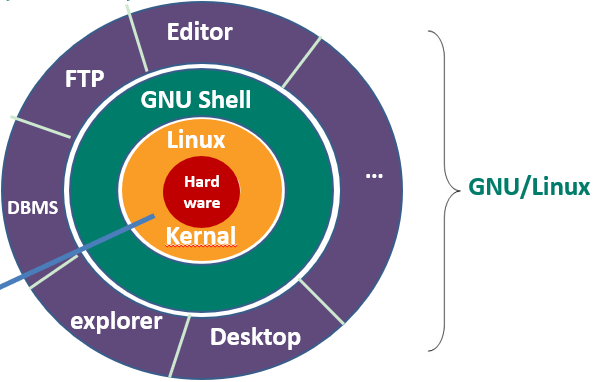
# Linux

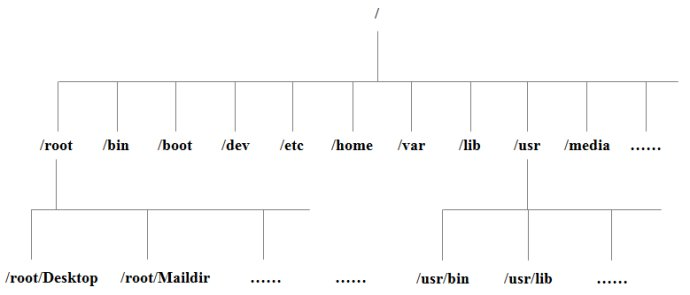
## Linux的简介

开源,稳定,漏洞少,一般的靠谱一点的公司都用它.

Linux是一套免费的使用和自由传播的类**Unix**操作系统



1. Linux的文件结构



## 项目结构

* /bin **★**(/usr/bin 、 /usr/local/bin)
  + 是Binary的缩写, 这个目录存放着最经常使用的命令
* /sbin (/usr/sbin 、 /usr/local/sbin)
  + s就是Super User的意思，这里存放的是系统管理员使用的系统管理程序。
* /home**★**
  + 存放普通用户的主目录，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。
* /root**★**
  + 该目录为系统管理员，也称作超级者的用户主目录。
* /lib
  + 系统开机所需要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。
* /etc**★**
  + 所有的系统管理所需要的配置文件和子目录
* /usr **★**
  + 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，类似与windows下的program files目录。
* /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，自己的安装别放这里
* /proc
  + 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。
* /srv
  + service缩写，该目录存放一些服务启动之后需要提取的数据。
* /sys
  + 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 。
* /tmp
  + 这个目录是用来存放一些临时文件的。
* /dev**★**
  + 类似于windows的设备管理器，把所有的硬件用文件的形式存储。
* /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，我们可以将外部的存储挂载在/mnt/上，然后进入该目录就可以查看里的内容了。
* /opt **★**
  + 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。
* /usr/local **★**
  + 这是另一个给主机额外安装软件所摆放的目录。一般是通过编译源码方式安装的程序。
* /var **★**
  + 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。
* /selinux
  + SELinux是一种安全子系统,它能控制程序只能访问特定文件。

## 1.3三种模式:

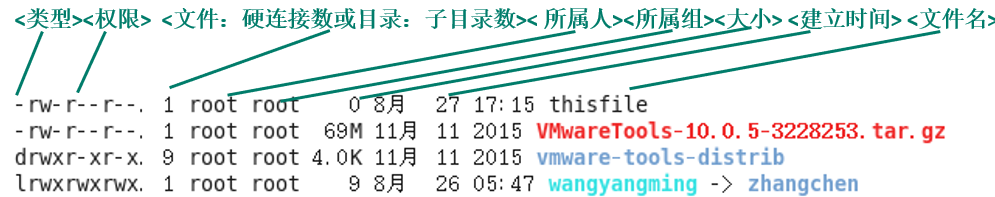
1. 一般模式
2. 编辑模式
3. 命令模式

### 1.1.1三种模式之间的切换

|  |
| --- |
| 在命令下  #vi xxx  ESC  ESC  :或/  I或者a  编辑模式  命令模式  一般模式 |

1. 帮助手册
   1. Man
   2. –help
2. 日期类
   1. Date , cal

**date +%Y-%m-%d' '%H:%M:%S**

1. 现实当前目录 : pwd
2. cd切换目录
   1. cd.. 返回上一层目录
   2. cd / 返回至/
   3. cd~ 返回家目录
   4. cd /etc/sysconfig 通过结对路径访问(利用tab键)
   5. cd ./sysconfig 通过相对路径访问
3. ls
   1. -a全部文件,连同隐藏文件
   2. -l 列出详细的列表 别名ll
   3. 列表信息
   4. 
   5. D开头就是目录
   6. -是文件
4. Grep
   1. 配合显示内容得名,根据跟随的内容现实,该行内容
   2. 例如:ls-l | grep xxx

10. mkdir (建立一个新的子目录（DOS命令）)

-p 可以一下建好多级目录

例： mkdir 目录名

1. Touch(接触)
   1. 新建一个文件
   2. touch aa.conf
2. **rmdir(**删除目录**)**
   1. 删除一个空目录
   2. 例： rmdir dirname
3. **Rm**
   1. 移除文件或目录
   2. -rf 递归删除所有目录内容，不提示
4. **Cp(复制)**
   1. cp 【要复制的文件】 【到哪里】
   2. -r 递归复制整个文件夹
   3. -v 显示复制过程中文件的列表
   4. 强制覆盖不提示的方法
      1. 临时方法：用\cp
5. **mv**
   1. mv oldFileName newFileName 重命名
   2. mv /temp/movefile /targetFolder 移动文件
6. **cat**
   1. cat 文件名 查看轻量级的文本文件
   2. cat 文件1 文件2 连接显示多个文件
   3. cat 文件1 文件2 > 文件3 合并为新文件
7. **more(**更多**)**
   1. **查看较长的文件**
   2. 空白键 (space)：代表向下翻一页
   3. Enter:代表向下翻一行
   4. q:代表立刻离开 more ，不再显示该文件内容。
   5. Ctrl+F 向下滚动一屏
   6. b 返回上一屏
8. **Less(**较少**)**
   1. **同more类似，比more功能更多。**
   2. [pagedown]：向下翻动一页；
   3. [pageup] ：向上翻动一页；
   4. /字串 ：向下搜寻『字串』的功能；
   5. ?字串 ：向上搜寻『字串』的功能；
   6. n ：重复前一个搜寻 (与 / 或 ? 有关！)
   7. N ：反向的重复前一个搜寻 (与 / 或 ? 有关！)
9. Tail(尾部) head(头部 )
   1. 从尾部开始查看，比较适合查看日志
   2. -f 跟随查看，实时查看数据
   3. -n200 显示的行数，n可以省略。
10. History(历史)
    1. 查看历史命令
11. echo
    1. 输出环境变量
    2. 在shell脚本中当system.out.print用
12. Find(发现)
    1. 查找文件，提供了丰富的模糊搜索及条件搜索
    2. find+搜索路径+参数+搜索关键字
    3. 按文件名：**find /home/esop -name '\*.txt'** 根据名称查找/目录下的filename.txt文件。
13. Locate(定位)
    1. 查找文件，查询速度更快，使用更方便。
    2. 运行前需要 执行updatedb来更新文件索引。
14. 软链接
    1. 也成为符号链接，类似于windows里的快捷方式，有自己的数据块，主要存放了链接其他文件的路径。
    2. **命令 ln –s 原文件或目录名 软链接名**
    3. **查询：通过 ls -l 就可以查看,列表属性第1位是l,尾部会有位置指向。**

**25. yum install para yum安装软件**

## 1.4 vi和vim的区别:

Vi是白色

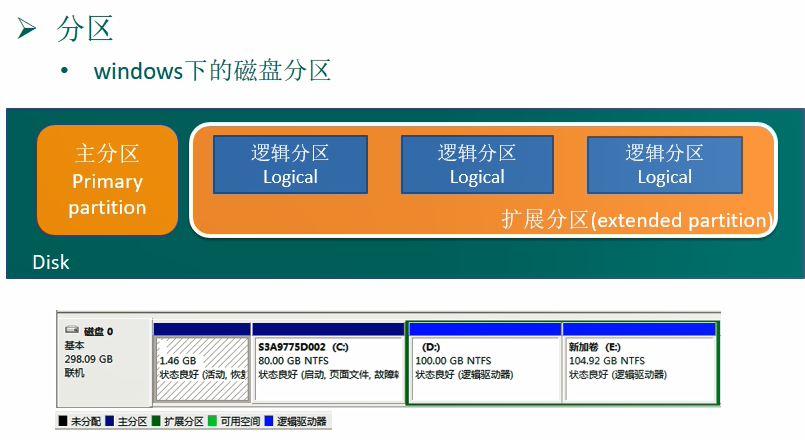
Vim是彩色

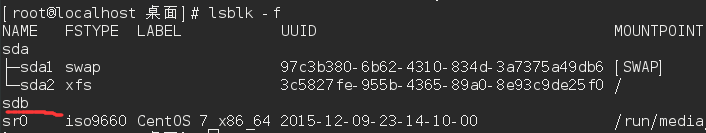
软连接

也成为符号

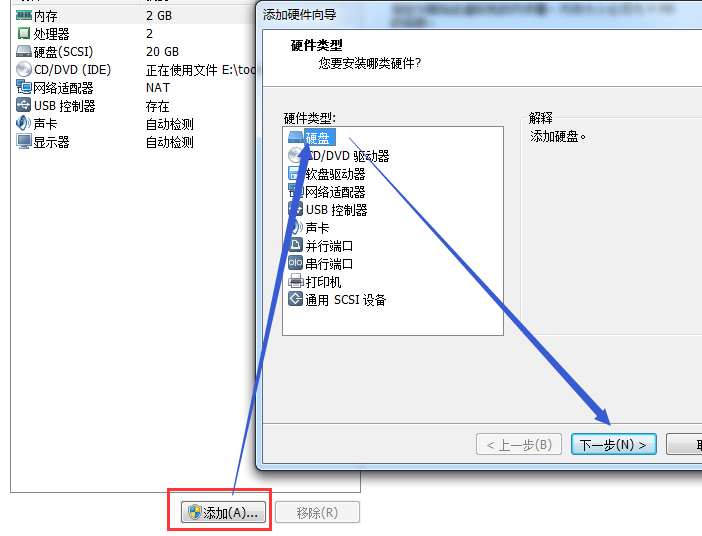
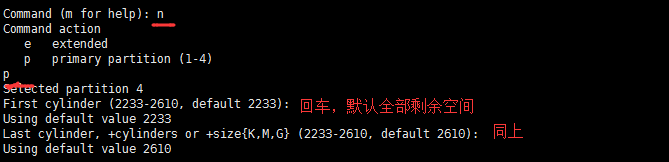
## 分区 (磁盘分区类)

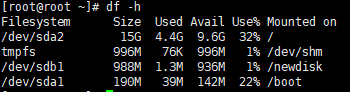
1. Mbr
2. 只支持4个主分区
3. 系统只安装主分区
4. 扩展分区占一个主分区
5. Gpt
6. Win7 64位以后支持
7. 无线主分区
8. 支持4t以上的硬盘



* 查看所有设备挂载情况
  + 命令 ：**lsblk 或者 lsblk -f**
  + 

### 向虚拟机增加一块硬盘

1. **虚拟机插硬盘** 
   1. **虚拟机增加硬盘**
   2. **在【虚拟机】菜单中，选择【设置】，然后设备列表里添加硬盘，然后一路【下一步】，中间只有选择磁盘大小的地方需要修改，至到完成。然后重启系统！**
   3. ****
2. **分区**
   1. **分区命令 fdisk /dev/sdb**
      1. **开始对/sda分区**
         1. **m :显示命令列表**
         2. **p :显示磁盘分区**
         3. **n :新增分区**
         4. **d :删除分区**
         5. **w :写入并退出**
   2. **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w:(写入分区并退出),如果不保存就按p退出**
   3. ****
   4. ****
3. **格式化**
   1. **格式化磁盘**
      1. **分区命令: mkfs -t ext4 /dev/sdb1**
      2. **其中ext4是分区的类型**
4. **挂载**
   1. **挂载**
      1. **将一个分区与一个目录联系起来**
      2. **Mount 设备名称 挂载目录**
         1. **例如: mount /dev/sdb1 /newdisk**
         2. **Umount 设备名称 或者 挂载目录**
            1. **例如: umount /dev/sdb1 或者 umount /newdisk**
         3. **命令行挂载重启后会失效**
   2. **永久挂载**
      1. **通过修改/ect/fstab实现挂载**
      2. **添加完成后 执行mount -a 立即成效**
5. **磁盘情况查看**
   1. **hf -h**
      1. **查询系统整体使用情况**



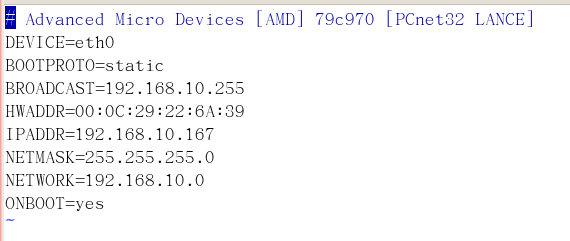
* 1. du -h /目录
     1. 查询指定目录的磁盘占用情况，默认为当前目录
     2. -s 指定目录占用大小汇总
     3. -h 带计量单位
     4. -a 含文件
     5. --max-depth=1 子目录深度
     6. -c 列出明细的同时，增加汇总值

例： du -ach --max-depth=1 /opt

## Ifconfig (网络配置)

### 查看网络配置ip addr

* 1. 查看网络配置
  2. 如何修改IP
  3. 图形化操作
  4. 命令行方式 vi /etc/sysconfig/network-scripts/ifcfg-en0s3



|  |
| --- |
| **DEVICE=eth0** #接口名（设备,网卡） |
| **BOOTPROTO=static** |
| # 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） |
| **BROADCAST=192.168.1.255** #广播地址 |
| **HWADDR=00:0C:2x:6x:0x:xx** #MAC地址 |
| **IPADDR=192.168.1.23** #IP地址 |
| **NETMASK=255.255.255.0** # 网络掩码 |
| **NETWORK=192.168.1.0** #网络地址 |
| **ONBOOT=yes** #系统启动的时候网络接口是否有效（yes/no） |
| **TYPE=Ethernet** #网络类型（通常是Ethemet） |

* 1. vim /etc/sysconfig/network-scripts/ifcfg-eth0
  2. 刷新网络设置
  3. **service network restart**

## ps (进程类)

* **ps**
  + ps –aux|grep xxx
    - **System V展示风格**
    - USER：用户名称
    - PID：进程号
    - **%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（单位：KB）**
    - **RSS：进程占用的物理内存大小（单位：KB）**
    - TT：终端名称,缩写 .
    - STAT：进程状态，其中S-睡眠，s-表示该进程是会话的先导进程，N-表示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
    - STARTED：进程的启动时间
    - TIME：CPU时间，即进程使用CPU的总时间
    - COMMAND：启动进程所用的命令和参数**，如果过长会被截断显示**
* **ps**
  + **ps -ef|grep xxx**
    - **是BSD风格**
    - **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，表明进程是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：进程启动的时间**
    - **TTY：完整的终端名称**
    - **TIME：CPU时间**
    - **CMD：启动进程所用的命令和参数**

**综上：**

* **如果想查看进程的CPU占用率和内存占用率，可以使用aux**
* **如果想查看进程的父进程ID可以使用ef**
* pid
* 每个进程都有一个pid作为唯一标识。
* Kill
* Kill pid 通过唯一标识pid杀死进程的工程
* Killall
* Killall name 通过进程名杀死进程
* 运行级别(runlevel)

开机 🡪 BIOS 🡪 /boot 🡪 init进程 🡪 运行级别 🡪 运行级对应服务

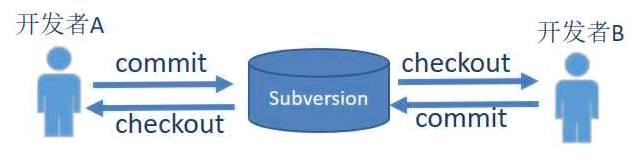
* **查看运行级别 : vi /ect /inittab**

## Git与GitHub

### 版本管理系统能干什么:

* + - 1. 冲突解决
      2. 协同开发
      3. 版本记录
      4. 历史追查
      5. 代码备份
      6. 版本还原
      7. 权限管理
      8. 代码审查
      9. 分支管理

### 管理



经典的集中管理( CVS . VSS . SVN )

特点:

实现了大部分开发中对版本管理的需求

结构简单,上手容易

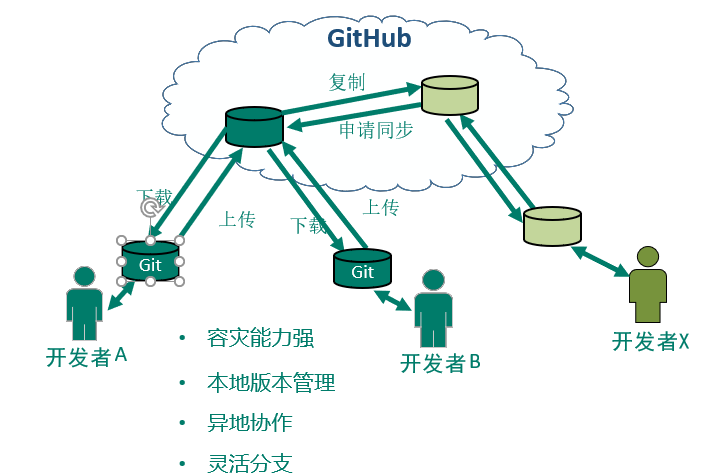
缺点:

1、版本管理的服务器一旦崩溃，硬盘损坏，代码如何恢复？

2、程序员上传到服务器的代码要求是完整版本，但是程序员开发过程中想做小版本的管理，以便追溯查询，怎么破？

3、系统正在上线运行，时不时还要修改bug，要增加好几个功能要几个月，如何管理几个版本？

4、如何管理一个分布在世界各地、互不相识的大型开发团队？



### git下载

1、命令行工具：Git for windows

下载地址：<https://git-for-windows.github.io/>

2、 操作系统中可视化工具：TortoiseGit

下载地址： <https://tortoisegit.org/>

3、 Eclipse插件： Egit

Eclipse自带，插件市场搜索最新版

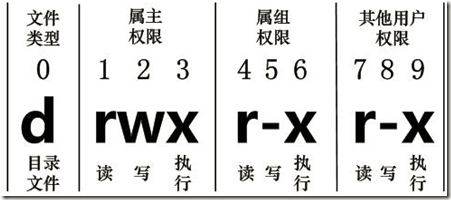
4、 GitHub网站

[http://www.github.com](http://www.github.com/)

## Linux用户与权限管理

* 用户

Linux系统是一个多用户多任务的操作系统，任何一个要使用系统资源的用户，都必须首先向系统管理员申请一个账号，然后以这个账号的身份进入系统。

* 新增用户:useradd 新用户名
* 设置密码: passwd 用户名
* 用户是否存在：id 用户名
* 切换用户：su – 切换用户名
* 查看当前用户/登录用户：whoami/ who am I
* 删除用户：userdel
* 用户组
* 类似于角色，系统可以对有共性的多个用户进行统一的管理。
* 新增组:groupadd 组名
* 删除组:groupdel 组名
* 修改用户的组：usermod –g 用户组 用户名
* 增加用户时直接加上组：useradd –g 用户组 用户名
  + 系统中用户和组的相关文件
* 用户（user）的配置文件： /etc/passwd
* 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**
* 口令的配置文件: /etc/shadow
* 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**
* 组(group)的配置文件： /etc/group
* 每行含义：**组名:口令:组标识号:组内用户列表**
  + 文件的全限管理
* 再说ls -l
* 
* 0-9位说明
* 第0位确定文件类型(d, - , l , c , b)
* 第1-3位确定所有者（该文件的所有者）拥有该文件的权限。---User
* 第4-6位确定所属组（同用户组的）拥有该文件的权限，---Group
* 第7-9位确定其他用户拥有该文件的权限 ---Other
  + 作用到文件
* 作用到文件
* [ r ]代表可读(read): 可以读取,查看
* [ w ]代表可写(write): 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(execute):可以被系统执行
  + 作用到目录
* 作用到目录
* [ r ]代表可读(read): 可以读取，ls查看目录内容
* [ w ]代表可写(write): 可以修改,目录内创建+删除+重命名目录
* [ x ]代表可执行(execute):可以进入该目录
  + **chmod**
* 第一种方式：+ 、-、= 变更权限
* u:所有者 g:所有组 o:其他人 a:所有人(u、g、o的总和)
* 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
* 第二种方式：通过数字变更权限
* r=4 w=2 x=1 rwx=4+2+1=7
* chmod u=rwx,g=rx,o=x 文件目录名
* 相当于 chmod 751 文件目录名
* 
* **chown**
  + chown newowner file 改变文件的所有者
  + chown newowner:newgroup file 改变用户的所有者和所有组
  + -R 如果是目录 则使其下所有子文件或目录递归生效
* **chgrp**
  + chgrp newgroup file 改变文件的所有组

一般模式