：连续 2 空行替换为 1 行
    - more — 分页显示文本内容
      - more [选项] 文件名
        - -c 或 -p ：不滚屏，清屏再显示
        - -s ：连续 2 空行替换为 1 行
        - +n ：从 n 行开始显示
        - -n ：一次显示行数
        - 不带参数 ：清屏并显示文件内容
      - less [选项] 文件名　功能兼容 more
    - head — 显示文件前若干行(不设置选项默认显示 10 行) — head [选项] 文件名
      - -n k ：显示前 k 行
      - -c k ：显示前 k 字节
      - -v ：显示文件名
    - tail — 显示文件末若干行 — 与 head 相反
      - -n k ：显示后 k 行(此处不加 n 效果相同)
      - -c k ：显示后 k 字节
      - -f ：文件变化后新增的数据(可用于监听文件的新增内容)
    - touch — 文件不存在，创建文件；文件存在，修改时间 — touch [选项] 文件名
      - -a ：修改访问时间
      - -m ：修改数据修改时间
      - -c ：修改时间参数(3 个时间参数都会修改)
      - -t ：需要修订的时间，格式 ：YYYYMMDDhhmm
      - 无选项则创建文件/修改已存在文件参数
      - 文件的三个时间参数(stat 命令可查看)：
        - 访问时间 atime, access time ：只要被读取即更新
        - 数据修改时间 mtime, modify time ：数据变化则更新
        - 状态修改时间 ctime, change time ：状态(权限等)变化则更新
      - 如 ：
        - touch file　# 创建文件/修改已存在的文件的访问时间
        - touch -t 202107191432 file # 修改已存在的文件的时间
        - ls -l --time=atime file　# 查看访问时间
    - grep — 筛选信息(正则表达式)
      - 正则表达式的特殊字符 ：
        - c* ：0 个或多个字符
        - . ：匹配任意单字符
        - [abc] ：匹配括号中任意单字符
        - [^abc] ：匹配不在括号中的任意单字符
        - ^ ：首行定位符
        - $ ：尾行定位符
      - grep [选项] 模式 文件名　模式 ：字符串/正则表达式
        - -E ：字符串
        - -F ：正则表达式
        - -c ：列出文件包含模式的行数
        - -i ：忽略模式中的大小写
        - -n ：列出行号
        - -v ：列出没匹配的行
    - **sed** — 流编辑器 — 将当前处理行存储于称为"模式空间"的缓冲区，然后用 sed 命令逐行处理存放区内容并输出 — sed [选项] sed 命令 文件名
      - -n ：显示匹配行
      - -e ：执行多个编辑命令时使用
      - -i ：直接修改文件而非输出到屏幕
      - -r ：支持扩展正则表达式
      - -f ：从脚本读取执行
      - -p ：打印匹配行
      - -d ：删除指定行
      - -a ：于匹配行后追加
      - -i ：于匹配行前追加
      - -c ：整行替换
      - -r ：读取文件
      - -w ：写入文件
      - -s ：字符串替换
      - sed -n '2,3p' file　# 显示 2~3 行
      - sed -n '10p' file　# 显示第 10 行
      - sed '3,5d' file　# 删除 3~5 行
      - sed '2a hello' file　# 在第 2 行后追加 hello
      - sed '2i hello' file　# 在第 2 行前插入 hello
      - sed '2c hello' file　# 将第 2 行替换为 hello
      - sed 's/tea/water/' file　# 将 tea 替换为 water
    - **awk** — 格式化文本
      - 由匹配规则 及 执行命令组成；匹配规则与 sed 相同；整体由 '' 括起，执行命令由 {} 括起；自动为匹配行分配变量 ：(任意空格/制表符分割，-F 手动指定)
        - $0 ：整个文本行
        - $1 ：文本行第 1 个字段
        - $n ：第 n 个
      - awk [选项] '匹配规则{执行命令}' 文件名

```
[nova@pc learning_linux]$ cat temp
ONE IS 1
TWO IS 2
THREE IS 3
[nova@pc learning_linux]$ awk '{print $1}' temp
ONE
TWO
THREE
```

```
[nova@pc learning_linux]$ echo "my name is BOB"|awk '{$4="TOM";print $0}'
my name is TOM
```

先给变量 $4 赋值，再用第二条命令打印

```
[nova@pc learning_linux]$ cat temp
第一行 1
第二行 2
第三行 3
[nova@pc learning_linux]$ awk 'BEGIN {print "文件行："}{print $0}'
temp
文件行：
第一行 1
第二行 2
第三行 3
```

BEGIN 关键字会强制先执行该命令再执行后续 awk 命令

```
[nova@pc learning_linux]$ cat temp
第一行 1
第二行 2
第三行 3
[nova@pc learning_linux]$ awk 'BEGIN {print "文件行："}{print $0} END {print "结束"}' temp
文件行：
第一行 1
第二行 2
第三行 3
结束
```

与 BEGIN 对应的是 END 关键字，强制在结束前执行

-
    - cp — 复制 — 用于复制文件和目录，源文件可为多个，则目标文件必须为目录 — cp [选项] 源文件 目标文件
      - -a ：相当于 -d -p -r 的集合
      - -d ：若源文件为软链接，复制的文件亦为软链接(硬链接无效)
      - -p ：复制后保留源文件属性
      - -r ：递归复制(目录)
      - -i ：询问是否覆盖已存在的目标文件
      - -l ：建立源文件的硬链接文件，非复制
      - -s ：建立源文件的软链接文件，非复制
      - -u ：目标文件与源文件对比差异，可用于更新目标文件
    - rm — 删除 — rm [选项] 文件/目录
      - -f ：强制删除
      - -r ：递归删除
      - -i ：提示是否确认删除
    - mv — 移动/重命名 — mv [选项] 源文件 目标文件；**mv 原名 重命名 文件/目录均可**
      - -f ：若目标文件存在则强制覆盖
      - -i ：若目标文件存在则提示
      - -n ：若存在则不会覆盖
      - -v ：显示移动进程
      - -u ：目标文件存在，对比两者保留更新者
    - sort — 排序文件内容行 — sort [选项] 文件名
      - -f ：忽略大小写
      - -b ：忽略行前空白
      - -u ：unique，删除重复行
      - -t ：指定分隔符
      - ####### 默认以字符串排序 #######
      - -n ：以数值排序
      - -r ：反向排序
      - -k [m, n] ：按指定字段范围排序
    - wc — 统计 — wc [选项] 文件名
      - -l ：统计行数
      - -w ：统计单词数
      - -m ：统计字符数
      - 不加选项则直接输出以上信息及文件名
  - 目录
    - mkdir — 创建目录 — mkdir [选项] 目录名
      - -m ：手动设置权限
      - -p ：以递归方式创建目录
    - rmdir — 删除目录 — rmdir [-p] 目录名
      - -p ：递归删除空目录　作用有限，更常用 rm 命令
    - cd — 切换工作目录 — cd 相对/绝对路径
      - ~ ：用户主目录，不加也可返回
      - - ：上次所在目录
      - . ：当前目录
      - .. ：父目录
    - pwd — 显示当前所在目录　print working dictionary
    - ls — 显示当前目录下内容 — ls [选项] 目录名
      - -a ：显示全部文件(包括隐藏文件)
      - -A ：显示全部文件，但不包括 . 以及 .. 两个目录
      - -d ：仅显示目录
      - -R ：同时列出目录下子目录文件
      - -l ：长格式列出
      - -h ：以易读方式显示文件体积
      - -i ：显示 inode 信息
      - -n ：以 UID/GID 显示用户/组
      - ####### 默认以文件名排序 #######
      - -r ：逆向排序
      - -S ：按文件容量排序
      - -t ：按时间排序
      - -F ：指示文件类型　如下 ：
        - * ：可运行文件
        - / ：目录
        - = ：socket 文件
        - | ：FIFO 文件
    - ln — 生成链接文件 (cp -l/s 亦可生成)
      - LINUX 文件系统 ：ext 文件系统(ext4) (暂不提供超级块)
        - 小部分 ：文件索引，inode
          - 默认 128 字节，记录权限所有者属组、大小、时间、状态以及数据块的编号
          - 注 ：并不记录文件名，文件名记录于数据块中
        - 大部分 ：数据块，block
          - 默认为 4KB，亦可为1、2、3KB；实际存储会根据文件大小占用多个数据块，数据块可连续/分散
          - 目录的 block ：记录该目录下所有一级子文件 & 目录的文件名 及 inode 对应关系
        - 1. 每个文件单独占用一个索引 inode，文件内容由 inode 指向　2. 文件内容的读取必须借由目录中的文件名找到该 inode，从而找到数据块
      - 链接方式 — 类似 windows 中的快捷方式，指向文件/目录
        - 软链接 — 特点
          1. 源文件被修改则软链接文件数据随之变化
          2. 软链接被删除不影响源文件，源文件被删除则找不到数据
          3. 软链接会建立自己的 inode 信息及 block，block 中存储源文件名及 inode 号
          4. 软链接可链接目录，跨分区
        - 硬链接 — 给一个文件的 inode 分配多个文件名，通过任意文件名均可找到该文件 — 特点
          1. 源文件/硬链接随者其一被修改，另一者亦改变
          2. 源文件/硬链接随者其一被删除，另一者(inode 号)仍可访问
          3. 硬链接不建立新的 inode 信息
          4. 硬链接不可跨文件系统(分区)建立，不同文件系统，inode 号重新建立
          5. 硬链接不可访问链接目录
      - ln [选项] 源文件 目标文件
        - -s ：创建软链接文件
        - 不加则创建硬链接文件
  - 权限
    - 权限位 — ls -l 命令显示的第一列为文件/目录的权限，由 10 个字符组成，如 ：-rwxrwxr-x　drwxr-xr-x
      - 第 1 个字符
        - - ：普通文件
        - d ：目录
        - l ：链接文件
        - b ：块设备文件
        - c ：字符设备文件
      - 后续 9 字符针对 3 组权限 ：所有者 & 所属组 & 其他人
        - r ：可读
        - w ：可写
        - x ：可执行
      - 附加
        - s ：针对可执行文件，文件在执行期间拥有所有者权限
        - t ：针对目录 ：任何用户可在该目录中建立文件，但只可删除他人文件
        - s/t 会占用 x 的权限位
      - 文件 & 目录的权限位区别
        - r
          - 文件 ：可读取实际内容
          - 目录 ：可读取目录结构列表(仅 r 权限用户只可查看而无法进入)
        - w(目录的最高权限)
          - 文件 ：可编辑内容(但不可删除)
          - 目录 ：可新建、删除、移动、重命名 文件/子目录(无视其权限)
        - x(文件的最高权限)
          - 文件 ：被系统执行的权限
          - 目录 ：表示用户可进入目录
        - 因此目录的权限位只有三个　0(---)　5(r-x)　7(rwx)
    - 修改
      - chmod — 使用数字/符号变更权限
        - r ：4　w ：2　x ：1　各组的权限和组成权限值，如 765 — chmod 权限值 文件名
        - 身份：u ：用户　g ：组　o ：其它　a ：以上所有；操作：+ ：加入　- ：删除　= ：设定 — chmod 身份 操作 权限 文件/目录
        - 加入 -R 选项可连同更改子目录中所有文件
      - chgrp — 更改所属组 — chgrp 所属组 文件/目录名
      - chown — 更改所有者/所属组 — chown 所有者 ：所属组 文件/目录名
      - umask — 系统为新建文件/目录赋予的初始权限值
        - 默认值 ：0022 (4个0表的权限)　第1个数 ：特殊权限　后3个数 ：umask 权限值
        - 默认最大权限　文件 ：666 ：-rw-rw-rw-　目录 ：777 ：drwxrwxrwx
        - 权限的换算 — 文件/目录的初始权限 = 默认最大权限 - umask 权限　以字母形式进行计算，重者则公共权限
        - umask 权限修改 — umask 001　直接设定

章7 ：进程管理

- 一般概念
  - 进程 ：正在执行的程序/命令，执行时执行者的权限、属性以及代码会载入内存，系统为之分配 PID(进程 ID)
  - 常驻内存进程 ：启动后驻留于内存
  - 父进程 & 子进程 ：进程运行中产生的新进程
  - 进程管理的作用 ：
    1. 判断健康状态
    2. 查看进程
    3. "杀死"进程
- 监控
  - ps — ps [选项]
    - -a ：除会话引线外所有终端下所有进程
    - -u ：归属用户 & 内存
    - -x ：无终端控制的进程
    - -l ：长格式详细显示
    - -e ：所有进程
    - -aux ：所有进程
      - USER ：所属用户
      - %CPU ：CPU 使用率
      - %MEM ：物理内存使用率
      - VSZ ：虚拟内存占用(KB)
      - RSS ：物理内存占用(KB)
      - TTY ：所属运行终端
      - STAT ：进程状态
      - START ：启动时间
      - TIME ：占用的 CPU 运算时间
      - COMMAND ：产生命令
      - 终端种类　tty1~6 ：本地字符界面终端　tty7 ：本地图形终端　pts0~255 ：虚拟终端(一般为远程)
      - 进程状态
        - R ：运行
        - S ：睡眠状态，可被唤醒
        - T ：停止状态，后台暂停/调试
        - D ：睡眠状态，不可被唤醒(I/O)
        - Z ：僵尸进程
        - t ：处于跟踪
        - l ：多线程
        - + ：位于后台
        - < ：高优先级
        - N ：低优先级
        - s ：含有子进程
        - L ：锁入内存
        - Z ："僵尸"，已死但仍存在内存中
        - W ：内存交换无效
        - X ："死掉"
      - 僵尸进程 ：即正常终止/程序编写错误，子进程先于父进程结束而未被父进程回收
    - -le ：所有进程 & 父进程 & PID & 优先级
      - F ：进程权限标志，1 可被复制但不可被执行；4 使用 root 权限
      - S ：与 STAT 相同，表状态
      - C ：CPU 占用
      - PPID ：父进程 ID
      - PRI & NI ：优先级，越小越高
      - ADDR ：处于内存位置
      - SZ ：占用内存大小
      - WATCH ：运行与否，"-" 表运行
  - pstree — 树状结构显示进程关系 — pstree [PID/用户名]
    - -p ：显示 PID
    - -u ：显示用户
    - -c ：连续显示
    - -a ：启动进程的完整指令，包括路径、参数
    - -n ：根据 PID 排序(默认为名称排序)
    - init 为系统第 1 个进程，为所有进程父进程
  - top — 动态持续监听 — top [选项]
    - -d ：更新间隔(默认1秒)
    - -p ：仅查看指定 PID
    - -s ：安全模式(无交互)
    - -i ：不显示闲置/僵尸进程
    - -b ：批处理模式输出(与 -n 配合重定向输出到文件)
    - -n ：执行次数
    - more in p206
    - 缓存 — 读取硬盘数据时，将最常用的数据保存在内存缓存区中读取
    - 缓冲 — 向硬盘写入数据时，先把数据放到缓冲区，再一次性写入，将分散的写操作集中进行，减少磁盘碎片和硬盘的反复寻道
  - lsof — 可进一步查看进程调用的文件 — lsof [选项]
    - -c ：用户进程打开的文件
    - -p ：指定进程打开的文件
    - -u ：以字符开头的进程打开的文件
    - +d 目录 ：指定目录中被打开的文件
- 结束
  - kill — 进程信号
    - 最重要信号　1 ：立即关闭，读取配置后重启　9 ：立即关闭，强制　15 ：正常结束
    - kill [-信号] PID
    - kill 命令发送的信号有被进程识别才会执行，否则无效
  - killall — 通过进程名杀死不依赖 PID — **killall [-信号] [-选项] 进程名**
    - -u ：用户
    - -I ：忽略大小写
    - -i ：交互式
  - pkill — 通过进程名杀进程，按终端踢出用户
    - pkill [-信号] 进程名
    - pkill [-信号] [-t 终端] 进程名　如 ：pkill -9 -t pts/1
- 优先级
  - 表示 ：PRI(priority) & NI(nice)　PRI 为内核动态调整，NI 手动修改
  - NI范围 ：-20~19　普通用户可用范围 ：0~19，仅可调高，不可调低，限定于自己的进程　仅 root 可调为负，可调高/低
  - 关系 ：PRI = PRI + NI
  - 修改 NI
    - 给要启动的程序赋值 — nice [-n NI 值] 命令
    - 给运行中的进程赋值 — renice [优先级] PID

章8 ：存储管理

- 硬盘
  - 机械硬盘(Hard Disk Drive), HHD — 采用磁性碟片
    - 组成 ：磁盘盘片　磁头　主轴　传动轴
    - 逻辑结构 ：磁道　扇区　柱面
    - 细节 ：扇区(最小存储单元) ：大小固定，早期 512 字节，现为 4096 字节；柱面 ：相同编号的磁道形成的圆柱；磁盘大小计算 ：磁头数 × 柱面数 × 扇区数 × 扇区大小
    - 接口 ：IDE ：ATA/PATA，早期主要接口；SATA ：更高速度更强纠错；SCSI ：范围广，多任务，大带宽，CPU 占用低，热插拔
  - 固态硬盘(Solid Stat Disk), SSD — 采用闪存颗粒 — DRAM　闪存(主)
- 硬盘识别 & Linux 初始化
  - MBR ：主引导记录(Master Boot Record)，占据硬盘0磁道第一个扇区 ：用于载入硬盘的可执行代码
    - 构成 ：前 446 字节 ：引导加载程序(boot loader)　中 64 字节 ：分区表(partition table)　最后 2 字节 ：结束标志
    - 分区表 ：16 × 4　每个分区表 16 字节，一块硬盘最多划分 4 个分区
      - 分区意义 ：1. 方便控制与控制　2. 提高系统效率　3. 方便用户空间配额　4. 便于备份 & 恢复
      - 大于 4 个分区的创建 ：在第一个扩展分区中的 64 字节空间中，将其上一扇区记录分区信息的 16 字节空间指向另一个空间，即扩展分区，再扩展空间中可以继续分区
      - 例 ：/dev/sda
        - dev/sda1 — 第 1 个主分区
        - dev/sda2 — 第 2 个主分区
        - dev/sda3 — 第 3 个主分区
        - dev/sda5　dev/sda6 — 扩展分区
- 设备挂载
  - 挂载 ：通过指定目录(挂载点)将硬件设备的文件系统和 Linux 系统中的文件系统进行关联，已建立的、空文件夹均可作为挂载点
  - mount
    - -l ：显示已挂载的设备信息
    - -a ：检测是否有因此尚未挂载的设备文件
    - -t ：指定欲挂载的文件系统类型
    - more in p218
    - unmount　卸载已装载的文件系统
  - /etc/fstab 文件 ：自动挂载文件，开机自动读取该配置自动挂载指定设备
- 分区
  - fdisk (<2TB) — fdisk -l ：识别本机磁盘及分区信息
    - 交互命令　q ：不保存退出　w ：保存并退出　l ：显示已知文件系统信息　n ：新建分区　v ：验证分区表
  - parted (>2TB)
- 磁盘管理
  - **df** — 显示各文件系统的硬盘使用情况，主要从各文件系统的超级块中读取 — df [选项] [目录/文件名]
    - -a ：显示所有信息
    - -m/k/h ：以何种单位显示
    - -T ：显示分区的文件系统
    - -i ：显示 inode 数量
    - 超级块 ：文件系统的第一个块，保存整个文件系统的全部信息
  - du — 统计目录/文件所占用的磁盘容量 — du [选项] [目录/文件名]
    - -a ：显示目录/子目录的磁盘占用量
    - -h ：以合适单位显示
    - -s ：总磁盘占用量
    - 加目录则分析该目录的分区
  - fsck — 检查文件系统并尝试修复出现的错误 — fsck [选项] 分区设备文件名　...
- 磁盘阵列(RAID) — 廉价 & 具有冗余功能，通过软/硬件将多块分区组成为较大容量组，性能更好，数据不会因为单块硬盘出问题而丢失
  - RAID 0 — 性能最好，速度最快，由多块硬盘组成，数据等份存储，但无冗余
  - RAID 1 — Mirror/Mirroring，镜像卷，由两块硬盘组成，数据同时存入两者，写入速度慢，读取分配速度快
  - RAID 10/01 — 将 RAID 0 与 RAID 1 组合起来，有速度有冗余
  - RAID 5 — 至少由三块硬盘组成，数据被划分为多个大小相等的区块，循环写入硬盘，每块硬盘中加入奇偶校验值(Parity)，即该次循环写入其它硬盘数据的备份
    - 真正容量 = 总硬盘容量 - 1 块硬盘容量(奇偶校验占用)　硬盘越多，损失容量越小　读写性能强于 RAID 1，写入弱于 RAID 0
  - 建立 RAID — mdadm [模式] [RAID 设备文件名] [选项] [成员设备名]　more in p237

章9 ：设备管理

- 设备文件
  - 设备运行由其驱动控制　驱动与 /dev 下特殊文件联系　设备与该文件名相对应
  - 命名规则 ：主设备号 + 次设备号　主设备号 ：设备类型，sd fd 等　次设备号 ：同类设备中的顺序，如 sda sdb sda1(第一块硬盘的第一个主分区)等
  - 主设备号相同的设备使用同一驱动
- 设备分类(按有无物理载体)
  - 物理设备 ：实际存在的物理硬件的抽象
  - 虚拟设备 ：不依赖特定物理硬件，由内核自身提供某种功能
- 虚拟设备文件
  - /dev/null — 空设备，可写入，等价只写文件，数据有进无回
  - /dev/zero — 输入设备，产生连续不断的 null 流，用于创建指定长度的初始化空文件
  - /dev/full — 写入时返回无可用空间，读取时则为zero返回无限二进制零流，用于测试程序在遇到空间的错误行为
  - /dev/random[unrandom] — 随机数/伪随机数发生器
- 常用命令
  - lscpu — 查看 CPU 及处理器单元信息 ：从 sysfs & /proc/cpuinfo 获取
  - **free** — 查看内存信息 — free [选项]
    - -b/k/m/g ：以何种单位显示内存，-h 为自动选择
    - -t ：额外显示内存 & swap 总量
    - -s ：刷新时间间隔
    - -o ：不显示缓冲区
  - vmstat — 查看虚拟内存
    - 虚拟内存 ：运行内存超出内存使用时的部分/全部物理内存转移至硬盘中的进程使用，并将部分资料存储于硬盘
    - Linux 主要使用 调页 & 交换 完成以上内存调度
      - 调页 ：将部分不常用页面换到磁盘，活动页面保留于内存供进程使用
      - 交换 ：将整个进程写入磁盘
      - page-out ：页面从内存被写入磁盘(频繁发生将导致性能急剧下降)
      - page-in ：页面从磁盘中被读回内存
      - page-fault ：内核在物理内存中寻找不到页面
    - vmstat [-a] [延时时间 刷新次数]　vmstat [选项]　more in p247
  - 块设备查看
    - 块设备，存储以"块"为单位数据的设备，包括硬盘、闪存盘、U盘、CD-rom 等
    - **dmesg** — 调用系统启动/运行中内核产生的信息在缓冲区中的信息 — 查看 gpu ：dmesg |grep gpu　查看第一块网卡 ：gmesg |grep eth0
    - lsblk — 列出系统中所有可用的块设备及其依赖关系(除 RAM 盘)
  - 磁盘配额
    - 用于限制普通用户使用时的磁盘空间/文件数
    - quota
      - quotacheck ：扫描文件系统 & 建立磁盘配额记录文件
      - quotaon ：启动磁盘配额服务
      - quotaoff ：关闭磁盘配额服务
      - edquota ：用于修改用户 & 组的配额限制参数
      - more in p255

章10 ：网络管理

- 网络接口
  - 网络接口卡(NCI) ：可由 lspci 命令检测所有被内核检测到的 PCI 设备
  - 网络接口 ：LINUX 不将 NCI 设备作为文件访问，内核将每个网卡识别出的 NCI 都定义成以所示命名的网络接口
    - eth0 ：以太网
    - lo ：回环设备
    - ppp0 ：使用 PPP 协议的串口设备
    - tr0 ：令牌环
    - fddi0 ：光纤
    - 字母指连接技术，数字用于区分多个接口
  - ifconfig
    - **ifconfig 暂时改变网络接口状态而不写入配置文件，重启后从配置文件中读取参数**
    - 检测接口 ：-a　不加该选项只列举活跃接口
    - 配置网络接口 — ifconfig 接口名 [选项] [地址]
      - 选项 ：
      - up/down ：激活/使失效
      - netmask addr ：用 addr 指定的子网掩码
      - hw addr ：将设备 MAC 地址设置为 addr
    - 配置文件 — /etc/sysconfig/network-scripts/ifcfg-<接口名>
      - DEVICE ：必须的设备名
      - BOOTPROTO ：接口如何接受其配置信息
      - IPADDR ：与接口关联的 IP
      - NETMASK ：与指定 IP 关联的子网掩码
      - GATEWAY ：数据包路由出本地网络的网关
      - more in p259
- IP 路由 & 网关
  - IP 网络
    - 所有连接到一 ip 网络的设备都有相同的 IP 地址，并使用相同的网络掩码　每台主机共享相同的网络地址(子网掩码也相同)　一个 IP 地址标识一块网卡接口
    - 包括 ：网络部分 & 主机部分
    - 3 类子网掩码 netmask
      - IP 地址与子网掩码成对出现，子网掩码用于确定IP网络部分的位数
      - A类 255.0.0.0　B类 255.255.0.0　C类 255.255.255.0
      - 得到的网络地址与IP地址的关系 ：如 A 类子网掩码 ：255.0.0.0　IP 地址 ：118.43.96.2　网络地址 ：118.0.0.0　子网掩码后面的 0 会覆盖 IP 地址相应位置的字段
  - 以太网硬件
    - IP 地址 ：由管理员分配，用于主机间通信(无论是否于同一网络下)
    - MAC 地址 ：用于同一 IP 网络中不同主机间通信
    - 习惯上给回环接口分配的 IP 及子网掩码 ：127.0.0.1/255.255.255.0
  - 通信过程
    - 同一网络　源 A ：192.168.10.1　目标 B ：192.168.10.8
      1. 验证是否与 B 处于同一网络
      2. A 向本地网络发送广播数据包，请求 B 响应 A(ARP 请求)
      3. 所有本地主机都收到请求，仅 B 会回应并将MAC地址发送
      4. A 得到 B 的 ip 与 MAC，相互通信并储存该信息(arp -a 可查看)
      - ARP 协议 ：地址解析协议，可确定目标机器的硬件地址
    - 不同网络　源 A ：192.168.10.1　目标 B ：66.182.232.50
      1. 验证是否与 B 处于同一网络
      2. A 向默认网关请求帮助
         - 2.1 A 知道默认网关为 192.168.10.1
         - 2.2 A 采取默认网关，执行 ARP 请求获取网关 MAC 地址
      3. 发送数据包于网关，目标 IP 为 B，目的 MAC 为网关 MAC
      4. 网关获取该数据包并检验，转发目标数据包并转发
      5. 网关使用 ARP 获取目标 MAC，将数据包转发至下一路由
      6. 经过一系列转发，数据包最终抵达目标 B
  - 路由表
    - 用于确定内核使用何种方式传递数据　如本地网络直接发出，及本地网络的网关的身份标识
    - route -n 命令查看
    - 标志(Flags)项含 "G" 则指网关
    - 网关列为*/0.0.0.0则可用于本地主机通信

```
zhou_35@nano-X1BOA1 :~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.1.1     0.0.0.0         UG    0      0        0 eth0
192.168.1.0     *               255.255.255.0   U     1      0        0 eth0
```

```
(base) zhou@nano :~$ route -n
内核 IP 路由表
目标            网关            子网掩码        标志  跃点   引用  使用 接口
0.0.0.0         10.100.255.254  0.0.0.0         UG    100    0        0 enp5s0
10.100.0.0      0.0.0.0         255.255.0.0     U     100    0        0 enp5s0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 enp5s0
```

- DNS(域名系统)客户机
  - DNS 域名系统 ：将域名和 ip 相互映射的分布式数据库　域名解析 ：域名与 ip 的转化工作　DNS 服务器 ：域名解析服务器
  - **/etc/hosts** — ip 与主机名对应起来，通过主机名称代为相应的 ip　少数情况下使用 ：如引导和使用网络时的机器
    - 包含条目 ：
      1. 本地主机(localhost) ：localhost.localdomain & localhost
      2. 静态分配的 ip 地址
      3. 方便地址
    - 如 ：添加行 192.168.0.111 info　则浏览器可通过 http ：//info 或 ftp ：//info 指向相应ip
  - /etc/resolv.conf — 解析主机名过程 ：使用函数 resolve 首先静态查找 /etc/hosts　未果咨询 /etc/resolv.conf 中的域名服务器
  - 代理服务器 — 一般情况下，主机可能需要通过代理服务器才能联网，主机将主机名传给代理服务器解析IP
- DHCP(动态主机配置协议)配置
  - 能够使计算机启动时自动获得(ISP)请求 IP 地址和网络参数
  - 通过 DHCP 获取的 IP 地址为非固定地址　增加限口 ifup，动态增加一个 ip　减少限口 ifdown，动态减少一个 ip　DHCP 客户机与服务器间通过广播数据包完成通信　more in p265
- Web 服务器 — 最常见应用的多媒体信息服务系统　采用C/S(客户/服务器)结构
  - 组成 ：web 服务器、浏览器、通信协议
  - 如 ：HTTP ：超文本传输协议，用于客户端与服务器间通信，默认请求端口 ：80
  - Apache — 开源 web 服务器，跨平台...一体化优点，不适合多处理器环境　实现 www 服务功能，在服务器端可对浏览提供 Web 服务　软件包名 ：httpd　more in p268
- Telnet — 常用远程控制 web 服务方法 — Redis　默认端口 ：6379　more in p269
- 其它配置
  - 主机名 — 显示主机名 ：hostname　修改 ：hostname [new name]
  - /etc/sysconfig/network ：通用联网配置信息
    - NETWORKING ：yes/no ：若为'no'则启动过程的网络配置将被忽略
    - HOSTNAME ：按设定值设定主机名
    - GATWAY ：默认网关
    - GATWAYDEV ：网由 DHCP 为相关接口所配置的 IP 作为默认网关
  - 网配置操作服务
    - ifup/ifdown ：添加/删除接口，如 ifup eth0　ifconfig 接口名 up/down ：激活/使失效
    - 禁用/启动/重启联网 ：service network stop/start/restart
- 网络诊断
  - ping ：测试两台机器是否 IP 的连通性
  - host ：执行 DNS 查询
  - traceroute ：为数据包在路由器间穿行做详细报告
  - tcpdump ：监视网络活动性
  - 诊断问题 ：
    - 使用 ifconfig 查看接口激活当前位、及 IP 地址 & 子网掩码
    - ping 本地 IP
    - ping 默认网关
    - ping 外部 ip
    - host 已知主机名

章11 ：系统服务 & 日志

- Linux 系统服务
  - 服务 service — 常驻内存的进程，可提供一些系统/网络功能
  - 守护程序 Deamon
    - 实现服务的系统的系统服务　系统启动时自动加载，在系统退出时自动停止的系统任务
    - 依靠守护程序的启动 & 管理方式分类
      - 可独立启动服务 ：不通过其它机制管理，持续服务
      - 通过总管程序统一管理的服务 ：有请求时通过统一守护程序唤醒相应的服务，结束后则关闭
  - 独立服务启动方式 (只负责启动服务，与服务的自动启无关)
    - 使用 /etc/init.d/ 目录下的启动脚本启动服务　命令 ：/etc/init.d/ 服务名 start/stop/status/restart
    - 使用 service 命令启动独立服务
- 日志文件 — 重要的系统信息文件
  - /var/ ：保存系统动态数据的目录　/var/log/ ：系统日志目录
  - 绝大多数日志文件由 rsylogd 服务统一管理，各进程服务所产生的日志，则日志按规则以格式记录到不同文件中
  - rsylog 服务的配置文件 ：/etc/rsylog.conf　确定各服务的不同等级的日志在由哪的记录的位置
    - 基本格式 ：服务名.[连接符号][日志等级] 日志记录位置
    - 连接符号分类
      - . ：等级等于 & 高于连接符后的日志均被记录，如"cron.info"，cron 服务产生等于&高于 info 的日志等级均被记录
      - .= ：仅记录等于该等级的日志
      - .! ：记录非该等级的日志
    - 日志等级
      - Info ：基本通知
      - 其它高等级 ：notice、warning、err、crit、alert、emerg、*
      - 特殊等级 ：none ：忽略日志，不记录
      - debug
    - 记录位置
      - 绝对路径
      - 系统设备文件，如打印机
      - 转发给远程主机
      - 用户名
      - 忽略/丢弃 ：接收对象为"~"
  - 格式 ：
    - 事件产生时间
    - 产生事件的服务的主机名
    - 产生事件的服务名/程序名
    - 事件的具体信息
  - 日志轮转/转储 logrotate　/etc/logrotate.conf — 将旧日志移动 & 改名，创建新日志文件，旧文件超出一定范围则删除
  - 日志分析工具 ：logwatch