

第 1 讲 概述

王晓庆

wangxiaoqing@outlook.com

June 20, 2016

Outline

- 1 课程简介
- 2 从 Unix 到 Linux
- 3 Linux 安装
- 4 Linux 入门

1.1 为什么要学 Linux?

Linux 与 IT 行业

- 免费、开源、全球参与开发
- 被业界广泛采用，成为全球 IT 基础设施核心
- 在移动和物联网时代发展愈加迅猛
- 是各种新技术产生的温床

Linux 与我们

- 我们平时都只用 Windows，从来没用过 Linux 呀？
 - 看新闻、购物、抢票、QQ 微信、邮件、听歌、看视频……
- 与 Unix 一脉相承，继承了 Unix 丰富的知识和软件宝库
- 通过学习命令行和脚本编程，可以实现高效且自动化地处理各种任务
- 通过学习系统配置管理，可以深入掌控系统的方方面面
- 学到的知识可以保值并随着积累不断增值
- 可以第一时间接触和了解业界最新技术

shell 命令行

- GUI(图形用户界面)
 - 拟物化（人操作物）：人亲自执行
 - 适合视觉型活动：图形图像设计、音视频制作、上网、看视频、玩游戏
 - 例：制作平面广告
- CLI(命令行界面)
 - 拟人化（人机对话）：让机器执行
 - 适合语言型活动：编程开发、系统管理、文档处理
 - 例：在 data 文件夹内创建 12 个文件夹分别存放 12 个月的数据、每个文件夹内再创建 30 个子文件夹存放每天的数据

```
mkdir -p data/month{01..12}/day{01..30}
```

1.2 主要学什么？

shell 脚本编程

- shell 变量
- shell 参数
- shell 控制语句
- shell 数组
- shell 函数
- shell 计算
- shell 中断处理
- sed
- awk

系统配置管理

1.2 主要学什么？

网络服务配置管理

- 网络基础配置
- 网络访问控制
- DNS 服务
- DHCP 服务
- NFS 服务
- Samba 服务
- Web 服务
- 远程登录配置
- 防火墙配置
- 代理服务

考核方式

- 考勤：20%
 - 随机考勤，每次旷课扣 3 分（早退按旷课处理），每次迟到扣 1 分，扣完为止。
- 实践：30%
 - 平时：14%
 - 根据每周上机实验情况打分
 - 说明：未在规定时间内完成和提交实验，分数乘 0.7 进行折扣
 - 期末：16%
 - 未在规定时间内完成，分数乘 0.8 进行折扣
- 期末笔试：50%
 - 闭卷考试，卷面分数不得低于 50 分，否则总评分会被系统自动定为不及格！

创世纪 (1969-1971)

- 出师不利的 Multics
- Ken 的 space travel 游戏
- 废弃的 DPD-7

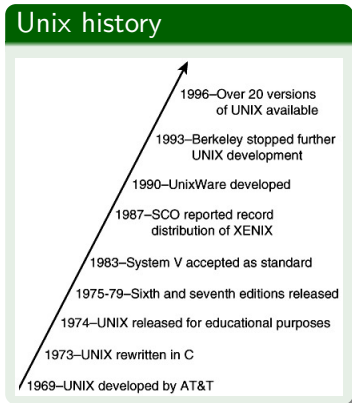
Thompson and Ritchie



2.1 Unix 简史

出埃及记 (1971-1983)

- 1971 年：C 语言诞生
- 1973 年：Unix 完全用 C 重写
- 1975 年：Unix V6 版发布
- 1977 年：Bill Joy 发布 BSD v1 版
- 1983 年：BSD 4.2 上首次实现 TCP/IP



- Unix 商业化，源代码封闭
- 荷兰教授 Andrew S. Tanenbaum 开发 Minix
- 类 Unix、小巧、免费、教学

Tanenbaum



Internet

- 80 年代，Internet 逐渐形成
- 早期以技术用户为主
- 通过网络切磋技术、协同工作、发布和获取软件代码
- 形成植根于 Internet 的“黑客”文化

GNU

- 1983 年，MIT 的 Richard Stallman 开创 GNU 计划
- GNU 致力于开发一个自由的类 Unix 系统
- 1988 年，发布 GPL 许可协议用以保护自由软件
- 1991 年，GNU 完成除内核外几乎所有必备软件的开发

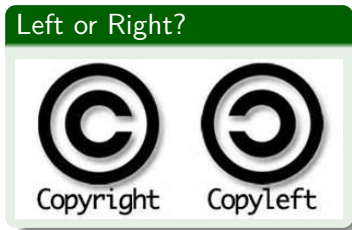


自由软件

- 自由软件赋予软件使用者四种自由：
 - (freedom 0) 为任何目的自由运行该软件的自由；
 - (freedom 1) 有研究该软件如何运行，以及按需修改该软件的自由；
 - (freedom 2) 有重新发布该软件拷贝的自由；
 - (freedom 3) 有改进该软件，以及向公众发布改进的自由，这样整个社群都可受惠。
- 源代码开放是获得自由 1 和自由 3 的前提条件！

GPL

- 当开发者以 GNU GPL 作为软件许可证发布其软件时，该软件就成为自由软件并能保持自由软件的性质。
- GPL 规定的版权为 Copyleft，它允许任何人修改并重新分发自由软件，但要求保证该软件是仍然是自由软件。
- Copyright(商业版权) 是为了限制用户
- Copyleft(自由版权) 是为了维护用户自由的权力



Linux 的诞生

- 1991 年 8 月 25 日，21 岁的芬兰赫尔辛基大学计算机科学系二年级学生 Linus Torvalds 在 comp.os.minix 新闻组中宣告了 Linux 的诞生。



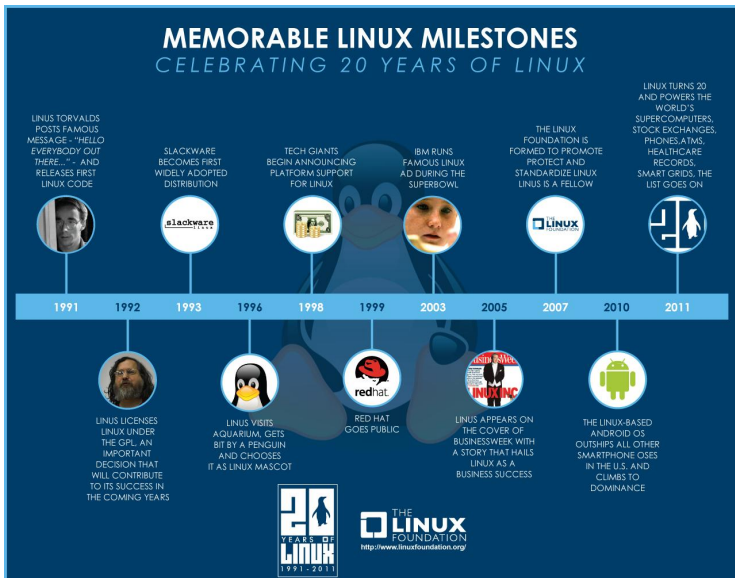
Linux 的快速发展

- Linux 为什么会成功？
 - 站在 GNU 的肩膀上
 - 采用 GPL 许可协议发布
 - 通过 Internet 协作开发
- 今天，Linux 已经成长为业界最重要的操作系统，得到几乎所有业界大公司的支持，也是目前运行硬件平台最多的操作系统。



2.3 Linux 简介

Linux 的快速发展



Linux 的特点

- 基于 UNIX 设计，性能出色
- 遵照 GPL 许可，自由软件
- 符合 POSIX 标准，兼容性好
- 可移植性好
- 网络功能强大
- 安全性好

Linux 的内核版本

- Linux 版本号由 3 个数字组成：r.x.y
 - r：目前发布的 Kernel 主版本
 - x：偶数是稳定版本，奇数是开发版本
 - y：错误修补次数
- 查看系统的内核版本号

```
uname -r
```

- 内核官网
 - www.kernel.org

Linux 发行版

- 各 IT 厂商和组织把 Linux 内核与大量应用软件按各自的方式打包成便于安装的形式，就称为 Linux 发行版
- 发行版本号随发布者的不同而不同，与系统内核版本号相对独立
- 目前世界上已经有超过百种不同的 Linux 发行版
 - <http://distrowatch.com>

2.3 Linux 简介

Linux 发行版



Windows 和 Linux

- Windows 和 Linux 各有特点
 - Windows 是商业软件，Linux 是自由软件
 - Windows 图形界面友好，Linux 命令行灵活高效
 - Windows 易用性好，Linux 定制性强
 - Windows 像傻瓜相机，Linux 像单反相机
- 在学习 Linux 时注意和 Windows 进行对比

无处不在的 Linux

- Android 基于 Linux，每天有超过 550,000 部 Android 设备被激活。



无处不在的 Linux

- 当前全球 top 500 超级计算机中有 469 台运行 Linux。



无处不在的 Linux

- 高铁运行管理



● 交通控制系统



无处不在的 Linux

- 工业制造



无处不在的 Linux

● 智能农牧业



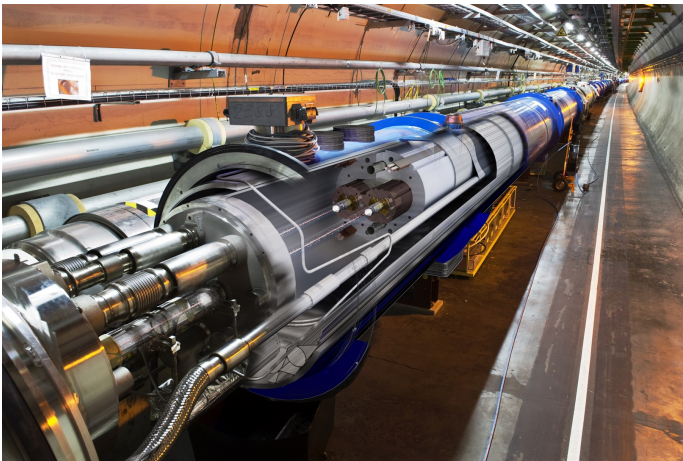
无处不在的 Linux

● 金融证券



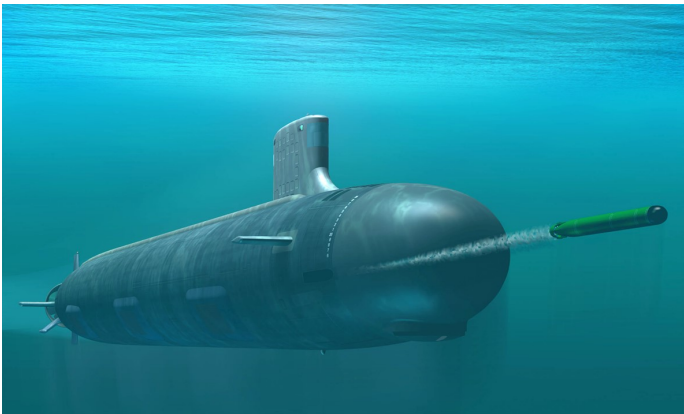
无处不在的 Linux

- 高能粒子加速器



无处不在的 Linux

- 核潜艇



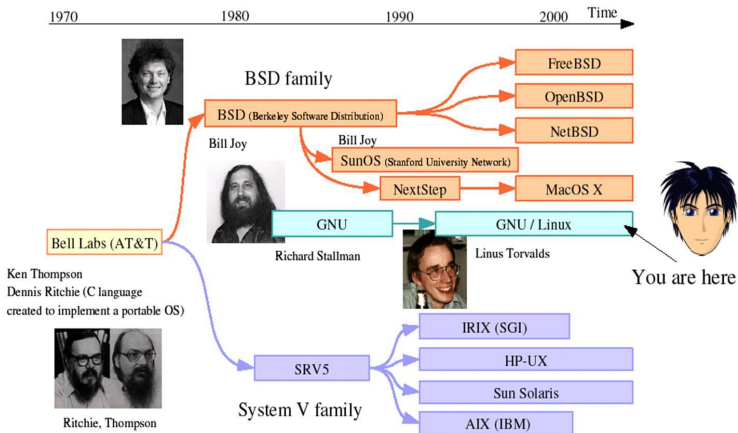
无处不在的 Linux

- 影视娱乐



2.4 Linux vs. Windows

You are here



虚拟机软件

- 常用虚拟机软件
 - vmware
 - virtualbox
 - kvm
- 利用虚拟机软件安装操作系统
 - ① 创建虚拟机 (指定虚拟机硬件参数)
 - ② 为虚拟机挂载操作系统安装光盘
 - ③ 在虚拟机上安装操作系统

分区与目录

- 硬盘分区规则
 - 一块普通硬盘最少要分 1 个主分区，最多能分 4 个主分区
 - 也可以分 1-3 个主分区，外加一个扩展分区
 - 扩展分区内可以划分更多的逻辑分区
- Windows 分区与 Linux 分区
 - Windows
 - 多根目录：每个分区都有一个根目录
 - Linux
 - 单根目录：整个系统只有一个根目录
 - 所有分区都挂载在目录树上的某个目录 (称为挂载点) 下面
 - 根分区挂载于 / 目录，其他分区挂载于根目录下的子目录
 - 对分区的文件访问，通过相应的挂载点进行

3.2 Linux 分区

Linux 目录结构

分区方案

- /分区 +Swap 交换分区
 - swap 分区：物理内存不够时进行内存交换，无挂载点
 - 一般设为物理内存的 1-2 倍大小
- /分区 +/boot 分区 +Swap 交换分区
 - /boot 用于存放系统引导程序、系统内核等
 - 一般不超过 500MB
- 其他常用来挂载分区的目录
 - /home 用于存放用户个人的数据
 - /usr 用于存放不太变化的内容，如应用程序和文档
 - /tmp 用于存放临时文件
 - /var 用于存放经常变化的内容，如日志、邮件等
 - /opt 用于存放系统附加软件包

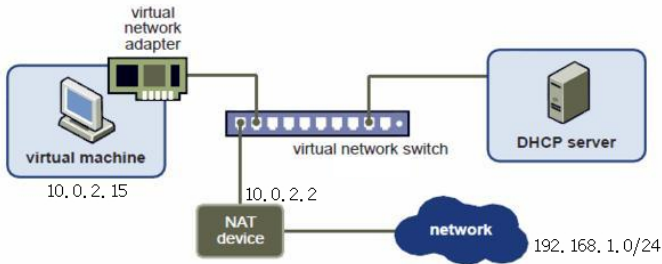
分区名称

- IDE 接口设备及其分区
 - 硬盘：`/dev/hda /dev/hdb ...`
 - 分区：`/dev/hda1 /dev/hda2 ...`
- SCSI 接口设备及其分区
 - 硬盘：`/dev/sda /dev/sdb ...`
 - 分区：`/dev/sdb1 /dev/sdb3 ...`
- 注意
 - 逻辑分区的编号从 5 开始
 - SATA 接口和 USB 接口也被认作 SCSI 接口
- 设备文件名 vs. 挂载点
 - 设备文件名：是一个文件，代表设备本身
 - 挂载点：是一个目录，可用于访问设备内容
 - 例：将 `/dev/cdrom` 挂载到 `/media/cdrom`

3.3 虚拟机网络设置

NAT

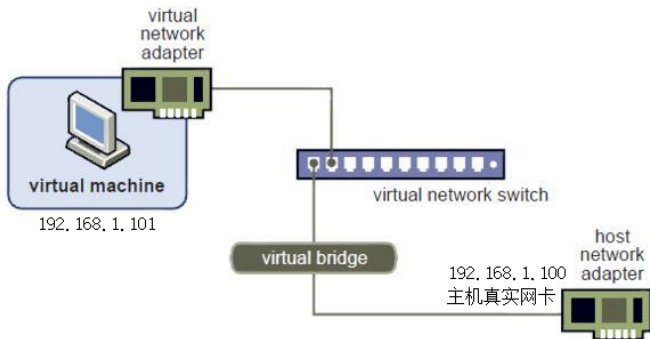
- 虚拟机通过虚拟 NAT 设备连接到主机，并可通过主机连接外部网络



3.3 虚拟机网络设置

Bridged

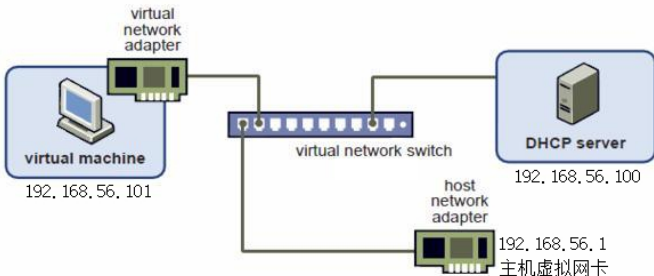
- 虚拟机通过虚拟交换机连接到主机所在局域网



3.3 虚拟机网络设置

Host-only

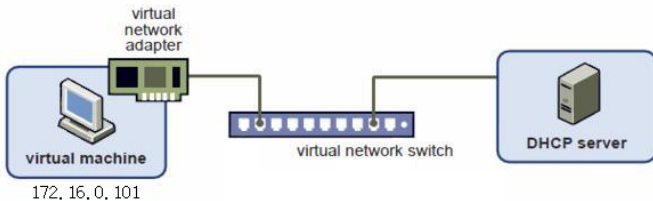
- 虚拟机通过虚拟交换机连接到主机虚拟网卡所连的局域网



3.3 虚拟机网络设置

Internal

- 虚拟机通过虚拟交换机连接到虚拟机内部的虚拟局域网



ssh 客户端

- Linux 客户端

```
ssh -l username 192.168.56.102
```

```
ssh username@192.168.56.102
```

```
ssh -p 22 -l username 192.168.56.102
```

- Windows 客户端

- putty
- SecureCRT

- ## ● 主机访问 NAT 后的虚拟机

- 虚拟机上设置端口转发: 假设设置 127.0.0.1:2222->10.0.2.15:22

```
ssh -p 2222 -l username 127.0.0.1
```

虚拟机常用操作

- 快照备份与恢复
 - 可备份某时刻的系统状态，将来虚拟机由于误操作出现问题时可以恢复到备份时的状态。
 - 可以对虚拟机状态进行多次备份，并随时切换到某个备份状态。
- 利用虚拟硬盘快速创建虚拟机
 - 利用已经安装好了虚拟机的虚拟硬盘快速创建新的虚拟机。
- 虚拟机复制
 - 利用现有虚拟机快速克隆出新的虚拟机。

终端与控制台

- 终端和控制台都不是个人电脑的概念，而是多人共用的小型中型大型计算机上的概念。
 - 终端
 - 字符终端
 - 图形终端
 - 控制台
 - 通过显示器和键盘接口与主机相连的键盘和显示器
 - 个人计算机只有控制台，没有终端
 - 本地虚拟终端：在控制台上用软件虚拟出来的终端
 - 图形终端：Ctrl-Alt-F1
 - 字符终端：Ctrl-Alt-F2~Ctrl-Alt-F6
 - 在字符终端上切换时，可以不按 Ctrl 键
 - 图形终端里面也有自己的虚拟字符终端
 - 远程虚拟终端
 - 通过网络连接到计算机的虚拟终端。

终端与多用户

- 多个终端可以供不同用户同时登录使用系统
- 单个用户可以同时使用多个终端
 - 用户可同时打开多个终端，每个终端上做不同的事情。
- 用户在终端登录后还可以切换为其他用户身份
 - 系统管理员 root 可以随时切换为其他用户身份
 - 普通用户 tom 在知道其他用户的登录名和登录密码时，也可以切换为其他用户身份

```
# su - mary # root切换为mary
$ su - root # tom切换为root, 会提示输入root密码
$ su -      # 同上
$ exit      # 返回原先用户身份, 快捷键为Ctrl-d
```


登录和注销

● 登录

- 系统提示 login :
- 输入用户名
- 输入用户密码
 - 注意：字符终端下输入密码时，系统会关闭屏幕回显。

● 注销

- 注销代表用户离开，不等于关机
- 注销命令

```
exit      # 快捷键为Ctrl-d
logout
```

打个招呼吧

• echo

```
echo hello
echo nice to meet you!
echo "nice to meet you!"
echo 'nice to meet you!'
echo nice to meet you\!
echo "nice to meet you\!"
echo 'nice to meet you\!'
```

试一试

请打印字符串 It's time to learn linux!

打个招呼吧

• echo

```
echo hello \  
nice to meet you!  
echo 'hello  
nice to meet you!'  
echo "hello  
nice to meet you."  
echo -e 'hello\nnice to meet you!'  
echo -e hello\nnice to meet you!
```

试一试

请打印所有天干 (甲乙丙丁戊己庚辛壬癸) 地支 (子丑寅卯辰巳午未申酉戌亥) 的组合

查查日历

- 日历

```
cal                # 给我看看这个月的月历吧
cal 2017           # 我想看看 2017 年的年历
cal 9 1752         # 我想看看 1752 年 9 月份的月历:-)
```

- 一次运行多条命令：命令组

```
cal;date
```

查看用户

● 查看用户

```
whoami      # 我是谁？
who am i    # 我究竟是谁？
who          # 都有谁在呀？
w           # 都有谁在呀？
```

了解系统

● 系统版本

```
uname      # 什么操作系统？
uname -r   # 什么内核版本？
uname -a   # 所有版本信息？
```

● 主机名

```
hostname      # 主机名是什么？
hostname NiuBi # 改个牛逼的名字吧！
```

● 系统状态

```
uptime      # 开机多久了呀？
uptime -p   # 能简单点吗？
```

计算器

- bc

```
bc          # 我想算点东西
```

```
bc -q      # 安静点!
```

 $3+4$

```
5*last # 计算 5*7
```

last/9

```
scale=16;x=4;y=7;3*(x+2)/y
```

```
obase=2;192
```

```
obase=10;ibase=2;11000011    # 11000011 等于几？
```

```
ibase=2;obase=10;11000011    # :- (
```

```
obase=1010;11000011      # :-)
```

```
quit # 退出
```


计算器

- bc

● 常用 bc 内置函数

- $s(x)$ 正弦函数, x 为弧度值
- $c(x)$ 余弦函数
- $a(x)$ 反正切函数
- $l(x)$ 对数函数 (以 2 为底)
- $e(x)$ e 的指数函数

```
bc -l          # 必须加-l 参数才能调用内置函数
scale=10;l(3)  # 计算  $\log(3)$ 
```

试一试

- ① 将 16 进制数 FFEFF 转换为 10 进制数
- ② 计算圆周率至小数点后 1000 位

查看文件列表

● ls 命令

```
ls          # 查看当前目录的文件列表
ls -a       # 查看当前目录所有文件列表
ls -l       # 查看当前目录的文件详细信息
ls -la      # 查看当前目录所有文件详细信息
ls -ld      # 查看当前目录本身的详细信息
ls -Rm /boot # 以递归紧凑方式查看/boot 目录
ls -lt      # 按时间排序（从新到旧）
ls -ltr     # 按时间反向排序（从旧到新）
ls -ls      # 按文件大小排序（从大到小）
```

到处逛逛

- cd 命令

```
cd /usr      # 进入 /usr 目录
cd           # 返回我的主目录
cd ~         # 返回我的主目录
cd ~xiaobai  # 进入 xiaobai 的主目录
cd -         # 回到上次去的目录
```

- **pwd 命令：我到哪儿了？**

pwd

查看文件内容

- cat (concatenate) 和 tac

```
cat /etc/hosts.allow
```

```
tac /etc/hosts.allow
```

```
cat -n /etc/hosts.allow
```

```
cat /etc/hosts.allow /etc/hosts.deny
```

```
tac /etc/hosts.{allow, deny}
```

[查看文件内容](#)

- more 和 less

```
more /etc/services
less /etc/services
```

- head 和 tail

```
head /etc/passwd  
head -n 3 /etc/passwd  
tail -3 /etc/passwd  
su  
tail -f /var/log/messages # 跟踪文件结尾追加的行  
                           # 按 Ctrl-C 结束  
service network restart
```

Linux 目录结构

- 遵循文件系统层次标准 (Filesystem Hierarchy Standard, FHS)
 - FHS 定义了两层规范
 - 第 1 层：规定了 / 下面的子目录应该存放什么文件
 - 第 2 层：规定了 /usr 和 /var 下面的子目录应该存放什么文件
 - 其他子目录内可以自行配置存放什么文件

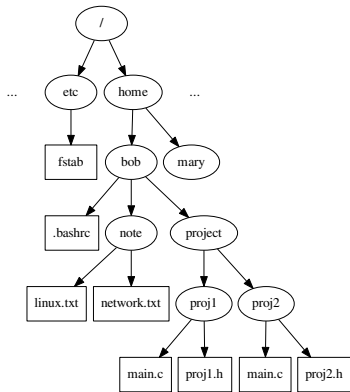
Linux 目录结构

/bin	存放用户常用命令
/boot	存放系统启动文件
/dev	存放设备文件
/etc	存放系统配置文件
/home	各用户主目录
/lib	存放动态连接共享库
/media	为可移动存储设备提供挂载点
/mnt	为某些设备提供挂载点
/root	root 用户主目录，不是根目录
/proc	系统自动产生的内存映射信息
/sbin	存放系统管理员使用的命令
/tmp	存放临时文件
/usr	存放应用程序和文件
/var	保存经常变化的内容，如日志、邮件、打印任务

绝对路径与相对路径

例

```
cd /home/bob
cat .bashrc
cat note/linux.txt
cd project/proj1
cat main.c
cat ../proj2/proj2.h
cat /etc/fstab
pwd
cd
```



聊聊天吧

xiaobai

write laohei

hello, laohei!

I'm learning linux, it's fun!

Thank you!

Ctrl-d

laohei

write xiaobai

hello, xiaobai.

Good luck.

You're welcome.

Ctrl-d

广而告之和请勿打扰

- wall

```
wall 'Pay attention! there comes BIG news!'
```

- mesg

```
mesg n      # 请勿打扰
mesg y      # 欢迎来访
```

- who

```
who -w      # 查看用户是否屏蔽消息
```

收发邮件

xiaobai 发邮件

```
mail laohei
help                # 输入邮件主题
Hello, laohei, I have some problems with my linux.
Can you help me?    # 输入邮件正文
Ctrl-d              # 结束邮件内容并发送邮件
```

laohei 收邮件和回复邮件

```
mail          # 查看邮箱
p             # 打印邮件
R            # 回复邮件
Hello, xiaobai, I have time at 10:00 am on Friday.
.            # 结束邮件内容
q            # 退出
```

关机/重启

- shutdown

```
shutdown -h now    # 立即关机
shutdown +10 "System will shutdown in 10 minutes."
shutdown -c "Shutdown has been canceled."
```

- halt

- poweroff

- reboot

命令行格式

- 命令格式：命令 [选项]... [参数]...
 - 选项（指示命令以什么方式执行）
 - Unix 简洁风：-a
 - 多个选项可以共用一个减号
 - GNU 友好风：-all
 - 参数（指示命令作用的对象，选项也可以有参数）
 - 注意
 - 命令、选项和参数之间要有空格（选项及其参数之间有时可以没有空格）
 - 选项及其参数之间不能放置其他选项

命令行示例

```
tar -cvf boot.tar /boot # 选项 f 必须放在最后！
```

- **外部命令**：具有独立的可执行文件

- **内部命令**：shell 内置命令，没有独立可执行文件

- 命令别名：为命令所取的别名

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ 🔍 ↺

命令搜索顺序

- shell 如何找到我们要执行的命令？
 - ① 是否是别名？
 - ② 是否是内部命令？
 - ③ 是否是外部命令？
 - 从左到右依次搜索 PATH 环境变量包含的路径
 - 一旦在某条路径中找到所要执行的命令，则不再搜索后续路径
 - ④ 找到了则执行命令；找不到则给出提示（一定要仔细看提示！）

选项 a-n 巡礼

Ⅱ 选项	含义
-a	all, append
-b	buffer, block, batch
-c	command, check
-d	debug, delete, directory
-D	define
-e	excute, edit, exclude, expression
-f	file, force
-h	header, help
-i	initialize, interactive
-I	include
-k	keep, kill
-l	list, long, load, login
-m	message, mail, mode, modification-time
-n	number, not

选项 o-z 巡礼

选项	含义
-o	output
-p	port, protocol
-q	quite
-r(-R)	recurse, reverse
-s	silent, subject, size
-t	tag
-u	user
-v	verbose, version
-V	version
-w	width, warning
-x	debug, extract
-y	yes
-z	zip

命令行编辑

- 命令自动完成：Tab 键
- 快捷键

按键	功能
Ctrl-a	移动光标至行首
Ctrl-e	移动光标至行尾
Ctrl-d	向后删除/结束输入
Ctrl-h	向前删除
Ctrl-k	向后删除到行尾
Ctrl-u	向前删除到行首
Ctrl-c	中止执行
Ctrl-p	上一条命令
Ctrl-n	下一条命令

命令历史

- 命令历史: 可用上下箭头翻阅用过的命令
- 命令历史保存在内存中, 用户注销时保存至 `.bash` 文件, 用户登录时从该文件加载至内存。
- 命令历史长度由环境变量 `HISTSIZE` 的值决定

```
echo $HISTSIZE # 打印环境变量 HISTSIZE 的值
history        # 列出所有历史命令
!20            # 执行第 20 条命令
!!            # 执行上一条命令
!-1           # 同上
!ls           # 执行最近一条以 ls 开头的命令
Ctrl-r        # 搜索最近一条包含 doc 的命令
doc            # 按回车执行, 按左右键编辑
history -c     # 清除命令历史
```

屏幕控制

- 向上翻页屏幕输出：Shift-PgUp
- 向下翻页屏幕输出：Shift-PgDn
- 暂停屏幕输出：Ctrl-s
- 恢复屏幕输出：Ctrl-q
- 清屏：clear（快捷键为 Ctrl-l）
- 重置终端：reset

man (manual)

```
man ls          # 查看 ls 的手册页
                # h(获取帮助)
                # q(退出)
man man         # 查看 man 手册的手册页
man passwd
man 5 passwd    # 查看第 5 章的 passwd 手册页
whatis cp       # 查看命令简要描述, 等同于 man -f cp
apropos who     # 根据关键字查询, 等同于 man -k who
```

4.4 获取帮助

help

```
help          # 列出所有内部命令
help cd       # 显示内部命令 cd 的帮助
cat --help    # 显示外部命令 cat 的简明帮助
```

info 多级结构，且支持超链接

```
info          # 打开 info 首页
info who      # 打开 who 的 info 页
#h(获取帮助)H(入门指南)
#q(退出)
```


学习 QQ 群

- 群号：494598324
- 实名验证：班级 + 姓名



群名称: learning-linux
群 号 494598324