[第 1 章 LINUX 开 山篇 1](#_bookmark0)

* 1. [本套 LINUX 课程的内容介绍 1](#_bookmark1)
  2. [LINUX 的学习方向 1](#_bookmark1)
     1. [Linux 运维工程师 1](#_bookmark2)
     2. [Linux 嵌入式开发工程师 2](#_bookmark3)
     3. [在 linux 下做各种程序开发 2](#_bookmark3)
     4. [示意图 2](#_bookmark3)
  3. [LINUX 的应用领域 2](#_bookmark4)
     1. [个人桌面应用领域 2](#_bookmark3)
     2. [服务器应用领域 3](#_bookmark5)
     3. [嵌入式应用领域 3](#_bookmark5)
  4. [学习 LINUX 的阶段（高手进阶过程） 3](#_bookmark6)
  5. [LINUX 的学习方法和建议 4](#_bookmark7)

[第 2 章 基 础 篇 LINUX 入门 5](#_bookmark8)

* 1. [LINUX 介绍 5](#_bookmark9)
  2. [UNIX 是怎么来的 6](#_bookmark10)
  3. [LINUX 是怎么来的 7](#_bookmark11)
  4. [LINUX 和 UNIX 关系一览图 8](#_bookmark12)
  5. [LINUX 和 WINDOWS 比较 9](#_bookmark13)

[第 3 章 基础篇 VM 和 LINUX 系统(CENTOS)安装 11](#_bookmark14)

* 1. [安装 VM 和 CENTOS 11](#_bookmark15)
  2. [VM 软件和 CENTOS 的安装软件 11](#_bookmark15)
  3. [VM 安装的步骤 12](#_bookmark16)
  4. [CENTOS 安装的步骤 12](#_bookmark16)
  5. [CENTOS 的终端使用和联网 12](#_bookmark16)
  6. [VMTOOLS 安装 13](#_bookmark17)

[3.6.1 介绍 13](#_bookmark18)

* 1. [VMTOOLS 的安装和使用 14](#_bookmark19)
     1. [安装 vmtools 的步骤说明 14](#_bookmark20)
     2. [使用 vmtools 来设置 windows 和 linux 的共享文件夹 14](#_bookmark20)
     3. [安装 vmtools 的课堂练习 15](#_bookmark21)

[第 4 章 基础篇 LINUX 的目录结构 16](#_bookmark22)

* 1. [基本介绍 16](#_bookmark23)
  2. [目录结构的具体介绍 16](#_bookmark23)
  3. [LINUX 目录总结一下 18](#_bookmark24)

[第 5 章 实操篇 远程登录 LINUX 系统 20](#_bookmark25)

* 1. [为什么需要远程登录 LINUX 20](#_bookmark26)
     1. [示意图 20](#_bookmark27)

[5.1.2 说明 20](#_bookmark27)

* 1. [远程登录 LINUX-XSHELL5 20](#_bookmark26)
  2. [安装 XSHELL5 并使用 21](#_bookmark28)
     1. [安装过程 21](#_bookmark29)
     2. [XShell5 的关键配置 21](#_bookmark29)
     3. [XShel5 远程登录到 Linux 后，就可以使用指令来操作 Linux 系统 21](#_bookmark29)
  3. [远程上传下载文件 XFTP5 22](#_bookmark30)
     1. [XFtp5 软件介绍 22](#_bookmark31)
     2. [XFtp5 软件的安装 22](#_bookmark31)
     3. [Xftp5 的配置和使用 22](#_bookmark31)
     4. [如何解决 XFTP5 中文乱码的问题 24](#_bookmark32)
  4. [XFTP5 和 XSHELL5 的使用 25](#_bookmark33)

[第 6 章 实操篇 VI 和 VIM 编辑器 26](#_bookmark34)

* 1. [VI 和 VIM 的基本介绍 26](#_bookmark35)
  2. [VI 和 VIM 的三种常见模式 26](#_bookmark35)
     1. [正常模式 26](#_bookmark36)
     2. [插入模式/编辑模式 26](#_bookmark36)
     3. [命令行模式 26](#_bookmark36)
  3. [快速入门案例 26](#_bookmark35)
  4. [VI 和 VIM 三种模式的相互转化图 27](#_bookmark37)
  5. [快捷键的使用案例 28](#_bookmark38)
  6. [VIM 和 VI 的快捷键键盘一览图 29](#_bookmark39)
  7. [VI 和 VIM 课堂练习 29](#_bookmark39)

[第](#_bookmark40) **[7](#_bookmark40)** [章 实操篇 开机、重启和用户登录注销](#_bookmark40) **[30](#_bookmark40)**

* 1. [关机&重启命令 30](#_bookmark41)
     1. [基本介绍 30](#_bookmark42)
     2. [注意细节 30](#_bookmark42)
  2. [用户登录和注销 30](#_bookmark41)
     1. [基本介绍 30](#_bookmark42)
     2. [使用细节 31](#_bookmark43)

[第 8 章 实操篇 用户管理 32](#_bookmark44)

* 1. [基本介绍 32](#_bookmark45)
  2. [添加用户 32](#_bookmark45)
     1. [基本语法 32](#_bookmark46)
     2. [实际案例 32](#_bookmark46)
     3. [细节说明 33](#_bookmark47)
  3. [给用户指定或者修改密码 33](#_bookmark48)
  4. [删除用户 34](#_bookmark49)
     1. [基本语法 34](#_bookmark50)
     2. [应用案例 34](#_bookmark50)
     3. [思考题 34](#_bookmark50)
  5. [查询用户信息 34](#_bookmark49)
     1. [基本语法 34](#_bookmark50)
     2. [应用实例 35](#_bookmark51)
     3. [细节说明 35](#_bookmark51)
  6. [切换用户 35](#_bookmark52)

[8.6.1 介绍 35](#_bookmark51)

* + 1. [基本语法 35](#_bookmark51)
    2. [应用实例 35](#_bookmark51)
    3. [细节说明 36](#_bookmark53)
  1. [用户组 36](#_bookmark54)

[8.7.1 介绍 36](#_bookmark53)

* + 1. [增加组 36](#_bookmark53)
    2. [案例演示 36](#_bookmark53)
    3. [删除组 36](#_bookmark53)
    4. [案例演示 36](#_bookmark53)
  1. [增加用户时直接加上组 37](#_bookmark55)
     1. [指令(基本语法) 37](#_bookmark56)
     2. [案例演示 37](#_bookmark56)
  2. [修改用户的组 37](#_bookmark55)
     1. [指令(基本语法) 37](#_bookmark56)
     2. [案例演示 37](#_bookmark56)
  3. [/ETC/PASSWD 文件 38](#_bookmark57)
  4. [/ETC/SHADOW 文件 38](#_bookmark57)
  5. [/ETC/GROUP 文件 38](#_bookmark57)

[第 9 章 实操篇 实用指令 39](#_bookmark58)

* 1. [指定运行级别 39](#_bookmark59)
  2. [切换到指定运行级别的指令 40](#_bookmark60)
     1. [基本语法 40](#_bookmark61)
     2. [应用实例 40](#_bookmark61)
     3. [面试题 40](#_bookmark61)
     4. [课堂练习 41](#_bookmark62)
  3. [帮助指令 41](#_bookmark63)

[9.3.1 介绍 41](#_bookmark62)

* + 1. [man 获得帮助信息 41](#_bookmark62)
    2. [help 指令 41](#_bookmark62)
    3. [当一个指令不熟悉如何学习的建议 42](#_bookmark64)
  1. [文件目录类 43](#_bookmark65)
     1. [pwd 指令 43](#_bookmark66)
     2. [ls 指令 43](#_bookmark66)
     3. [cd 指令 44](#_bookmark67)
     4. [mkdir 指令 45](#_bookmark68)
     5. [rmdir 指令 46](#_bookmark69)
     6. [touch 指令 47](#_bookmark70)
     7. [cp 指令[重要] 47](#_bookmark70)
     8. [rm 指令 49](#_bookmark71)
     9. [mv 指令 50](#_bookmark72)
     10. [cat 指令 50](#_bookmark73)
     11. [more 指令 51](#_bookmark74)
     12. [less 指令 52](#_bookmark75)

[9.4.13 > 指 令 和 >> 指令 53](#_bookmark76)

* + 1. [echo 指令 55](#_bookmark77)
    2. [head 指令 55](#_bookmark77)
    3. [tail 指令 56](#_bookmark78)
    4. [ln 指令 57](#_bookmark79)
    5. [history 指令 58](#_bookmark80)
  1. [时间日期类 59](#_bookmark81)
     1. [date 指令-显示当前日期 59](#_bookmark82)
     2. [date 指令-设置日期 60](#_bookmark83)
     3. [cal 指令 60](#_bookmark83)
  2. [搜索查找类 61](#_bookmark84)
     1. [find 指令 61](#_bookmark85)
     2. [locate 指令 63](#_bookmark86)
     3. [grep 指令和 管道符号 | 64](#_bookmark87)
  3. [压缩和解压类 64](#_bookmark88)
     1. [gzip/gunzip 指令 64](#_bookmark89)
     2. [zip/unzip 指令 65](#_bookmark90)
     3. [tar 指令 66](#_bookmark91)

[第](#_bookmark92) **[10](#_bookmark92)** [章 实操篇 组管理和权限管理](#_bookmark92) **[69](#_bookmark92)**

* 1. [LINUX 组基本介绍 69](#_bookmark93)
  2. [文件/目录 所有者 69](#_bookmark93)
     1. [查看文件的所有者 69](#_bookmark94)
     2. [修改文件所有者 70](#_bookmark95)
  3. [组的创建 71](#_bookmark96)
     1. [基本指令 71](#_bookmark97)
     2. [应用实例 71](#_bookmark97)
  4. [文件/目录 所在组 71](#_bookmark96)
     1. [查看文件/目录所在组 71](#_bookmark97)
     2. [修改文件所在的组 72](#_bookmark99)
  5. [其它组 72](#_bookmark98)
  6. [改变用户所在组 72](#_bookmark98)
     1. [改变用户所在组 73](#_bookmark100)
     2. [应用实例 73](#_bookmark100)
  7. [权限的基本介绍 73](#_bookmark101)
  8. [RWX 权限详解 74](#_bookmark102)
     1. [rwx 作用到文件 74](#_bookmark103)
     2. [rwx 作用到目录 74](#_bookmark103)
  9. [文件及目录权限实际案例 74](#_bookmark102)
  10. [修改权限-CHMOD 75](#_bookmark104)
      1. [基本说明： 75](#_bookmark105)
      2. [第一种方式：+ 、-、= 变更权限 75](#_bookmark105)
      3. [第二种方式：通过数字变更权限 76](#_bookmark106)
  11. [修改文件所有者-CHOWN 77](#_bookmark107)
      1. [基本介绍 77](#_bookmark108)
      2. [案例演示： 77](#_bookmark108)
  12. [修改文件所在组-CHGRP 78](#_bookmark109)
      1. [基本介绍 78](#_bookmark110)
      2. [案例演示： 78](#_bookmark110)
  13. [最佳实践-警察和土匪游戏 78](#_bookmark109)
  14. [课后练习 80](#_bookmark111)
  15. [课后练习题 2 80](#_bookmark111)

[第 11 章 实操篇 CROND 任务调度 82](#_bookmark112)

[11.1 原理示意图 82](#_bookmark113)

[11.2 概述 82](#_bookmark114)

* 1. [基本语法 83](#_bookmark115)
     1. [常用选项 83](#_bookmark116)
  2. [快速入门 83](#_bookmark117)
     1. [任务的要求 83](#_bookmark116)
     2. [步骤如下 83](#_bookmark116)
     3. [参数细节说明 83](#_bookmark116)
  3. [任务调度的几个应用实例 84](#_bookmark118)
     1. [案例 1：每隔 1 分钟，就将当前的日期信息，追加到 /tmp/mydate 文件中 85](#_bookmark119)
     2. [案例 2：每隔 1 分钟， 将当前日期和日历都追加到 /home/mycal 文件中 85](#_bookmark119)
     3. [案例 3: 每天凌晨 2:00 将 mysql 数据库 testdb ，备份到文件中 mydb.bak。 85](#_bookmark119)
  4. [CROND 相关指令 86](#_bookmark120)

[第 12 章 实操篇 LINUX 磁盘分区、挂载 87](#_bookmark121)

* 1. [分区基础知识 87](#_bookmark122)
     1. [分区的方式： 87](#_bookmark123)
     2. [windows 下的磁盘分区 87](#_bookmark123)
  2. [LINUX 分区 88](#_bookmark124)
     1. [原理介绍 88](#_bookmark125)
     2. [硬盘说明 89](#_bookmark126)
     3. [使用 lsblk 指令查看当前系统的分区情况 89](#_bookmark126)
  3. [挂载的经典案例 90](#_bookmark127)
     1. [如何增加一块硬盘 90](#_bookmark128)
  4. [具体的操作步骤整理 91](#_bookmark129)
     1. [虚拟机增加硬盘步骤 1 91](#_bookmark130)
     2. [虚拟机增加硬盘步骤 2 91](#_bookmark130)
     3. [虚拟机增加硬盘步骤 3 92](#_bookmark131)
     4. [虚拟机增加硬盘步骤 4 92](#_bookmark131)
     5. [虚拟机增加硬盘步骤 5 92](#_bookmark131)
  5. [磁盘情况查询 93](#_bookmark132)
     1. [查询系统整体磁盘使用情况 93](#_bookmark133)
     2. [查询指定目录的磁盘占用情况 93](#_bookmark133)
  6. [磁盘情况-工作实用指令 94](#_bookmark134)

[第 13 章 实操篇 网络配置 96](#_bookmark135)

* 1. [LINUX 网络配置原理图(含虚拟机) 96](#_bookmark136)
  2. [查看网络 IP 和网关 96](#_bookmark136)
     1. [查看虚拟网络编辑器 96](#_bookmark137)
     2. [修改 ip 地址(修改虚拟网络的 ip) 96](#_bookmark137)
     3. [查看网关 97](#_bookmark138)
     4. [查看 windows 环境的中 VMnet8 网络配置 (ipconfig 指令) 97](#_bookmark138)
  3. [PING 测试主机之间网络连通 98](#_bookmark139)
     1. [基本语法 98](#_bookmark140)
     2. [应用实例 98](#_bookmark140)
  4. [LINUX 网络环境配置 98](#_bookmark139)
     1. [第一种方法(自动获取) 98](#_bookmark140)
     2. [第二种方法(指定固定的 ip) 99](#_bookmark141)

[第 14 章 实操篇 进程管理 101](#_bookmark142)

* 1. [进程的基本介绍 101](#_bookmark143)
  2. [显示系统执行的进程 101](#_bookmark143)

[14.2.1 说 明： 101](#_bookmark144)

* + 1. [ps 指令详解 102](#_bookmark145)
    2. [应用实例 103](#_bookmark146)
  1. [终止进程 KILL 和 KILLALL 104](#_bookmark147)

[14.3.1 介绍 104](#_bookmark148)

[14.3.2 基本语法： 104](#_bookmark148)

[14.3.3 常用选项： 104](#_bookmark148)

[14.3.4 最佳实践： 104](#_bookmark148)

* 1. [查看进程树 PSTREE 106](#_bookmark149)

[14.4.1 基本语法： 106](#_bookmark150)

[14.4.2 常用选项： 106](#_bookmark150)

[14.4.3 应用实例： 106](#_bookmark150)

* 1. [服务(SERVICE)管理 106](#_bookmark149)

[14.5.1 介绍 107](#_bookmark151)

* + 1. [service 管理指令： 107](#_bookmark151)

[14.5.3 使用案例： 107](#_bookmark151)

[14.5.4 细节讨论： 108](#_bookmark152)

* + 1. [查看服务名 108](#_bookmark152)
    2. [服务的运行级别(runlevel) 109](#_bookmark153)
    3. [开机的流程说明 110](#_bookmark154)
    4. [chkconfig 指令 110](#_bookmark154)

[介绍 110](#_bookmark155)

[通过 chkconfig 命令可以给每个服务的各个运行级别设置自启动/关闭 110](#_bookmark155)

[基本语法 110](#_bookmark155)

1. [查看服务 chkconfig --list|grep xxx 110](#_bookmark156)
2. [chkconfig 服务名 --list 110](#_bookmark156)
3. [chkconfig --level 5 服务名 on/off 110](#_bookmark156)

[14.5.9 应用实例： 111](#_bookmark157)

[14.5.10 • 使 用 细节 111](#_bookmark157)

* 1. [动态监控进程 111](#_bookmark158)

[14.6.1 介 绍： 111](#_bookmark157)

[14.6.2 基本语法： 112](#_bookmark159)

[14.6.3 选项说明： 112](#_bookmark159)

[14.6.4 应用实例： 112](#_bookmark159)

* + 1. [查看系统网络情况 netstat(重要) 114](#_bookmark160)

[第 15 章 实 操 篇 RPM 和 YUM 115](#_bookmark161)

* 1. [RPM 包的管理 115](#_bookmark162)

[15.1.1 介 绍： 115](#_bookmark163)

* + 1. [rpm 包的简单查询指令： 115](#_bookmark163)
    2. [rpm 包名基本格式： 115](#_bookmark163)
    3. [rpm 包的其它查询指令： 116](#_bookmark164)
    4. [卸载 rpm 包： 118](#_bookmark165)
    5. [安装 rpm 包： 118](#_bookmark165)

[15.2 YUM 119](#_bookmark166)

[15.2.1 介 绍： 119](#_bookmark167)

* + 1. [yum 的基本指令 120](#_bookmark168)
    2. [yum 应用实例： 120](#_bookmark168)

[第 16 章 JAVAEE 定制篇 搭建 JAVAEE 环境 122](#_bookmark169)

[16.1 概述 122](#_bookmark170)

[16.1.1 示意图： 122](#_bookmark171)

* 1. [安装 JDK 122](#_bookmark170)
     1. [看老师演示，一会整理笔记： 122](#_bookmark171)
     2. [安装步骤 122](#_bookmark171)
     3. [测试是否安装成功 123](#_bookmark172)
  2. [安装 TOMCAT 124](#_bookmark173)

[16.3.1 步骤 124](#_bookmark174)

[16.3.2 测试是否安装成功： 125](#_bookmark175)

* 1. [ECLIPSE 的安装 126](#_bookmark176)

[16.4.1 步骤 126](#_bookmark177)

* 1. [MYSQL 的安装和配置 127](#_bookmark178)
     1. [安装的步骤和文档 127](#_bookmark179)

[16.5.2 说明 127](#_bookmark179)

[第 17 章 大数据定制篇 SHELL 编程 128](#_bookmark180)

* 1. [为什么要学习 SHELL 编程 128](#_bookmark181)
  2. [SHELL 是什么 128](#_bookmark181)
  3. [SHELL 编程快速入门-SHELL 脚本的执行方式 128](#_bookmark181)
     1. [脚本格式要求 128](#_bookmark182)
     2. [编写第一个 Shell 脚本 129](#_bookmark183)
     3. [脚本的常用执行方式 129](#_bookmark183)
  4. [SHELL 的变量 130](#_bookmark184)
     1. [Shell 的变量的介绍 130](#_bookmark185)
     2. [shell 变量的定义 130](#_bookmark185)
     3. [•定义变量的规则 131](#_bookmark186)
     4. [•将命令的返回值赋给变量（重点） 131](#_bookmark186)
  5. [设置环境变量 132](#_bookmark187)
     1. [基本语法 132](#_bookmark188)
     2. [快速入门 132](#_bookmark188)
  6. [位置参数变量 134](#_bookmark189)

[17.6.1 介绍 134](#_bookmark190)

* + 1. [基本语法 134](#_bookmark190)
    2. [位置参数变量应用实例 134](#_bookmark190)
  1. [预定义变量 135](#_bookmark191)
     1. [基本介绍 135](#_bookmark192)
     2. [基本语法 135](#_bookmark192)
     3. [应用实例 135](#_bookmark192)

[17.8 运 算符 135](#_bookmark191)

* + 1. [基本介绍 135](#_bookmark192)
    2. [基本语法 135](#_bookmark192)
  1. [条件判断 137](#_bookmark193)

[17.9.1 •基本语法 137](#_bookmark194)

[17.9.2 •应用实例 137](#_bookmark194)

* + 1. [•常用判断条件 137](#_bookmark194)
    2. [应用实例 138](#_bookmark195)
  1. [流程控制 139](#_bookmark196)
     1. [if 判断 139](#_bookmark197)
     2. [case 语句 140](#_bookmark198)
     3. [for 循环 141](#_bookmark199)
     4. [while 循环 143](#_bookmark200)
  2. [READ 读取控制台输入 143](#_bookmark201)
     1. [基本语法 143](#_bookmark200)
     2. [应用实例 144](#_bookmark202)

[17.12 函数 144](#_bookmark203)

* + 1. [函数介绍 144](#_bookmark202)
    2. [系统函数 144](#_bookmark202)

[17.12.3 •应用实例 145](#_bookmark204)

[17.12.4 自定义函数 145](#_bookmark204)

* 1. [SHELL 编程综合案例 146](#_bookmark205)

[第 18 章 PYTHON 定制篇 开发平台 UBUNTU 149](#_bookmark206)

* 1. [UBUNTU 的介绍 149](#_bookmark207)
  2. [UBUNTU 的安装 149](#_bookmark207)
     1. [安装的步骤 149](#_bookmark208)
     2. [设置 Ubuntu 支持中文 149](#_bookmark208)
  3. [UBUNTU 的 ROO 用户 150](#_bookmark209)

[18.3.1 介绍 151](#_bookmark210)

[18.3.2 给 root 用户设置密码并使用 151](#_bookmark210)

* 1. [UBUNTU 下开发 PYTHON 151](#_bookmark211)

[18.4.1 说明 151](#_bookmark210)

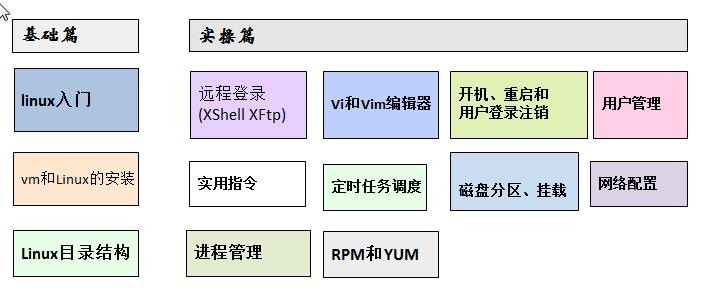
[18.4.2 在 Ubuntu 下开发一个 Python 程序 152](#_bookmark212)

[第 19 章 PYTHON 定制篇 APT 软件管理和远程登录 153](#_bookmark213)

* 1. [APT 介绍 153](#_bookmark214)
  2. [UBUNTU 软件操作的相关命令 153](#_bookmark214)
  3. [更新 UBUNTU 软件下载地址 154](#_bookmark215)
     1. [原理示意图 154](#_bookmark216)
     2. [寻找国内镜像源 154](#_bookmark216)
     3. [备份 Ubuntu 默认的源地址 156](#_bookmark217)
     4. [更新源服务器列表 157](#_bookmark218)
  4. [UBUNTU 软件安装，卸载的最佳实践 157](#_bookmark219)
     1. [案例说明：使用 apt 完成安装和卸载 vim 软件，并查询 vim 软件的信息： 157](#_bookmark218)
  5. [使用 SSH 远程登录 UBUNTU 159](#_bookmark220)
     1. [ssh 介绍 159](#_bookmark221)
     2. [原理示意图： 159](#_bookmark221)
  6. [使用 SSH 远程登录 UBUNTU 160](#_bookmark222)
     1. [安装 SSH 和启用 160](#_bookmark223)
     2. [在 Windows 使用 XShell5/XFTP5 登录 Ubuntu 160](#_bookmark223)
     3. [从 linux 系统客户机远程登陆 linux 系统服务机 161](#_bookmark224)

# 第 1 章 Linux 开山篇

## 本套 Linux 课程的内容介绍



## Linux 的学习方向

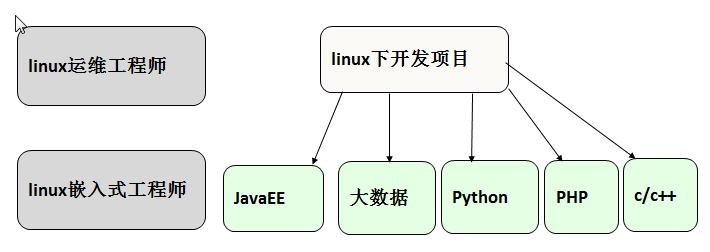
### Linux 运维工程师

### Linux 嵌入式开发工程师

### 在 linux 下做各种程序开发

### 示意图

1.



3 Linux 的应用领域

1.3.1个人桌面应用领域

此领域是传统 linux 应用最薄弱的环节，传统 linux 由于界面简单、操作复杂、应用软件少的缺点， 一直被 windows 所压制，但近些年来随着 ubuntu、fedora [fɪˈdɔ:rə] 等优秀桌面环境的兴起，同时各大硬件厂商对其支持的加大，linux 在个人桌面领域的占有率在逐渐的提高。



### 服务器应用领域

linux 在服务器领域的应用是最强的。

linux 免费、稳定、高效等特点在这里得到了很好的体现，近些年来 linux 服务器市场得到了飞速的提升，尤其在一些高端领域尤为广泛。

### 嵌入式应用领域

近些年来 linux 在嵌入式领域的应用得到了飞速的提高

linux 运行稳定、对网络的良好支持性、低成本，且可以根据需要进行软件裁剪，内核最小可以达到几百 KB 等特点，使其近些年来在嵌入式领域的应用得到非常大的提高

主要应用：机顶盒、数字电视、网络电话、程控交换机、手机、PDA、智能家居、智能硬件等都是其应用领域。以后再物联网中应用会更加广泛

## 学习 Linux 的阶段（高手进阶过程）

linux 是一个开源、免费的操作系统，其稳定性、安全性、处理多并发已经得到

业界的认可，目前很多中型，大型甚至是集群项目都在使用 linux,很多软件公司考虑到开发成本都首选 linux,在中国软件公司得到广泛的使用。

我个人认为学习 linux 流程为:

第 1 阶段：linux 环境下的基本操作命令，包括 文件操作命令(rm mkdir chmod, chown) 编辑工具使用（vi vim）linux 用户管理(useradd userdel usermod)等

第 2 阶段：linux 的各种配置（环境变量配置，网络配置，服务配置）

第 3 阶段：linux 下如何搭建对应语言的开发环境（大数据，JavaEE, Python 等） 第 4 阶段：能编写 shell 脚本，对 Linux 服务器进行维护。

第 5 阶段：能进行安全设置，防止攻击，保障服务器正常运行，能对系统调优。

第 6 阶段：深入理解 Linux 系统（对内核有研究），熟练掌握大型网站应用架构组成、并熟悉各个环节的部署和维护方法。

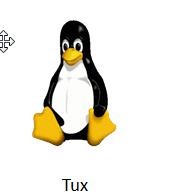
## Linux 的学习方法和建议

1. 高效而愉快的学习
2. 先建立一个整体框架，然后细节
3. 不需要掌握所有的 Linux 指令，要学会查询手册和百度
4. 先 know how ,再 know why
5. 计算机是一门”做中学” 的学科 ,不是会了再做，而是做了才会.
6. 适当的囫囵吞枣
7. Linux 不是编程，重点是实际操作，各种常用指令要玩的溜

# 第 2 章 基础篇 Linux 入门

## Linux 介绍

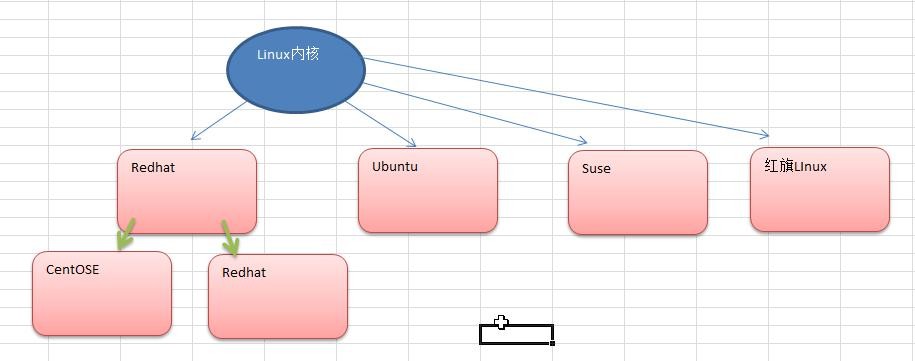
* + 1. Linux 怎么读 【里纽克斯，利尼克斯，里纳克斯】
    2. Linux 是一款操作系统，免费，开源， 安全，高效，稳定， 处理高并发非常强悍，现在很多的企业级的项目都部署到 Linux/unix 服务器运行。
    3. Linux 创始人-linux 林纳斯



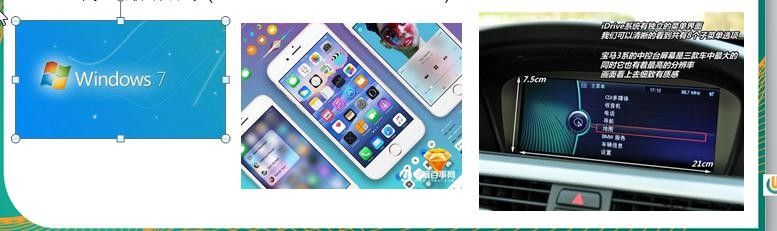
4) Linux 的吉祥物

企 鹅 tux

1. Linux 的主要的发行版

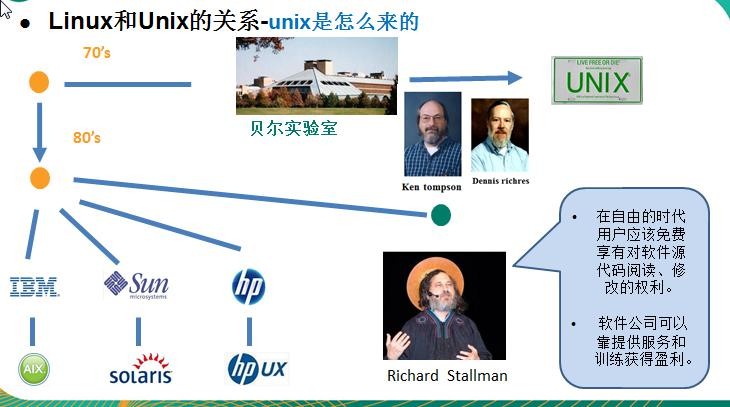


1. 目前主要的操作系统有



windows, android ，车载系统,linux 等。

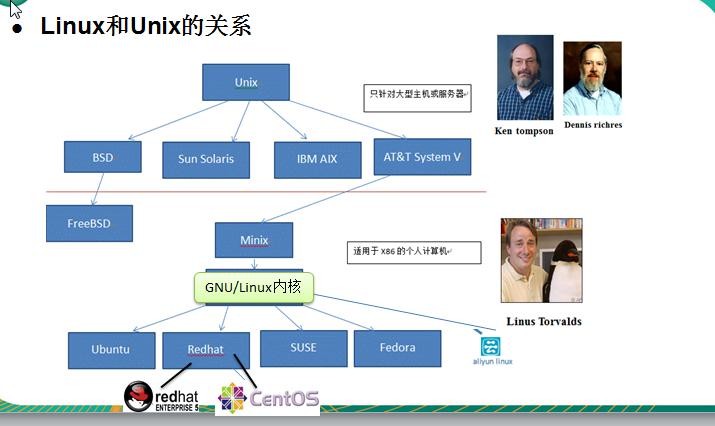
## unix 是怎么来的



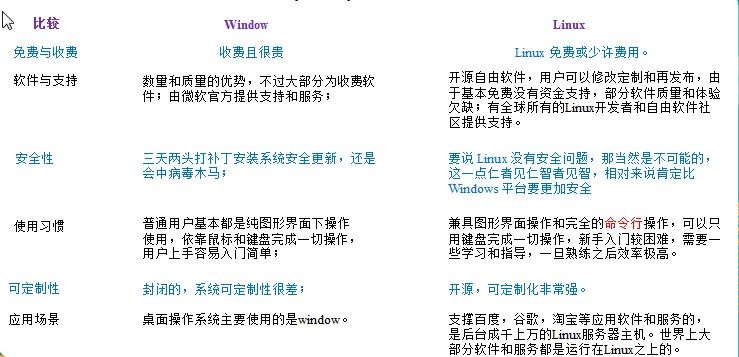
Linux 是怎么来的



linux 和 unix 关系一览图



linux 和 Windows 比较



# 第 3 章 基础篇 VM 和 Linux 系统(CentOS)安装

* 1. 安装 **vm** 和 **Centos**

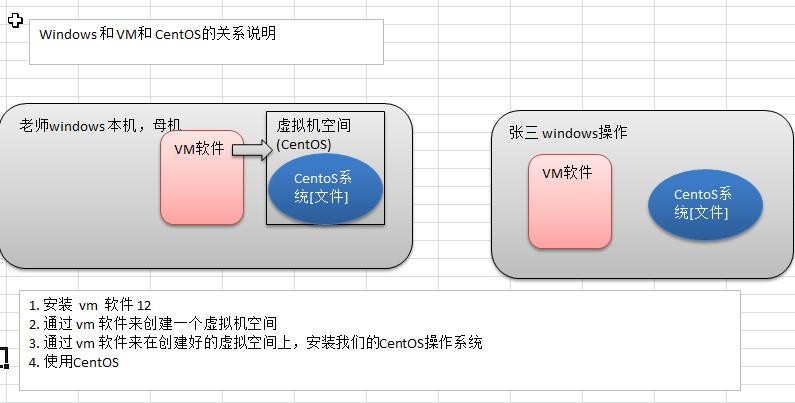
学习 Linux 需要一个环境，我们需要创建一个虚拟机，然后在虚拟机上安装一个 Centos 系统来学

习。

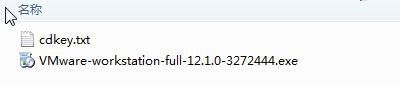
1)先安装 virtual machine ,vm12

2)再安装 Linux (CentOS 6.8)

3)原理示意图，这里我们画图说明一下 VM 和 CentOS 的关系。



## vm 软件和 CentOS 的安装软件



* 1. **VM** 安装的步骤
     1. 去 BIOS 里修改设置开启虚拟化设备支持（f2, f10）
     2. 安装虚拟机软件（vm12）

看老师的具体演示：因为老师的电脑上已经安装了 VM12，因此我先卸载然后再安装。

* 1. **CentOS** 安装的步骤

1. 创建虚拟机(空间)



这里在配置网络连接时，有三种形式，需要大家伙注意 ：

1. 开始安装系统(CentOS6.8)
2. 看老师安装的演示过程。

## CentOS 的终端使用和联网

* + 1. 终端的使用，点击鼠标右键，即可选择打开终端



* + 1. 配置网络，可以上网。

点击上面右侧的；两个计算机图片，选择启用 eth0,即可成功连接到网络，就可以上网。



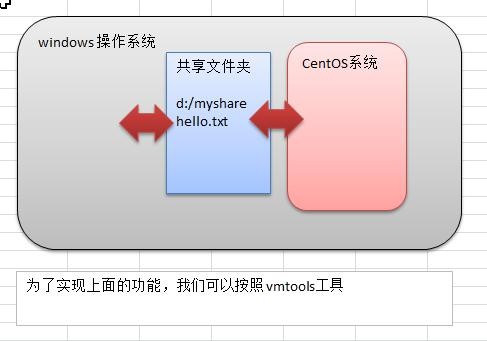
## vmtools 安装

### 3.6.1介绍

1)可以直接粘贴命令在 windows 和 centos 系统之间

2)可以设置 windows 和 centos 的共享文件夹

3)示意图



## vmtools 的安装和使用



3.7.1安装 vmtools 的步骤说明

1.进入 centos

2.点击 vm 菜单的->install vmware tools

3.centos 会出现一个 vm 的安装包

4.点击右键解压, 得到一个安装文件

5.进入该 vm 解压的目录 ，该文件在 /root/桌面/vmware-tools-distrib/下

6.安装 ./vmware-install.pl

7.全部使用默认设置即可

8.需要 reboot 重新启动即可生效

### 使用 vmtools 来设置 windows 和 linux 的共享文件夹

* + - 1. 菜单->vm->setting, 如图设置即可注意:设置选项为 always enable , 这样可以读写了
      2. windows 和 centos 可共享 d:/share 目录可以读写文件了

3) 在 centos 的 /mnt/hgfs/ 下

### 安装 vmtools 的课堂练习



5min



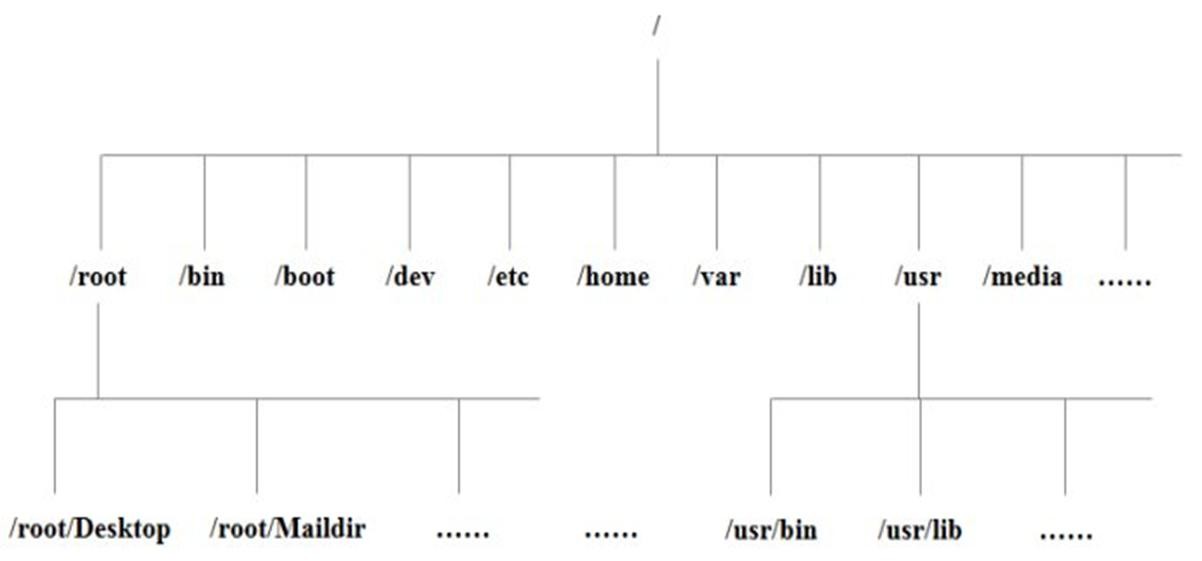
# 第 4 章 基础篇 Linux 的目录结构

## 基本介绍

linux 的文件系统是采用级层式的树状目录结构，在此结构中的最上层是根目录“/”，然后在此目录下再创建其他的目录。

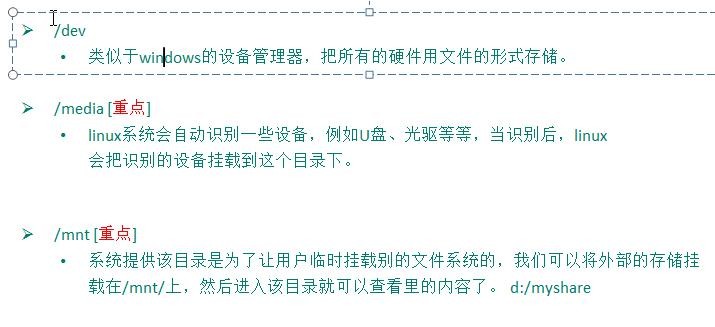
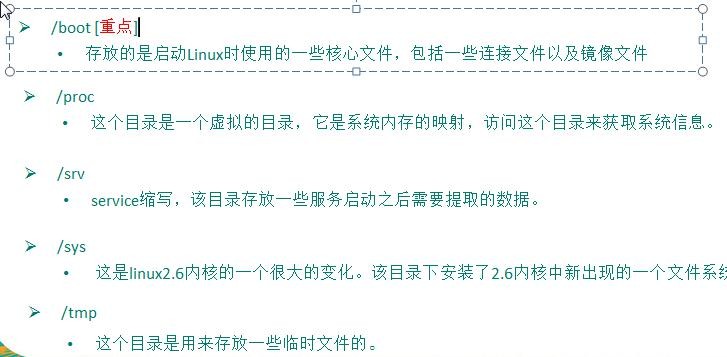
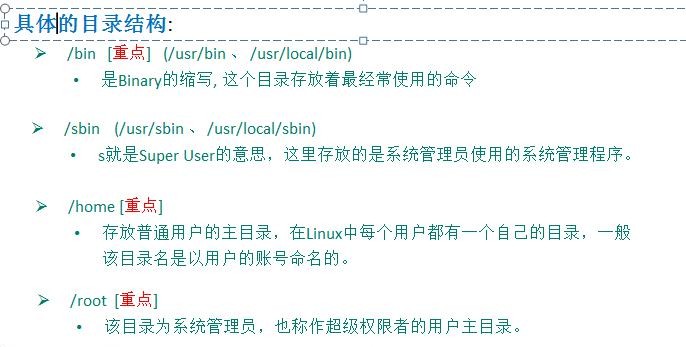
深刻理解 linux 树状文件目录是非常重要的，这里我给大家说明一下。

记住一句经典的话：在 Linux 世界里，一切皆文件。





## 目录结构的具体介绍



# 第 6 章 实操篇 vi 和 vim 编辑器



## 快捷键的使用案例

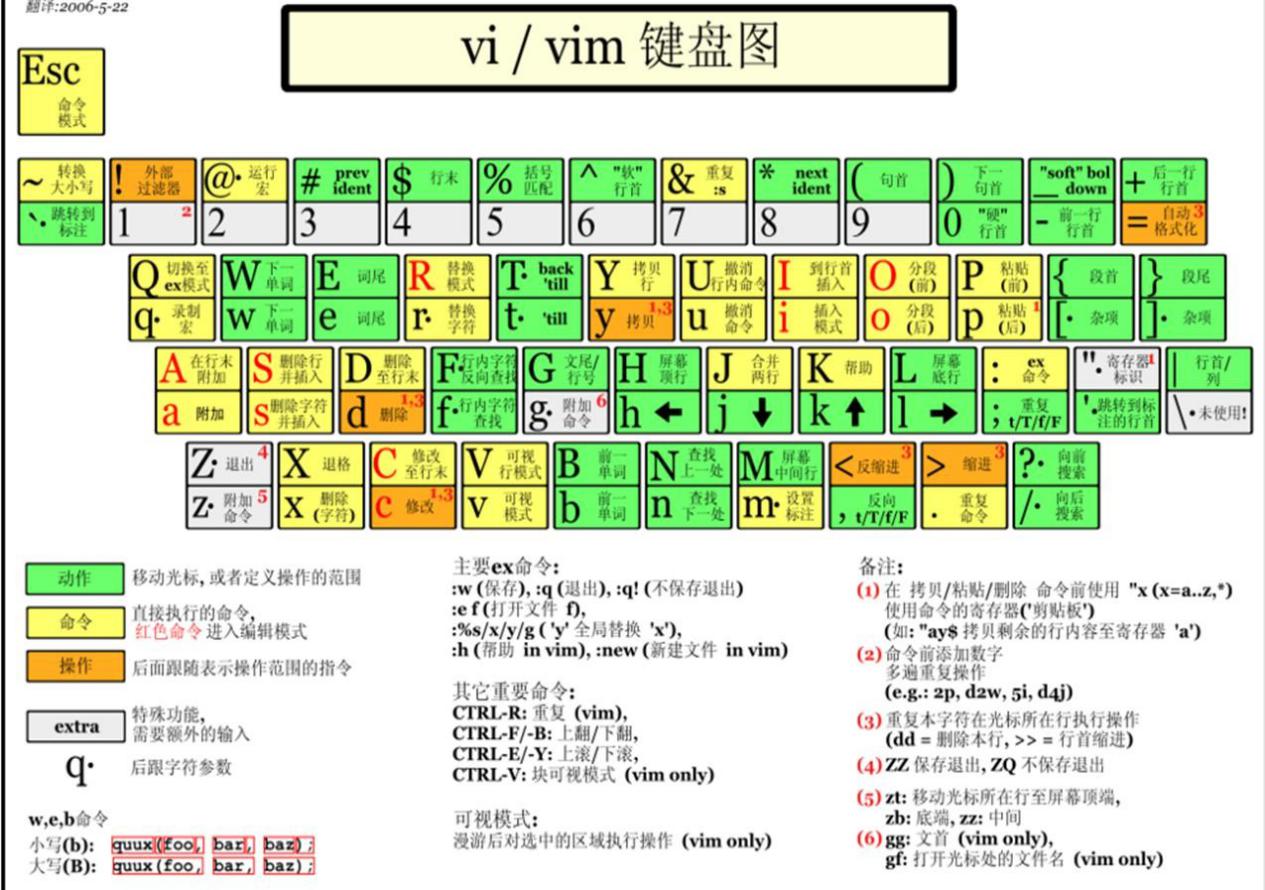
1. 拷贝当前行 yy , 拷贝当前行向下的 5 行 5yy，并粘贴（p）。
2. 删除当前行 dd , 删除当前行向下的 5 行 5dd
3. 在文件中查找某个单词 [命令行下 /关键字 ， 回车 查找 , 输入 n 就是查找下一个 ],查询

hello.

1. 设置文件的行号，取消文件的行号.[命令行下 : set nu 和 :set nonu]
2. 编辑 /etc/profile 文件，使用快捷键到底文档的最末行[G]和最首行[gg],注意这些都是在正常模式下执行的。
3. 在一个文件中输入 "hello" ,然后又撤销这个动作，再正常模式下输入 u
4. 编辑 /etc/profile 文件，并将光标移动到 第 20 行 shift+g

第一步：显示行号 :set nu 第二步：输入 20 这个数第三步: 输入 shift+g

## vim 和 vi 的快捷键键盘一览图



## vi 和 vim 课堂练习

# 第 7 章 实操篇 开机、重启和用户登录注销

## 关机&重启命令

### 基本介绍

shutdown

shutdown -h now : 表示立即关机shutdown -h 1 : 表示 1 分钟后关机shutdown -r now: 立即重启

halt



就是直接使用，效果等价于关机

reboot

就是重启系统。

sync ： 把内存的数据同步到磁盘

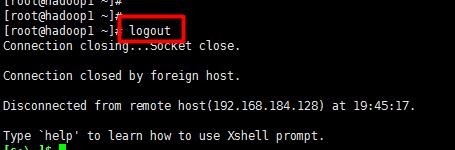
### 注意细节

当我们关机或者重启时，都应该先执行以下 sync 指令，把内存的数据写入磁盘，防止数据丢失。

## 用户登录和注销

### 基本介绍

* + - 1. 登录时尽量少用 root 帐号登录，因为它是系统管理员，最大的权限，避免操作失误。可以利用普通用户登录，登录后再用”su - 用户名’命令来切换成系统管理员身份.
      2. 在提示符下输入 logout 即可注销用户



### 使用细节

* + - 1. logout 注销指令在图形运行级别(5)无效，在 运行级别 3 下有效.

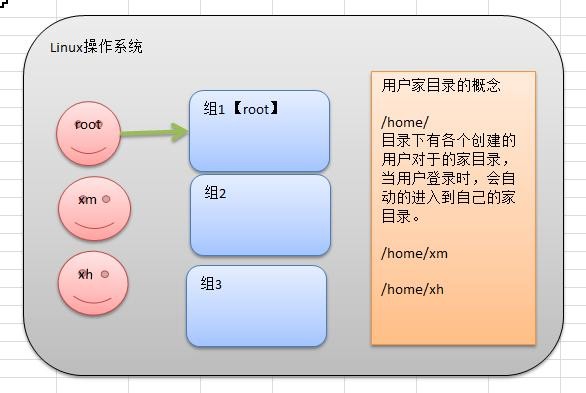
2)运行级别这个概念，后面给大家介绍



# 第 8 章 实操篇 用户管理

## 基本介绍

给大家画一个示意图，帮助大家理解用户管理的规则。



说明

* + 1. 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。
    2. Linux 的用户需要至少要属于一个组。

## 添加用户

### 基本语法

useradd [选项] 用户名

### 实际案例

添加一个用户 xm.



特别说明：

cd 表示 change directory, 切换目录.

### 细节说明



1. 当创建用户成功后，会自动的创建和用户同名的家目录
2. 也可以通过 useradd -d 指定目录 新的用户名，给新创建的用户指定家目录

## 给用户指定或者修改密码

基本语法

passwd 用户名应用案例

1) 给 xiaoming 指定密码



## 删除用户

### 基本语法

userdel 用户名



* + 1. 应用案例
       1. 删除用户 xm，但是要保留家目录

2) 删除用户 xh 以及用户主目录

### 8.4.3思考题

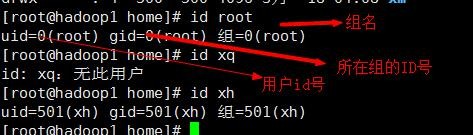
在删除用户时，我们一般不会将家目录删除。

## 查询用户信息

### 基本语法

id 用户名

### 应用实例

案例 1：请查询 root 信息

### 细节说明

* + - 1. 当用户不存在时，返回”无此用户”

## 切换用户

### 介绍

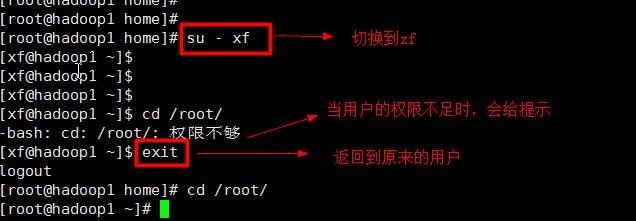
在操作 Linux 中，如果当前用户的权限不够，可以通过 su - 指令，切换到高权限用户，比如 root

### 基本语法

su – 切换用户名

### 应用实例

* + - 1. 创建一个用户 zf, ，指定密码，然后切换到 zf.



### 细节说明

1)从权限高的用户切换到权限低的用户，**不需要输入密码**，反之需要。

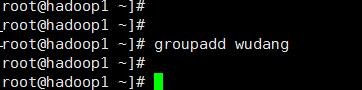
2)当需要返回到原来用户时，使用 exit 指令

## 用户组

### 介绍

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。

### 增加组



groupadd 组 名

8.7.3案例演示

### 删除组

指令(基本语法)

groupdel 组 名

### 案例演示

## 增加用户时直接加上组

### 指令(基本语法)

useradd -g 用户组 用户名

### 案例演示

增加一个用户 zwj, 直接将他指定到 wudang

步骤看演示：

## 8.9

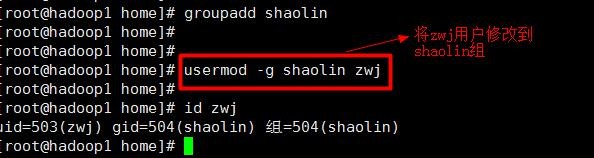


修改用户的组

### 指令(基本语法)

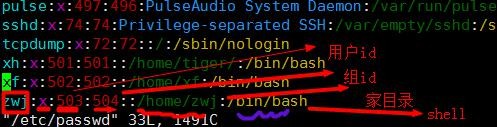
usermod -g 用户组 用户名

### 案例演示

创建一个 shaolin 组，让将 zwj 用户修改到 shaolin

## /etc/passwd 文件

用户（user）的配置文件，记录用户的各种信息

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

## /etc/shadow 文件

口令的配置文件

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

## /etc/group 文件

组(group)的配置文件，记录 Linux 包含的组的信息每行含义：组名:口令:组标识号:组内用户列表

# 第 9 章 实操篇 实用指令

## 指定运行级别

运行级别说明：

1. ：关机
2. ：单用户【找回丢失密码】

2：多用户状态没有网络服务

3：多用户状态有网络服务

4：系统未使用保留给用户



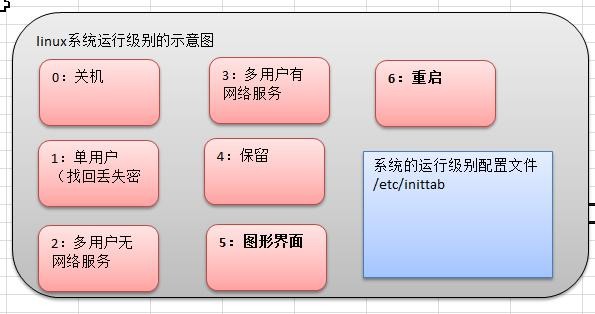
5：图形界面

6：系统重启q

常用运行级别是 3 和 5 ，要修改默认的运行级别可改文件

/etc/inittab 的 id:5:initdefault:这一行中的数字

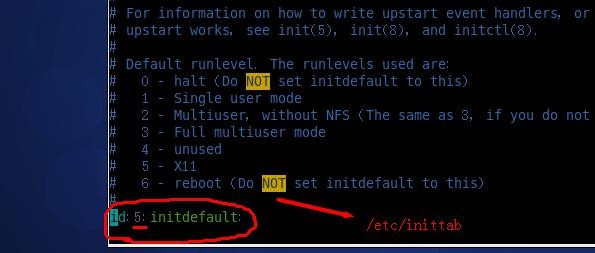
运行级别的示意图：



## 切换到指定运行级别的指令

### 9.2.1基本语法

init [012356]



9.2.2应用实例

案例 1： 通过 init 来切换不同的运行级别，比如动 5->3 ， 然后关机

init 3

init 5

init 0

### 面试题

如何找回 root 密码，如果我们不小心，忘记 root 密码，怎么找回。

思路： 进入到 单用户模式，然后修改 root 密码。因为进入单用户模式，root 不需要密码就可以登录。

演示一把（注意观察）：

总结

开机->在引导时输入 回车键-> 看到一个界面输入 e -> 看到一个新的界面，选中第二行（编辑内核）在输入 e-> 在这行最后输入 1 ,再输入 回车键->再次输入 b ,这时就会进入到单用户模式。

这时，我们就进入到单用户模式，使用 passwd 指令来修改 root 密码。

### 课堂练习:

* + - 1. 假设我们的 root 密码忘记了，请问如何找回密码
      2. 请设置我们的 运行级别，linux 运行后，直接进入到 命令行界面，即进入到 3 运行级别

vim /etc/inittab

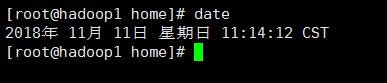
将 id:5:initdefault:这一行中的数字, 5 这个数字改成对应的运行级别即可。

## 时间日期类

### date 指令-显示当前日期

* + - * 基本语法

1) date （功能描述：显示当前时间）



1. date +%Y
2. date +%m
3. date +%d

（功能描述：显示当前年份）

（功能描述：显示当前月份）

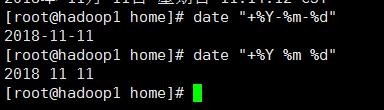
（功能描述：显示当前是哪一天）

5) date "+%Y-%m-%d %H:%M:%S"（功能描述：显示年月日时分秒）

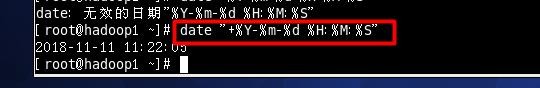
* 应用实例

案例 1: 显示当前时间信息

案例 2: 显示当前时间年月日



案例 3: 显示当前时间年月日时分秒

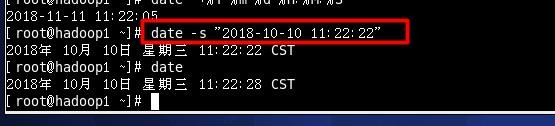


### date 指令-设置日期

* + - * 基本语法

date -s 字符串时间

* + - * 应用实例



案例 1: 设置系统当前时间 ， 比如设置成 2018-10-10 11:22:22

### cal 指令

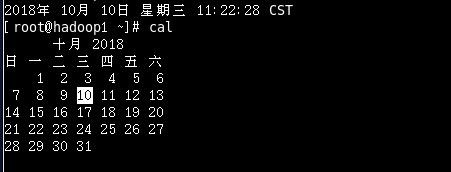
查看日历指令

* + - * 基本语法

cal [选项] （功能描述：不加选项，显示本月日历）

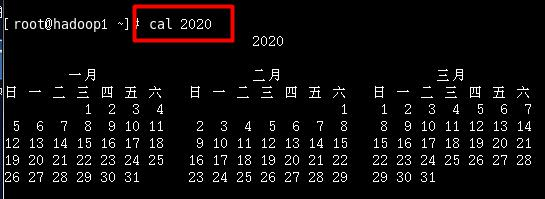
* + - * 应用实例

案例 1: 显示当前日历



案例 2: 显示 2020 年日历

## 搜索查找类



### find 指令

find 指令将从指定目录向下递归地遍历其各个子目录，将满足条件的文件或者目录显示在终端。

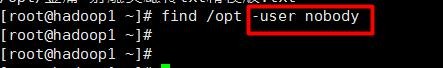
* + - * 基本语法

find [搜索范围] [选项]

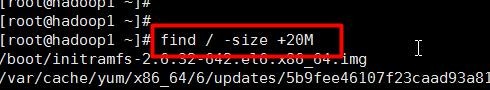
* + - * 选项说明
      * 应用实例

案例 1: 按文件名：根据名称查找/home 目录下的 hello.txt 文件

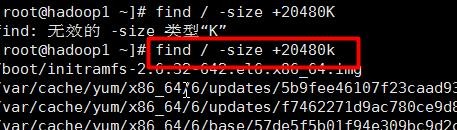


案例 2：按拥有者：查找/opt 目录下，用户名称为 nobody 的文件

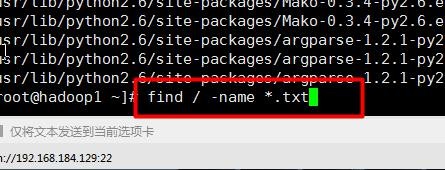
案例 3：查找整个 linux 系统下大于 20m 的文件（+n 大于 -n 小于 n 等于）



1m = 1024k



查询 / 目录下，所有 .txt 的文件



### locate 指令

locaate 指令可以快速定位文件路径。locate 指令利用事先建立的系统中所有文件名称及路径的locate 数据库实现快速定位给定的文件。Locate 指令无需遍历整个文件系统，查询速度较快。为了保证查询结果的准确度，管理员必须定期更新 locate 时刻。

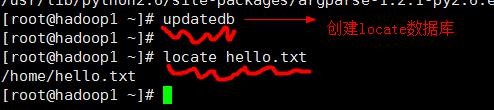
* + - * 基本语法

locate 搜索文件

* + - * 特别说明

由于 locate 指令基于数据库进行查询，所以第一次运行前，必须使用 updatedb 指令创建 locate 数据库。

* + - * 应用实例

案例 1: 请使用 locate 指令快速定位 hello.txt 文件所在目录

## 压缩和解压类

### gzip/gunzip 指令

gzip 用于压缩文件， gunzip 用于解压的

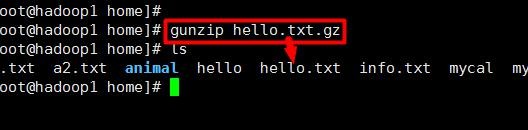
* 基本语法

gzip 文件 （功能描述：压缩文件，只能将文件压缩为\*.gz 文件）

gunzip 文 件.gz （功能描述：解压缩文件命令）

* 应用实例

案例 1: gzip 压缩， 将 /home 下的 hello.txt 文件进行压缩



案例 2: gunzip 压缩， 将 /home 下的 hello.txt.gz 文件进行解压缩

细节说明

当我们使用 gzip 对文件进行压缩后，不会保留原来的文件。

### zip/unzip 指令

zip 用于压缩文件， unzip 用于解压的，这个在项目打包发布中很有用的

* + - * 基本语法

zip [选项] XXX.zip 将要压缩的内容（功能描述：压缩文件和目录的命令）

unzip [选项] XXX.zip （功能描述：解压缩文件）

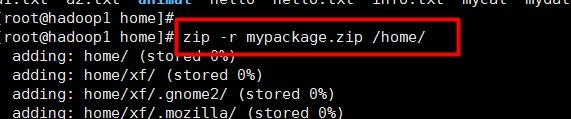
* + - * zip 常用选项

-r：递归压缩，即压缩目录

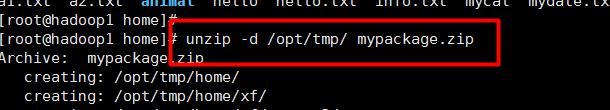
* + - * unzip 的常用选项

-d<目录> ：指定解压后文件的存放目录

* + - * 应用实例

案例 1: 将 /home 下的 所有文件进行压缩成 mypackage.zip





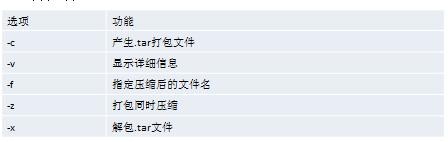
案例 2: 将 mypackge.zip 解压到 /opt/tmp 目录下

### tar 打包指令

* + - * 基本语法

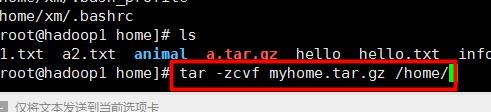
tar [选项] XXX.tar.gz 打包的内容 (功能描述：打包目录，压缩后的文件格式.tar.gz)

* + - * 选项说明



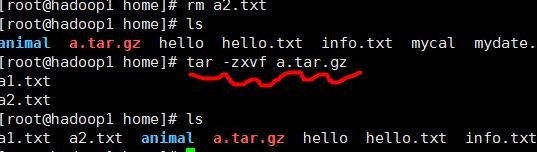
* + - * 应用实例

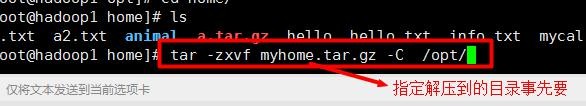
案例 1: 压缩多个文件，将 /home/a1.txt 和 /home/a2.txt 压缩成 a.tar.gz



案例 2: 将/home 的文件夹 压缩成 myhome.tar.gz

案例 3: 将 a.tar.gz 解压到当前目录

案 例 4: 将 myhome.tar.gz 解压到 /opt/ 目录下



指定解压到的那个目录，事先要存在才能成功，否则会报错。

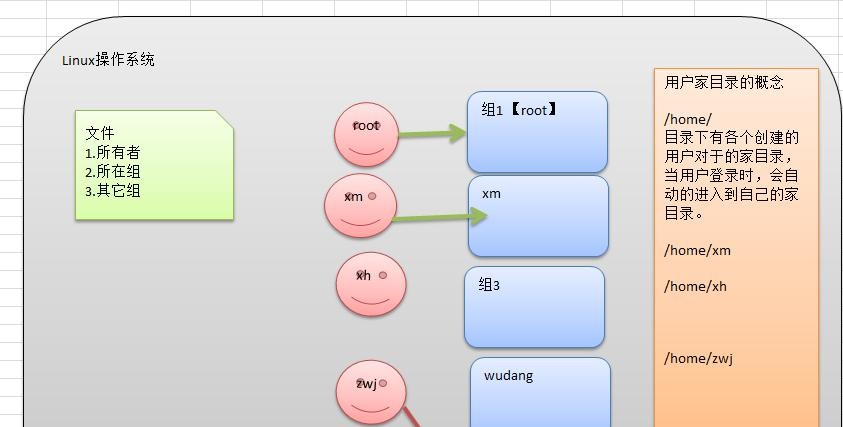


# 第 10 章 实操篇 组管理和权限管理

## Linux 组基本介绍

在 linux 中的每个用户必须属于一个组，不能独立于组外。在 linux 中每个文件有所有者、所在组、其它组的概念。

* + 1. 所有者
    2. 所在组
    3. 其它组
    4. 改变用户所在的组

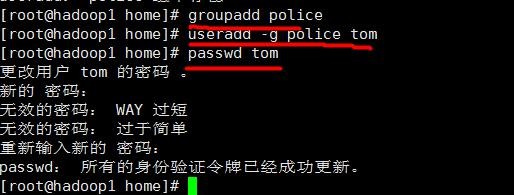


## 文件/目录 所有者

一般为文件的创建者,谁创建了该文件，就自然的成为该文件的所有者。

### 查看文件的所有者

* + - 1. 指令：ls -ahl
      2. 应用实例：创建一个组 police,再创建一个用户 tom,将 tom 放在 police 组 ,然后使用 tom 来创建一个文件 ok.txt，看看情况如何

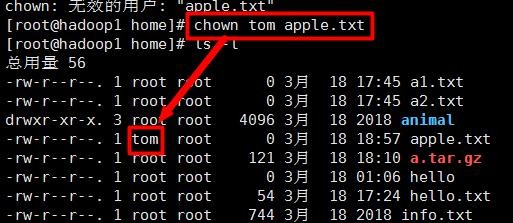




### 修改文件所有者

* 指令：chown 用户名 文件名
* 应用案例

要求：使用 root 创建一个文件 apple.txt ，然后将其所有者修改成 tom



## 组的创建

### 基本指令



groupadd 组 名

10.3.2 应用实例:

创建一个组, ,monster

创建一个用户 fox ，并放入到 monster 组中

## 文件/目录 所在组

当某个用户创建了一个文件后，默认这个文件的所在组就是该用户所在的组。

### 查看文件/目录所在组

* 基本指令

ls –ahl

* 应用实例

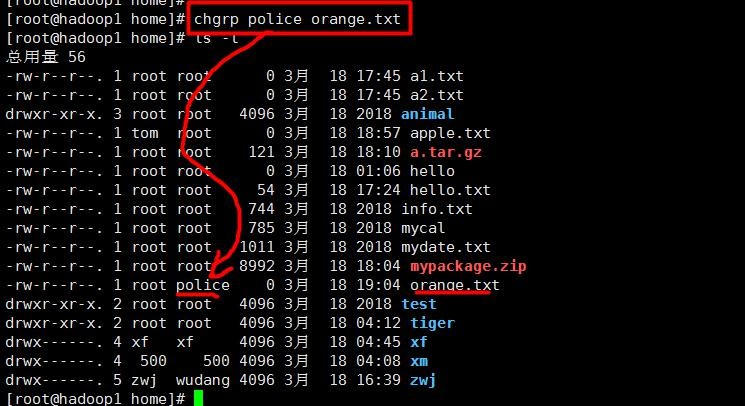
### 修改文件所在的组

* 基本指令

chgrp 组名 文件名

* 应用实例

使用 root 用户创建文件 orange.txt ,看看当前这个文件属于哪个组，然后将这个文件所在组，修改到 police 组 。



## 其它组

除文件的所有者和所在组的用户外，系统的其它用户都是文件的其它组.

## 改变用户所在组

在添加用户时，可以指定将该用户添加到哪个组中，同样的用 root 的管理权限可以改变某个用户所在的组。

### 改变用户所在组

* + - 1. usermod –g 组名 用户名
      2. usermod –d 目录名 用户名 改变该用户登陆的初始目录。

### 应用实例

创建一个土匪组（bandit）将 tom 这个用户从原来所在的 police 组，修改到 bandit(土匪) 组

## 权限的基本介绍



ls -l 中显示的内容如下：

**-rwxrw-r--** 1 root root 1213 Feb 2 09:39 abc

0-9 位说明

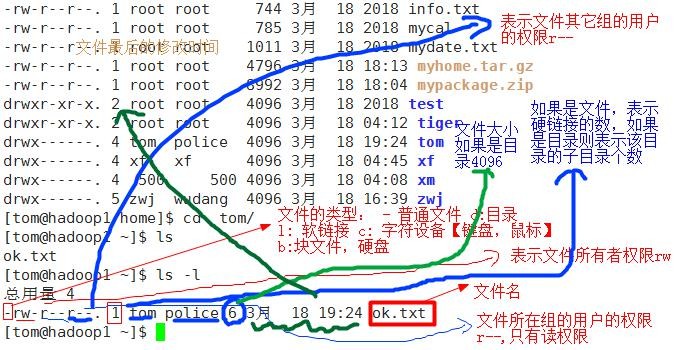
1)第 0 位确定文件类型(d, - , l , c , b)

2)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

3)第 4-6 位确定所属组（同用户组的）拥有该文件的权限，---Group

4)第 7-9 位确定其他用户拥有该文件的权限 ---Other

## rwx 权限详解



### rwx 作用到文件

* + - 1. [ r ]代表可读(read): 可以读取,查看
      2. [ 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.
      3. [ x ]代表可执行(execute):可以被执行

### rwx 作用到目录

* + - 1. [ r ]代表可读(read): 可以读取，ls 查看目录内容
      2. [ w ]代表可写(write): 可以修改,目录内创建+删除+重命名目录
      3. **[ x ]代表可执行(execute):可以进入该目录**

## 文件及目录权限实际案例

ls -l 中显示的内容如下：(记住)

-rwxrw-r-- 1 root root 1213 Feb 2 09:39 abc

10 个字符确定不同用户能对文件干什么

第一个字符代表文件类型： 文件 (-),目录(d),链接(l)

其余字符每 3 个一组(rwx) 读(r) 写(w) 执行(x) 第一组 rwx : 文件拥有者的权限是读、写和执行

第二组 rw- : 与文件拥有者同一组的用户的权限是读、写但不能执行第三组 r-- : 不与文件拥有者同组的其他用户的权限是读不能写和执行可用数字表示为: r=4,w=2,x=1 因此 rwx=4+2+1=7

## 修改权限-chmod

### 基本说明：

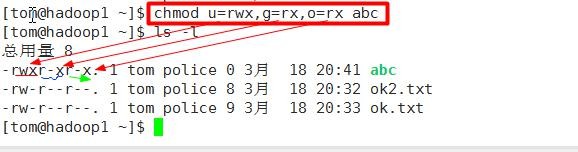
通过 chmod 指令，可以修改文件或者目录的权限

### 第一种方式：+ 、-、= 变更权限

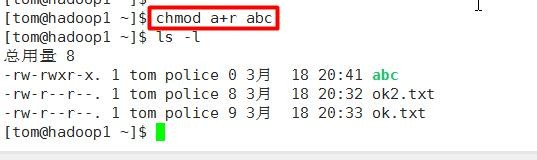
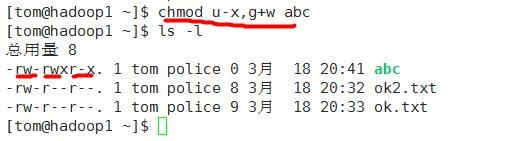
u:所有者 g:所有组 o:其他人 a:所有人(u、g、o 的总和)

* + - 1. chmod u=rwx,g=rx,o=x 文件目录名
      2. chmod o+w 文件目录名
      3. chmod a-x 文件目录名
* 案例演示

1. 给 abc 文件 的所有者读写执行的权限，给所在组读执行权限，给其它组读执行权限。



1. 给 abc 文件的所有者除去执行的权限，增加组写的权限



3) 给 abc 文件的所有用户添加读的权限

### 第二种方式：通过数字变更权限

规则：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

* 案例演示

要求：将 /home/abc.txt 文件的权限修改成 rwxr-xr-x, 使用给数字的方式实现：

rwx = 4+2+1 = 7

r-x = 4+1=5

r-x = 4+1 =5

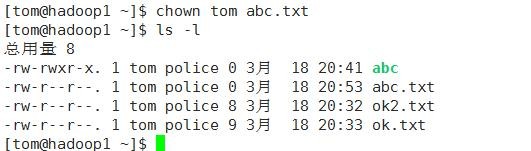
指令：chmod 755 /home/abc.txt

## 修改文件所有者-chown

### 基本介绍

chown newowner file 改变文件的所有者

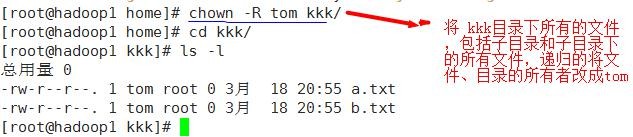
chown newowner:newgroup file 改变用户的所有者和所有组



-R 如果是目录 则使其下所有子文件或目录递归生效

### 案例演示：

* + - 1. 请将 /home/abc .txt 文件的所有者修改成 tom
      2. 请将 /home/kkk 目录下所有的文件和目录的所有者都修改成 tom

首选我们应该使用 root 操作。

## 修改文件所在组-chgrp

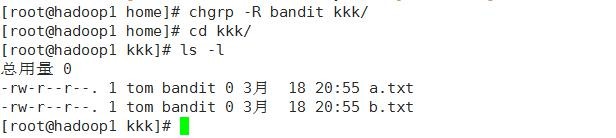
### 基本介绍

chgrp newgroup file 改变文件的所有组

### 案例演示：

* + - 1. 请将 /home/abc .txt 文件的所在组修改成 bandit (土匪) chgrp bandit /home/abc.txt
      2. 请将 /home/kkk 目录下所有的文件和目录的所在组都修改成 bandit(土匪) chgrp -R bandit /home/kkk

## 最佳实践-警察和土匪游戏



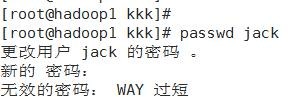
police ， bandit jack, jerry: 警 察xh, xq: 土 匪

1. 创建组

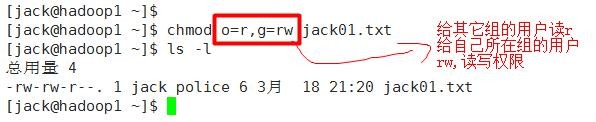
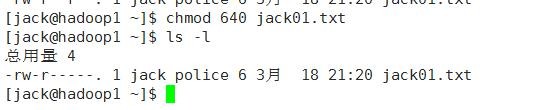
bash> groupadd police bash> groupadd bandit

1. 创建用户





1. jack 创建一个文件，自己可以读写，本组人可以读，其它组没人任何权限



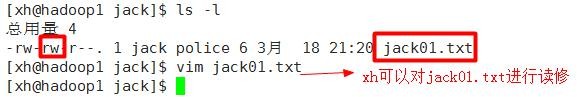
(4) jack 修改该文件，让其它组人可以读, 本组人可以读写

(5) xh 投靠 警察，看看是否可以读写. 先用 root 修改 xh 的组 ：



使用 jack 给他的家目录 /home/jack 的所在组一个 rx 的权限



xh 需要重新注销在到 jack 目录就可以操作 jack 的文件

## 课后练习题 2

1 用 root 登录，建立用户 mycentos,自己设定密码

2.用 mycentos 登录，在主目录下建立目录 test/t11/t1

3.在 t1 中建立一个文本文件 aa,用 vi 编辑其内容为 ls –al

4.改变 aa 的权限为可执行文件[可以将当前日期追加到一个文件],运行该文件./aa

5.删除新建立的目录 test/t11/t1

6.删除用户 mycentos 及其主目录中的内容

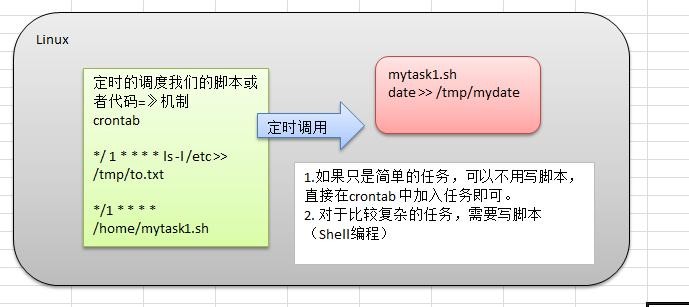
7.将 linux 设置成进入到图形界面的

8. 重新启动 linux 或关机



# 第 11 章 实操篇 crond 任务调度

## 原理示意图



crontab 进行 定时任务的设置，。

## 概述

任务调度：是指系统在某个时间执行的特定的命令或程序。

任务调度分类：1.系统工作：有些重要的工作必须周而复始地执行。如病毒扫描等

2.个别用户工作：个别用户可能希望执行某些程序，比如对 mysql 数据库的备份。

## 基本语法

crontab [选项]

### 常用选项

## 快速入门

### 任务的要求

设置任务调度文件：/etc/crontab

设置个人任务调度。执行 crontab –e 命令。接着输入任务到调度文件

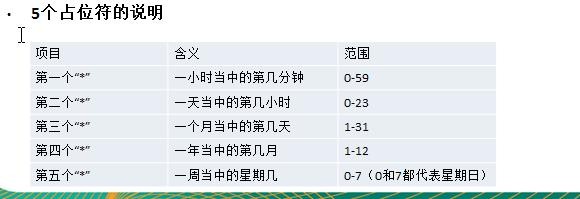
如：\*/1 \* \* \* \* ls –l /etc/ > /tmp/to.txt

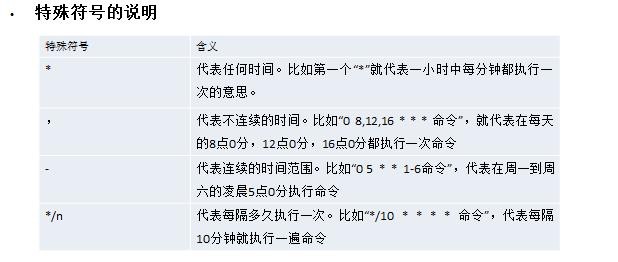
意思说每小时的每分钟执行 ls –l /etc/ > /tmp/to.txt 命令

### 步骤如下

* + - 1. crontab -e
      2. \*/ 1 \* \* \* \* ls -l /etc >> /tmp/to.txt
      3. 当保存退出后就程序。
      4. 在每一分钟都会自动的调用 ls -l /etc >> /tmp/to.txt

### 参数细节说明





## 任务调度的几个应用实例

* + 1. 案例 1：每隔 1 分钟，就将当前的日期信息，追加到 /tmp/mydate 文件

中

1. 先编写一个文件 /home/mytask1.sh date >> /tmp/mydate
2. 给 mytask1.sh 一个可以执行权限

chmod 744 /home/mytask1.sh

1. crontab -e
2. \*/1 \* \* \* \* /home/mytask1.sh
3. 成功

### 案例 2：每隔 1 分钟， 将当前日期和日历都追加到 /home/mycal 文件中

* + - 1. 先编写一个文件 /home/mytask2.sh

date >> /tmp/mycal cal >> /tmp/mycal

* + - 1. 给 mytask1.sh 一个可以执行权限

chmod 744 /home/mytask2.sh

* + - 1. crontab -e
      2. \*/1 \* \* \* \* /home/mytask2.sh
      3. 成功

### 案例 3: 每天凌晨 2:00 将 mysql 数据库 testdb ，备份到文件中

mydb.bak。

* + - 1. 先编写一个文件 /home/mytask3.sh

/usr/local/mysql/bin/mysqldump -u root -proot testdb > /tmp/mydb.bak

* + - 1. 给 mytask3.sh 一个可以执行权限

chmod 744 /home/mytask3.sh

* + - 1. crontab -e
      2. 0 2 \* \* \* /home/mytask3.sh
      3. 成功

## crond 相关指令:

1. service crond restart [重启任务调度]

# 第 12 章 实操篇 Linux 磁盘分区、挂载

## 分区基础知识

### 分区的方式：

* + - 1. mbr 分区:

1.最多支持四个主分区

2.系统只能安装在主分区

3.扩展分区要占一个主分区

4.MBR 最大只支持 2TB，但拥有最好的兼容性



2) gtp 分区:

1.支持无限多个主分区（但操作系统可能限制，比如 windows 下最多 128 个分区）

2.最大支持 18EB 的大容量（1EB=1024 PB，1PB=1024 TB ）

3.windows7 64 位以后支持 gtp

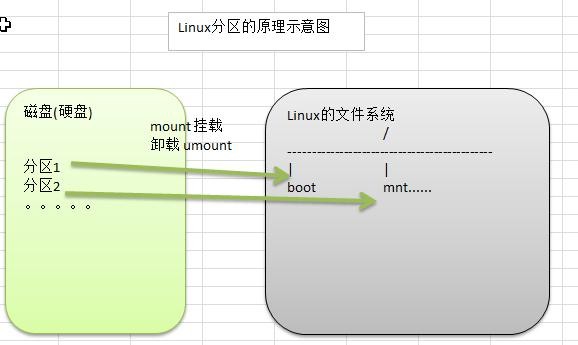
### windows 下的磁盘分区



## Linux 分区

### 原理介绍

* + - 1. Linux 来说无论有几个分区，分给哪一目录使用，它归根结底就只有一个根目录，一个独立且唯一的文件结构 , Linux 中**每个分区都是用来组成整个文件系统的一部分**。
      2. Linux 采用了一种叫“载入”的处理方法，它的整个文件系统中包含了一整套的文件和目录， 且将一个分区和一个目录联系起来。这时要载入的一个分区将使它的存储空间在一个目录下获得。

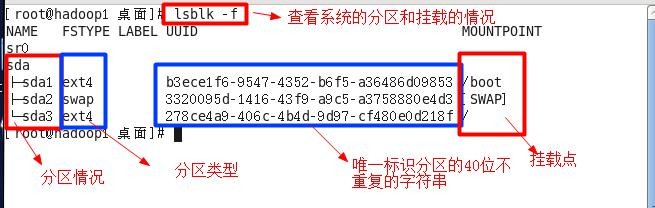
3)示意图

### 硬盘说明

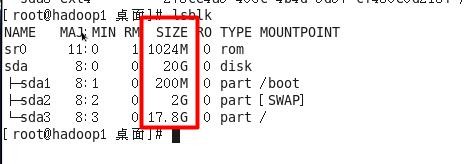
* + - 1. Linux 硬盘分 IDE 硬盘和 SCSI 硬盘，目前基本上是 SCSI 硬盘

2)对于 IDE 硬盘，驱动器标识符为“hdx~”,其中“hd”表明分区所在设备的类型，这里是指 IDE 硬盘了。“x”为盘号（a 为基本盘，b 为基本从属盘，c 为辅助主盘，d 为辅助从属盘）,“~”代表分区，前四个分区用数字 1 到 4 表示，它们是主分区或扩展分区，从 5 开始就是逻辑分区。例，hda3 表示为第一个 IDE 硬盘上的第三个主分区或扩展分区,hdb2 表示为第二个 IDE 硬盘上的第二个主分区或扩展分区。

3)对于 SCSI 硬盘则标识为“sdx~”，SCSI 硬盘是用“sd”来表示分区所在设备的类型的，其余则和 IDE 硬盘的表示方法一样。

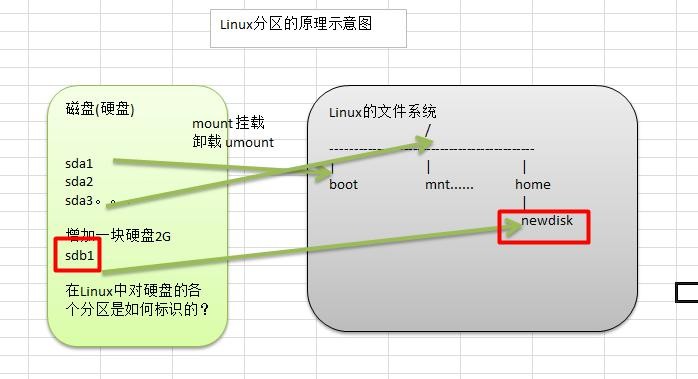


12.2.3 使用 lsblk 指令查看当前系统的分区情况



## 挂载的经典案例

需求是给我们的 Linux 系统增加一个新的硬盘，并且挂载到/home/newdisk



### 如何增加一块硬盘

1)虚拟机添加硬盘

2)分区 fdisk /dev/sdb

3)格式化 mkfs -t ext4 /dev/sdb1

4)挂载 先创建一个 /home/newdisk , 挂 载 mount /dev/sdb1 /home/newdisk

5)设置可以自动挂载(永久挂载，当你重启系统，仍然可以挂载到 /home/newdisk) 。

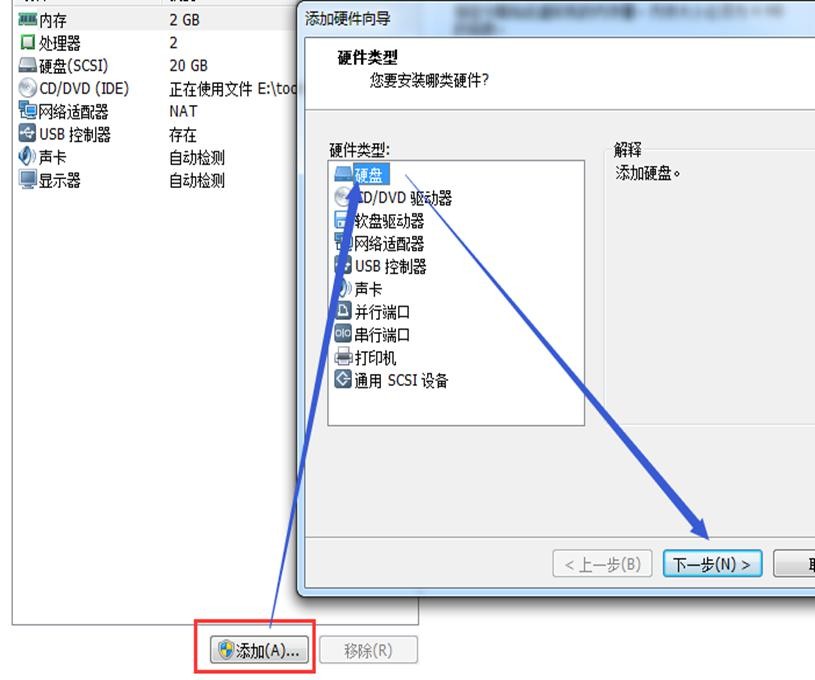
vim /etc/fstab

/dev/sdb1 /home/newdisk ext4 defaults 0 0

## 具体的操作步骤整理

### 虚拟机增加硬盘步骤 1

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（才能识别）！



### 虚拟机增加硬盘步骤 2

分区命令 fdisk /dev/sdb

开始对/sdb 分区

* m 显示命令列表
* p 显示磁盘分区 同 fdisk –l
* n 新增分区
* d 删除分区
* w 写入并退出

说明： 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。最后输入 w 写入分区并退出，若不保存退出输入 q。



### 虚拟机增加硬盘步骤 3

格式化磁盘

分区命令:mkfs -t ext4 /dev/sdb1

其中 ext4 是分区类型

### 虚拟机增加硬盘步骤 4

挂载: 将一个分区与一个目录联系起来，

* mount 设备名称 挂载目录
* 例如： mount /dev/sdb1 /newdisk
* umount 设备名称 或者 挂载目录
* 例如： umount /dev/sdb1 或 者 umount /newdisk

### 虚拟机增加硬盘步骤 5

永久挂载: 通过修改/etc/fstab 实现挂载添完成后 执行 mount –a 即刻生效

## 磁盘情况查询

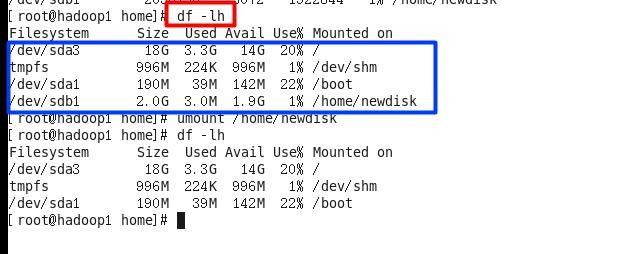
### 查询系统整体磁盘使用情况

基本语法

df -h

应用实例

查询系统整体磁盘使用情况



### 查询指定目录的磁盘占用情况

* 基本语法

du /路径

查询指定目录的磁盘占用情况，默认为当前目录

-s 指定目录占用大小汇总

-h 带计量单位

-a 含文件

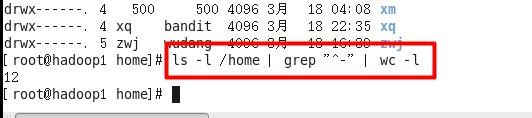
--max-depth=1 子目录深度

-c 列出明细的同时，增加汇总值

* 应用实例

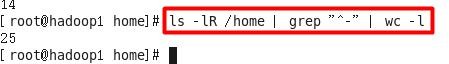
查询 /opt 目录的磁盘占用情况，深度为 1

## 磁盘情况-工作实用指令



1. 统计/home 文件夹下文件的个数
2. 统计/home 文件夹下目录的个数



1. 统计/home 文件夹下文件的个数，包括子文件夹里的
2. 统计文件夹下目录的个数，包括子文件夹里的



1. 以树状显示目录结构



# 第 13 章 实操篇 网络配置

## Linux 网络配置原理图(含虚拟机)

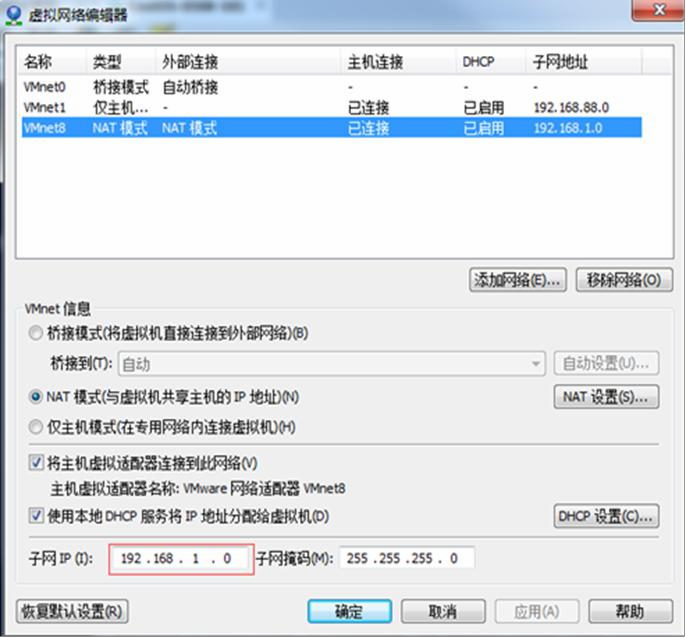
目前我们的网络配置采用的是 NAT。

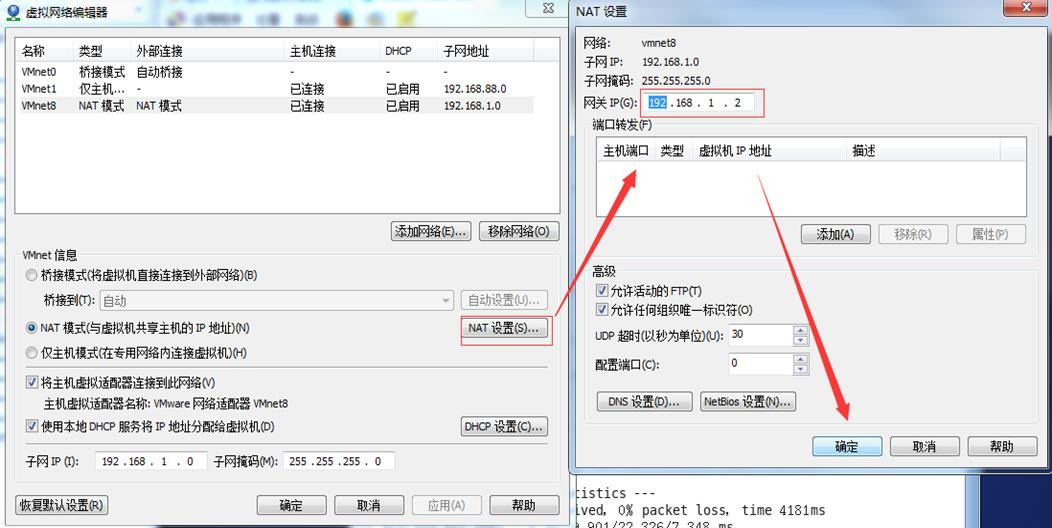
## 查看网络 IP 和网关



### 查看虚拟网络编辑器

### 修改 ip 地址(修改虚拟网络的 ip)



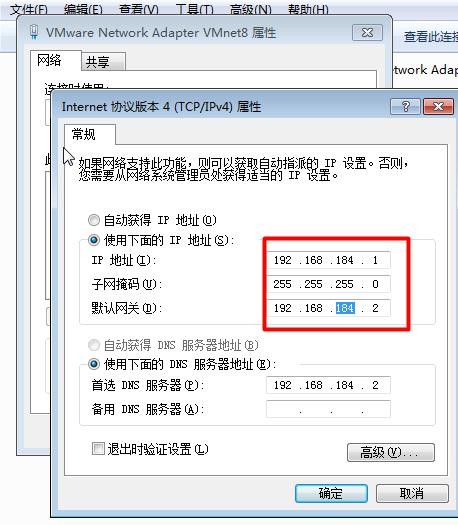


13.2.3 查看网关

### 查看 windows 环境的中 VMnet8 网络配置 (ipconfig 指令)

* + - 1. 使用 ipconfig 查看
      2. 界面查看

## ping 测试主机之间网络连通



### 基本语法

ping 目的主机 （功能描述：测试当前服务器是否可以连接目的主机）

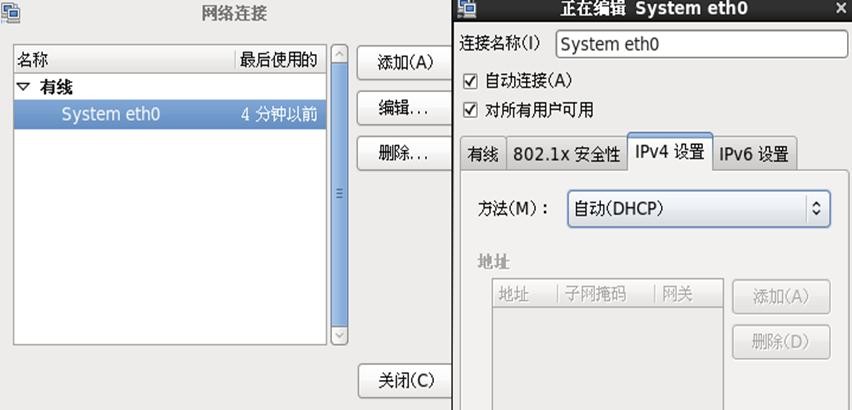
### 应用实例

测试当前服务器是否可以连接百度

[root@hadoop100 桌面]# ping [www.baidu.com](http://www.baidu.com/)

## linux 网络环境配置

### 第一种方法(自动获取)



缺点: linux 启动后会自动获取 IP,缺点是每次自动获取的 ip 地址可能不一样。这个不适用于做服务器，因为我们的服务器的 ip 需要时固定的。

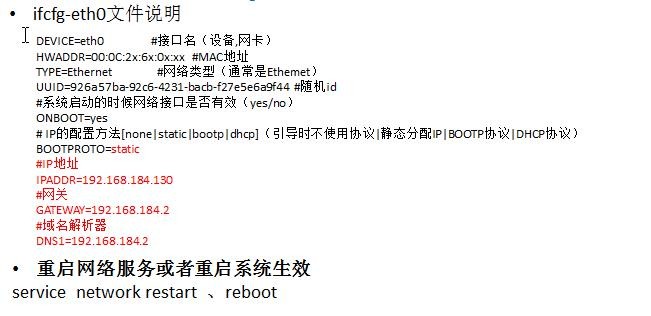
### 第二种方法(指定固定的 ip)

说明

直 接 修 改 配 置 文 件 来 指 定 IP, 并 可 以 连 接 到 外 网 ( 程 序 员 推 荐 ) ， 编 辑 vi

/etc/sysconfig/network-scripts/ifcfg-eth0

要求：将 ip 地址配置的静态的，ip 地址为 192.168.184.130



修改后，一定要 重启服务

* + - 1. service network restart
      2. reboot 重启系统

# 第 14 章 实操篇 进程管理

## 进程的基本介绍

1)在 LINUX 中，每个执行的程序（代码）都称为一个进程。每一个进程都分配一个 ID 号。

2)每一个进程，都会对应一个父进程，而这个父进程可以复制多个子进程。例如 www 服务器。

3)每个进程都可能以两种方式存在的。前台与后台，所谓前台进程就是用户目前的屏幕上可以进行操作的。后台进程则是实际在操作，但由于屏幕上无法看到的进程，通常使用后台方式执行。

4)一般系统的服务都是以后台进程的方式存在，而且都会常驻在系统中。直到关机才才结束。

## 显示系统执行的进程



### 说明：

查看进行使用的指令是 ps ,一般来说使用的参数是 ps -aux



### ps 指令详解

1)指令：ps –aux|grep xxx ，比如我看看有没有 sshd 服务



2)指令说明

* 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

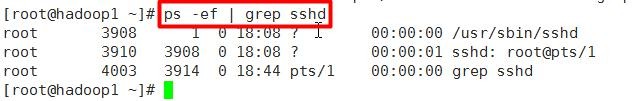
### 应用实例

要求：以全格式显示当前所有的进程，查看进程的父进程。



* ps -ef 是以全格式显示当前所有的进程
* -e 显示所有进程。-f 全格式。
* 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

思考题，如果我们希望查看 sshd 进程的父进程号是多少，应该怎样查询 ？



## 终止进程 kill 和 killall

### 介绍:

若是某个进程执行一半需要停止时，或是已消了很大的系统资源时，此时可以考虑停止该进程。使用 kill 命令来完成此项任务。

### 基本语法：

kill [选项] 进程号（功能描述：通过进程号杀死进程）

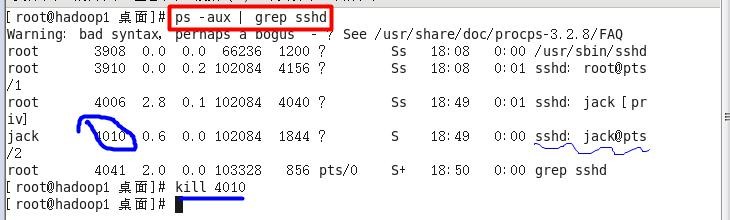
killall 进程名称（功能描述：通过进程名称杀死进程，也支持通配符，这在系统因负载过大而变得很慢时很有用）

### 常用选项：

-9 :表示强迫进程立即停止

### 最佳实践：

案例 1：踢掉某个非法登录用户



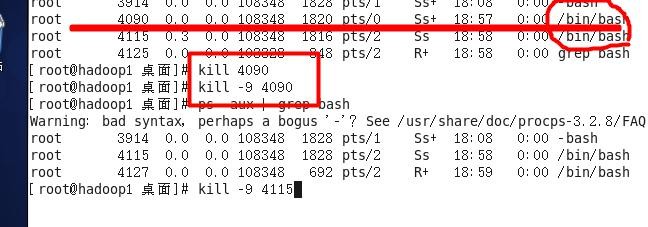
案例 2: 终止远程登录服务 sshd, 在适当时候再次重启 sshd 服务



案例 3: 终止多个 gedit 编辑器 【killall , 通过进程名称来终止进程】



案例 4：强制杀掉一个终端



## 查看进程树 pstree

### 基本语法：



pstree [选项] ,可以更加直观的来看进程信息

* + 1. 常用选项：

-p :显示进程的 PID

-u :显示进程的所属用户

* + 1. 应用实例：

案例 1：请你树状的形式显示进程的 pid

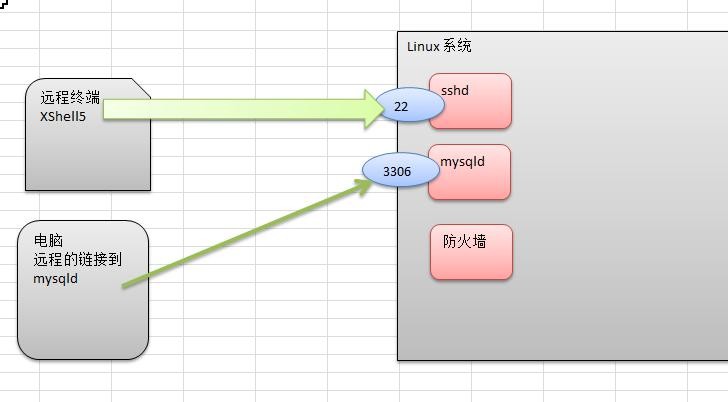


案例 2：请你树状的形式进程的用户 id pstree -u 即可。

## 服务(Service)管理

### 介绍:

服务(service) 本质就是进程，但是是运行在后台的，通常都会监听某个端口，等待其它程序的请求，比如(mysql , sshd 防火墙等)，因此我们又称为守护进程，是 Linux 中非常重要的知识点。【原理图】



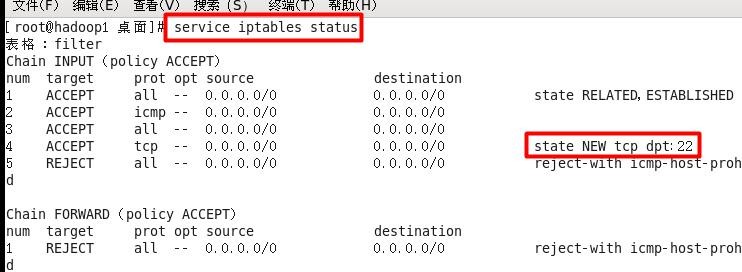
14.5.2 service 管理指令：

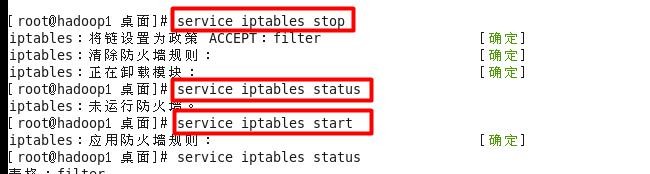
service/systemctl(centos7) 服务名 [start | stop | restart | reload | status]

**在 CentOS7.0 后 不再使用 service ,而是 systemctl**

### 使用案例：

* + - 1. 查看当前防火墙的状况，关闭防火墙和重启防火墙。





* + 1. 细节讨论：
       1. 关闭或者启用防火墙后，立即生效。[telnet 测试 某个端口即可]



2)这种方式只是临时生效，当重启系统后，还是回归以前对服务的设置。

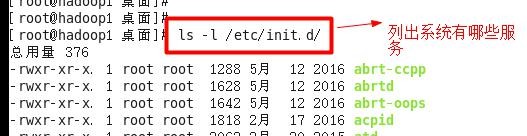
如果希望设置某个服务自启动或关闭永久生效，要使用 chkconfig 指令，马上讲

### 查看服务名:

方式 1：使用 setup -> 系统服务 就可以看到。



方式 2: /etc/init.d/服务名称



14.5.6 服务的运行级别(runlevel):

查看或者修改默认级别： vi /etc/inittab

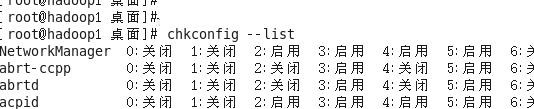
Linux 系统有 7 种运行级别(runlevel)：常用的是级别 3 和 5

* + - * 运行级别 0：系统停机状态，系统默认运行级别不能设为 0，否则不能正常启动
      * 运行级别 1：单用户工作状态，root 权限，用于系统维护，禁止远程登陆
      * 运行级别 2：多用户状态(没有 NFS)，不支持网络
      * 运行级别 3：完全的多用户状态(有 NFS)，登陆后进入控制台命令行模式
      * 运行级别 4：系统未使用，保留
      * 运行级别 5：X11 控制台，登陆后进入图形 GUI 模式
      * 运行级别 6：系统正常关闭并重启，默认运行级别不能设为 6，否则不能正常启动

### 开机的流程说明

* + 1. chkconfig 指令介绍

### 通过 chkconfig 命令可以给每个服务的各个运行级别设置自启动/关闭



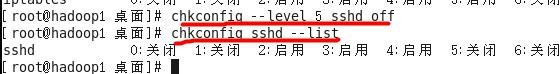
基本语法

1) 查看服务 chkconfig

--list|grep xxx

### chkconfig 服务名 --list

### chkconfig --level 5 服务名 on/off

请将 sshd 服务在运行级别为 5 的情况下，不要自启动。

### 应用实例：

* + - 1. 案例 1： 请显示当前系统所有服务的各个运行级别的运行状态

bash> chkconfig --list

* + - 1. 案例 2 ：请查看 sshd 服务的运行状态

bash> service sshd status



1. 案例 3： 将 sshd 服务在运行级别 5 下设置为不自动启动，看看有什么效果？

bash> chkconfig --level 5 sshd off

1. 案例 4： 当运行级别为 5 时，关闭防火墙。

bash> chkconfig --level 5 iptables off

1. 案例 5： 在所有运行级别下，关闭防火墙

bash> chkconfig iptables off

6) 案例 6： 在所有运行级别下，开启防火墙

bash> chkconfig iptables on

### • 使用细节

* + - 1. chkconfig 重新设置服务后自启动或关闭，需要重启机器 reboot 才能生效.

## 动态监控进程

### 介绍：

top 与 ps 命令很相似。它们都用来显示正在执行的进程。Top 与 ps 最大的不同之处，在于 top 在执行一段时间可以更新正在运行的的进程。

### 基本语法：

top [选项]

### 选项说明：



### 应用实例：

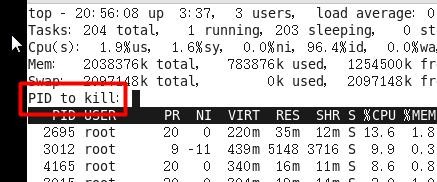
案例 1.监视特定用户

top：输入此命令，按回车键，查看执行的进程。

u：然后输入“u”回车，再输入用户名，即可



案例 2：终止指定的进程。



top：输入此命令，按回车键，查看执行的进程。

k：然后输入“k”回车，再输入要结束的进程 ID 号

案例 3:指定系统状态更新的时间(每隔 10 秒自动更新， 默认是 3 秒)：

bash> top -d 10

### 查看系统网络情况 netstat(重要)

* + - * 基本语法

netstat [选项]

netstat -anp

* + - * 选项说明

-an 按一定顺序排列输出

-p 显示哪个进程在调用

* + - * 应用案例



查看系统所有的网络服务

请查看服务名为 sshd 的服务的信息。

# 第 15 章 实操篇 RPM 和 YUM

## 15.1 rpm 包的管理

### 15.1.1 介绍：

一种用于互联网下载包的打包及安装工具，它包含在某些 Linux 分发版中。它生成具有.RPM 扩展名的文件。RPM 是 RedHat Package Manager（RedHat 软件包管理工具）的缩写，类似 windows 的 setup.exe，这一文件格式名称虽然打上了 RedHat 的标志，但理念是通用的。

Linux 的分发版本都有采用（suse,redhat, centos 等等），可以算是公认的行业标准了。



15.1.2 rpm 包的简单查询指令：

查询已安装的 rpm 列表 rpm –qa|grep xx

请查询看一下，当前的 Linux 有没有安装 firefox .

### rpm 包名基本格式：

一个 rpm 包名：firefox-45.0.1-1.el6.centos.x86\_64.rpm 名称:firefox

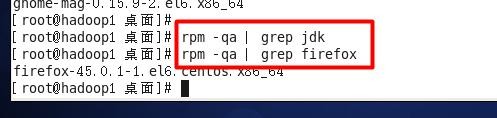
版本号：45.0.1-1

适用操作系统: el6.centos.x86\_64

表示 centos6.x 的 64 位系统

如果是 i686、i386 表示 32 位系统，noarch 表示通用。。

* + 1. rpm 包的其它查询指令： rpm -qa :查询所安装的所有 rpm 软件包rpm -qa | more [分页显示]



rpm -qa | grep X [rpm -qa | grep firefox ]

rpm -q 软件包名 :查询软件包是否安装rpm -q firefox

rpm -qi 软件包名 ：查询软件包信息



rpm -qi file



rpm -ql 软件包名 :查询软件包中的文件

rpm -ql firefox

rpm -qf 文件全路径名 查询文件所属的软件包

rpm -qf /etc/passwd rpm -qf /root/install.log



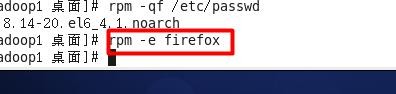
### 卸载 rpm 包：

* + - * 基本语法

rpm -e RPM 包的名称

* + - * 应用案例

1. 删除 firefox 软件包
   * + - 细节问题



1. 如果其它软件包依赖于您要卸载的软件包，卸载时则会产生错误信息。如： $ rpm -e foo

removing these packages would break dependencies:foo is needed by bar-1.0-1

1. 如果我们就是要删除 foo 这个 rpm 包，可以增加参数 --nodeps ,就可以强制删除，但是一般不推荐这样做，因为依赖于该软件包的程序可能无法运行

如：$ rpm -e --nodeps foo

带上 --nodeps 就是强制删除。

### 安装 rpm 包：

* + - * 基本语法

rpm -ivh RPM 包全路径名称

* + - * 参数说明

i=install 安 装

v=verbose 提 示

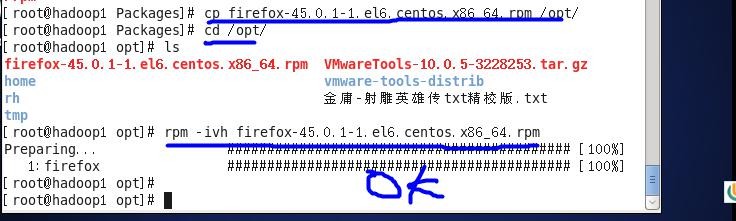
h=hash 进度条

* + - * 应用实例

1) 演示安装 firefox 浏览器

步骤先找到 firefox 的安装 rpm 包,你需要挂载上我们安装 centos 的 iso 文件，然后到/media/下去找 rpm 找 。

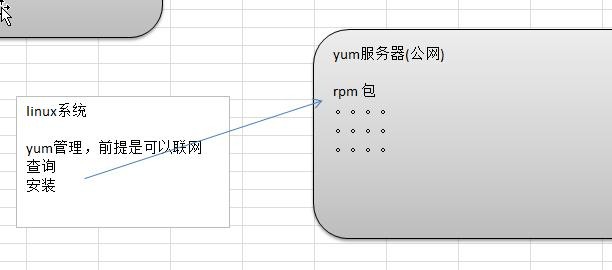
cp firefox-45.0.1-1.el6.centos.x86\_64.rpm /opt/



## yum

### 介绍：

Yum 是一个 [Shell](https://baike.baidu.com/item/Shell) 前端软件包管理器。基于 [RPM](https://baike.baidu.com/item/RPM) 包管理，能够从指定的服务器自动下载 RPM 包并且安装，可以自动处理依赖性关系，并且一次安装所有依赖的软件包。使用 yum 的前提是可以联网。



### yum 的基本指令



* 查询 yum 服务器是否有需要安装的软件

yum list|grep xx 软件列表

* 安装指定的 yum 包

yum install xxx 下载安装

### yum 应用实例：

案例：请使用 yum 的方式来安装 firefox

* + - 1. 先查看一下 firefox rpm 在 yum 服务器有没有
      2. 安装

yum install firefox

# 第 18 章 Python 定制篇 开发平台 Ubuntu

## Ubuntu 的介绍

Ubuntu（友帮拓、优般图、乌班图）[是一个以桌面](https://baike.baidu.com/item/%E6%A1%8C%E9%9D%A2)应用为主的开源 GNU/Linux 操作系统，Ubuntu 是基于 GNU/[Linux](https://baike.baidu.com/item/Linux)，支持 x86、amd64（即 x64）和 [ppc](https://baike.baidu.com/item/ppc/150) 架构，由全球化的专业开发团队（Canonical Ltd） 打造的。

专业的 Python 开发者一般会选择 Ubuntu 这款 Linux 系统作为生产平台. 温馨提示：

Ubuntu 和 Centos 都是基于 GNU/Linux 内核的，因此基本使用和 Centos 是几乎一样的，它们的各种指令可以通用，同学们在学习和使用 Ubuntu 的过程中，会发现各种操作指令在前面学习 CentOS 都使用过。只是界面和预安装的软件有所差别。



Ubuntu 下载地址：<http://cn.ubuntu.com/download/>

## Ubuntu 的安装

### 安装的步骤

尚硅谷Python安装Ubuntu.zip

### 设置 Ubuntu 支持中文

默认安装的 ubuntu 中只有英文语言，因此是不能显示汉字的。要正确显示汉字，需要安装中文语言包。

安装中文支持步骤：

1.单击左侧图标栏打开 System Settings（系统设置）菜单，点击打开 Language Support（语言支持）选项卡。

2.点击 Install / Remove Languages，在弹出的选项卡中下拉找到 Chinese(Simplified)，即中文简体， 在后面的选项框中打勾。然后点击 Apply Changes 提交，系统会自动联网下载中文语言包。（保证

ubuntu 是联网的）。

3.这时“汉语（中国）”在最后一位因为当前第一位是”English”，所以默认显示都是英文。我们如果希望默认显示用中文，则应该将“汉语（中国）”设置为第一位。设置方法是拖动，鼠标单击

“汉语（中国）”，当底色变化（表示选中了）后，按住鼠标左键不松手，向上拖动放置到第一位。

4.设置后不会即刻生效，需要下一次登录时才会生效。



## Ubuntu 的 roo 用户

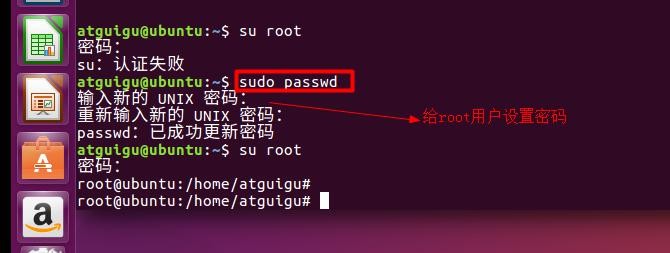
### 介绍

安装 ubuntu 成功后，都是普通用户权限，并没有最高 root 权限，如果需要使用 root 权限的时候， 通常都会在命令前面加上 sudo 。有的时候感觉很麻烦。

我们一般使用su 命令来直接切换到root 用户的，但是如果没有给root 设置初始密码，就会抛出 su : Authentication failure 这样的问题。所以，我们只要给 root 用户设置一个初始密码就好了。

### 给 root 用户设置密码并使用

* + - 1. 输入 sudo passwd 命令，输入一般用户密码并设定 root 用户密码。
      2. 设定 root 密码成功后，输入 su 命令，并输入刚才设定的 root 密码，就可以切换成 root 了。提示符$代表一般用户，提示符#代表 root 用户。

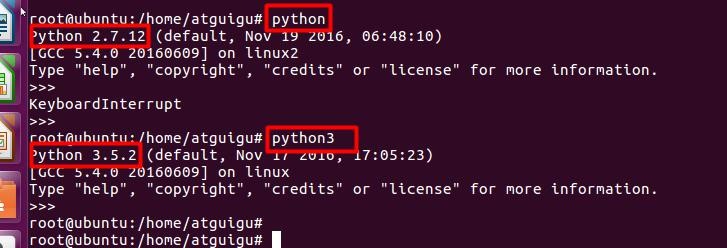


* + - 1. 输入 exit 命令，退出 root 并返回一般用户
      2. 以后就可以使用 root 用户了

## Ubuntu 下开发 Python

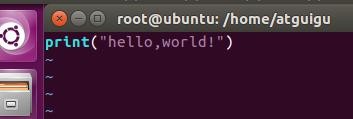
### 说明

安装好 Ubuntu 后，默认就已经安装好 Python 的开发环境[Python2.7 和 Python3.5]。



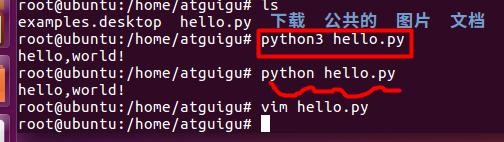
### 在 Ubuntu 下开发一个 Python 程序

* + - 1. vim hello.py [编写 hello.py]



提示：如果 Ubuntu 没有 vim 我们可以根据提示信息安装一个 vim apt install vim

* + - 1. python3 hello.py [运行 hello.py]



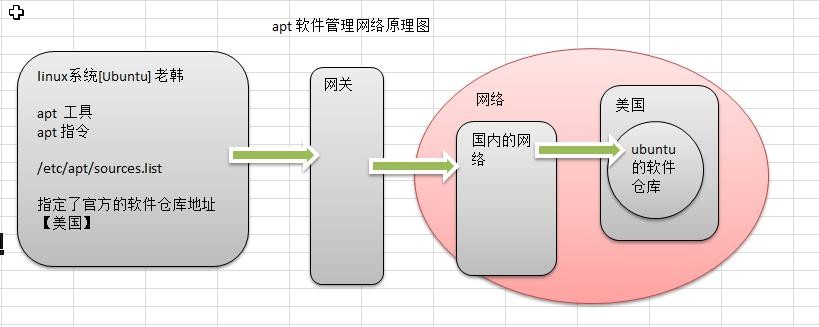
# 第 19 章 Python 定制篇 apt 软件管理和远程登录

## apt 介绍

apt 是 Advanced Packaging Tool 的简称，是一款安装包管理工具。在 Ubuntu 下，我们可以使用 apt

命令可用于软件包的安装、删除、清理等，类似于 Windows 中的软件管理工具。

unbuntu 软件管理的原理示意图：



## Ubuntu 软件操作的相关命令

sudo apt-get update 更新源

sudo apt-get install package 安装包

sudo apt-get remove package 删除包

sudo apt-cache search package 搜索软件包

sudo apt-cache show package 获取包的相关信息，如说明、大小、版本等

sudo apt-get install package --reinstall 重新安装包

sudo apt-get -f install 修复安装

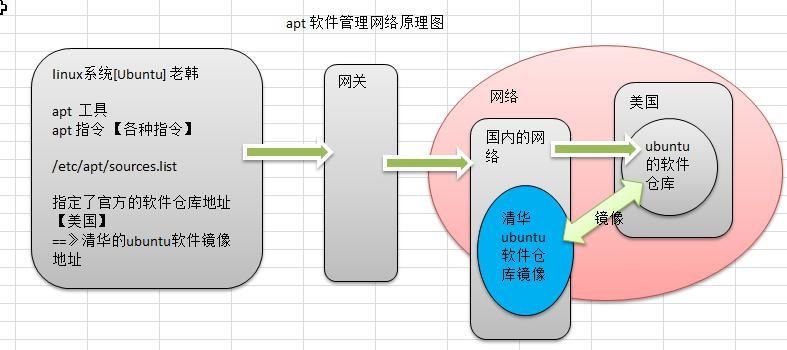
sudo apt-get remove package --purge 删除包，包括配置文件等

sudo apt-get build-dep package 安装相关的编译环境

sudo apt-get upgrade 更新已安装的包

sudo apt-get dist-upgrade 升级系统

sudo apt-cache depends package 了解使用该包依赖那些包sudo apt-cache rdepends package 查看该包被哪些包依赖sudo apt-get source package 下载该包的源代码



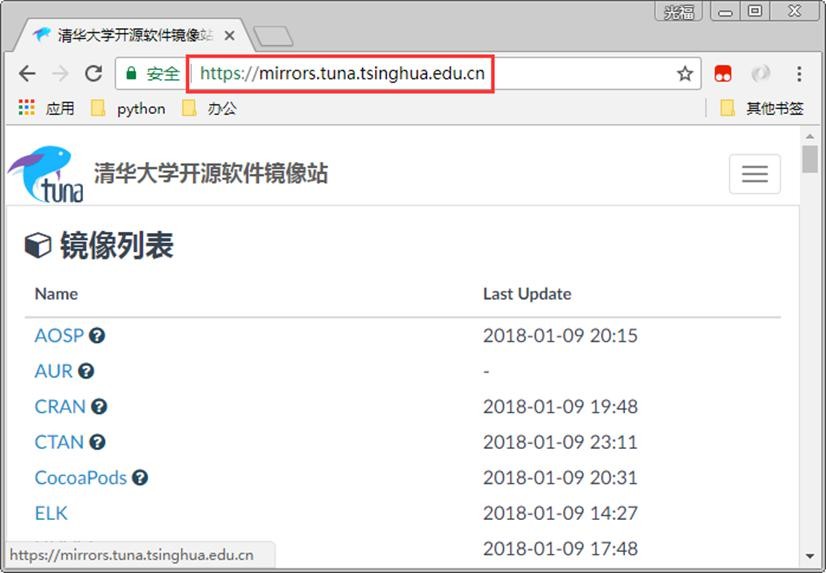
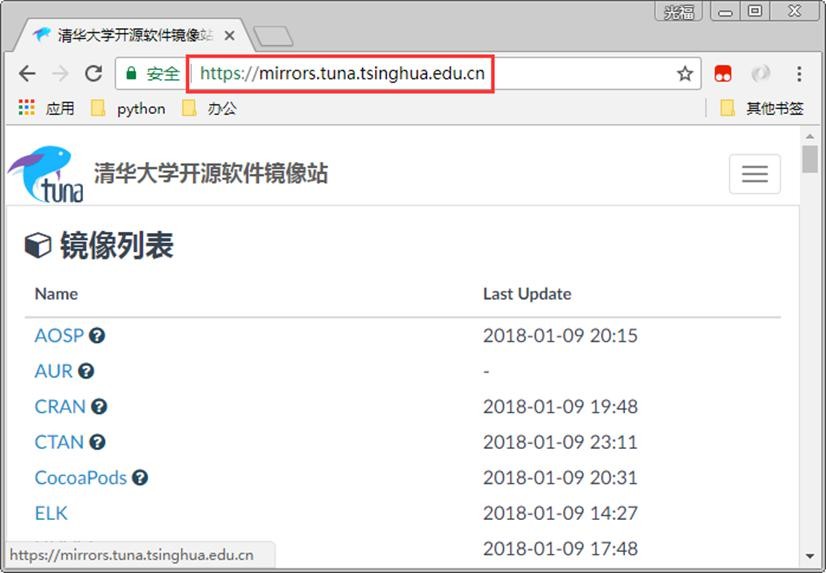
## 更新 Ubuntu 软件下载地址

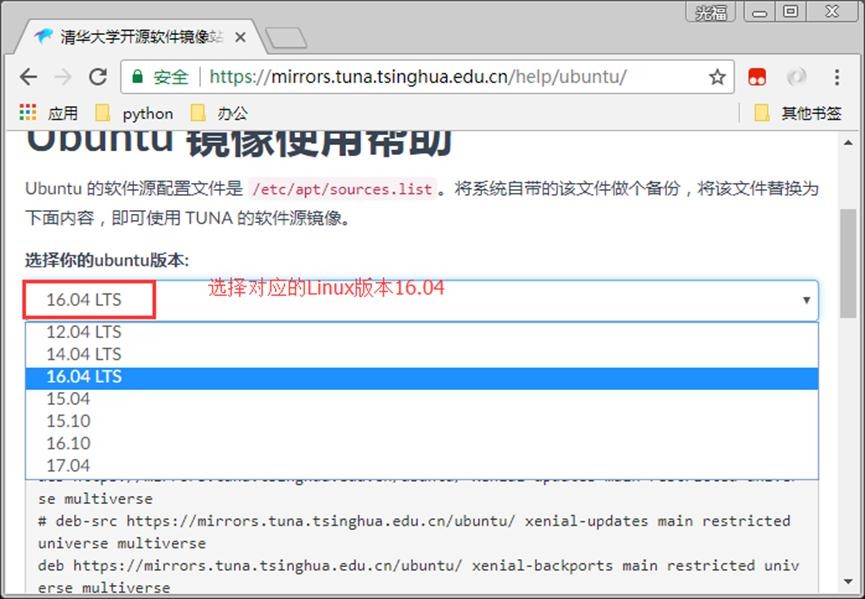
### 原理示意图

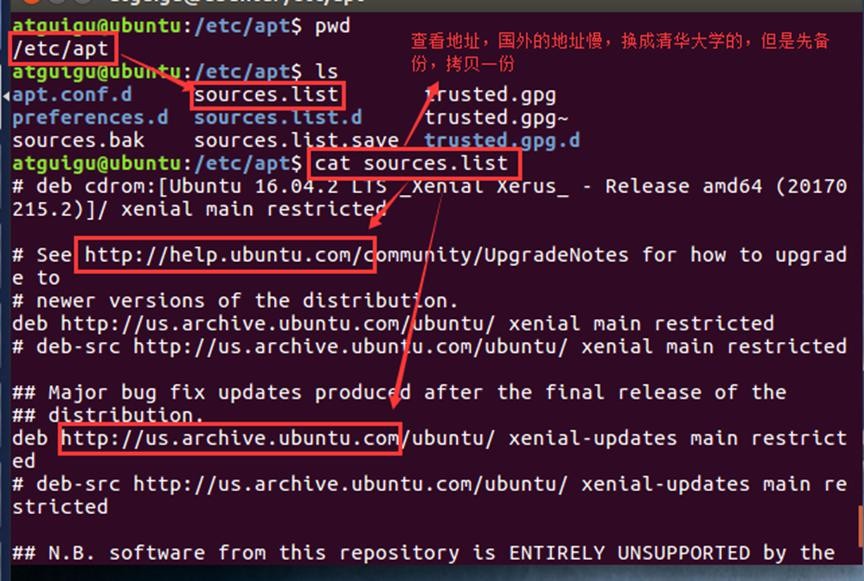
### 寻找国内镜像源

https://mirrors.tuna.tsinghua.edu.cn/

所谓的镜像源：可以理解为提供下载软件的地方，比如 Android 手机上可以下载软件的安卓市场；iOS 手机上可以下载软件的 AppStore

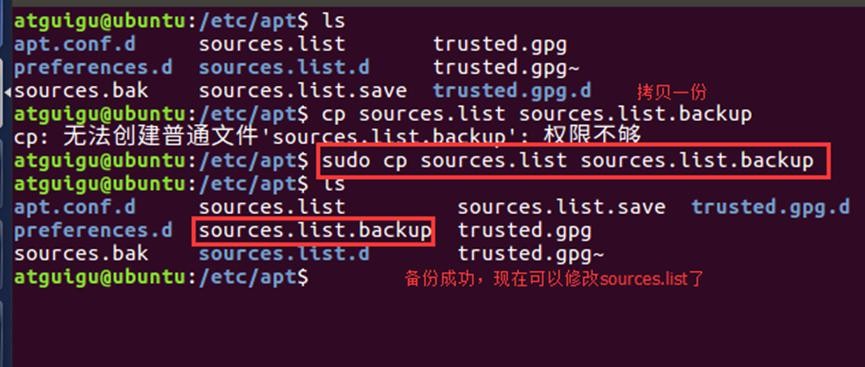




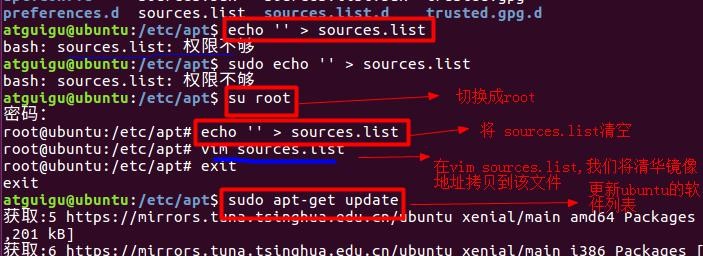


19.3.3 备份 Ubuntu 默认的源地址

sudo cp /etc/apt/sources.list /etc/apt/sources.list.backup



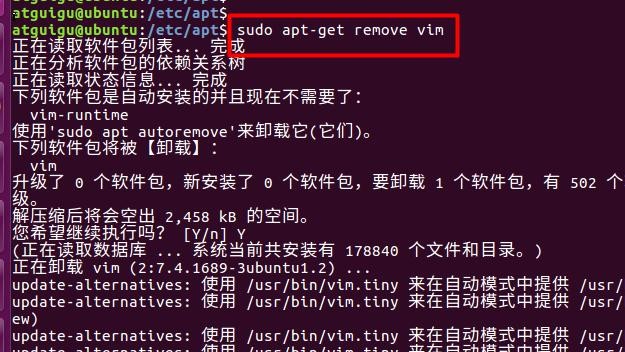
### 19.3.4 更新源服务器列表



## Ubuntu 软件安装，卸载的最佳实践

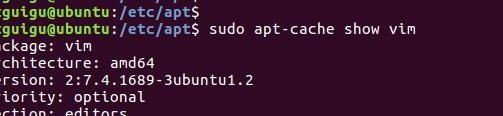
### 案例说明：使用 apt 完成安装和卸载 vim 软件，并查询 vim 软件的信息：

sudo apt-get remove vim



sudo apt-get install vim

sudo apt-cache show vim



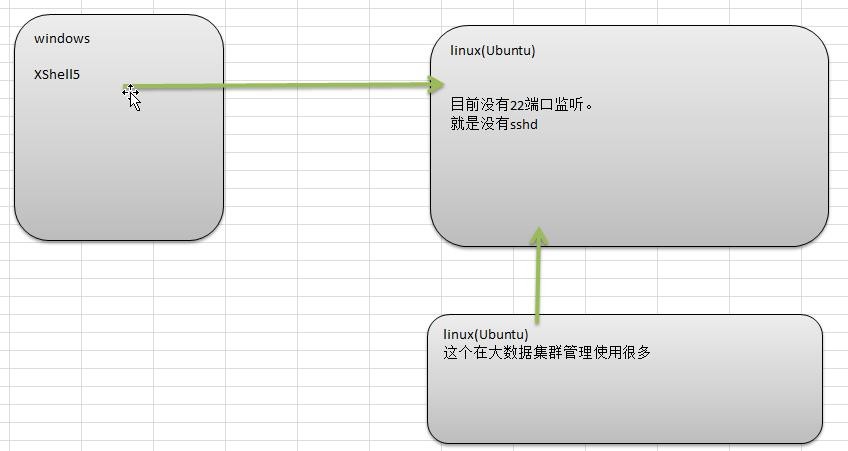
## 使用 ssh 远程登录 Ubuntu

### ssh 介绍

SSH 为 Secure Shell 的缩写，由 IETF 的网络工作小组（Network Working Group）所制定；SSH 为建立在应用层和传输层基础上的安全协议。

SSH 是目前较可靠，专为远程登录会话和其他网络服务提供安全性的协议。常用于远程登录，以及用户之间进行资料拷贝。几乎所有 UNIX 平台—包括 HP-UX、Linux、AIX、Solaris、Digital UNIX、Irix，以及其他平台，都可运行 SSH。

使用 SSH 服务，需要安装相应的服务器和客户端。客户端和服务器的关系：如果，A 机器想被 B



机器远程控制，那么，A 机器需要安装 SSH 服务器，B 机器需要安装 SSH 客户端。

和 CentOS 不一样，Ubuntu 默认没有安装 SSHD 服务，因此，我们不能进行远程登录。

### 原理示意图：

## 使用 ssh 远程登录 Ubuntu

### 安装 SSH 和启用

sudo apt-get install openssh-server

执行上面指令后，在当前这台 Linux 上就安装了 SSH 服务端和客户端。

service sshd restart

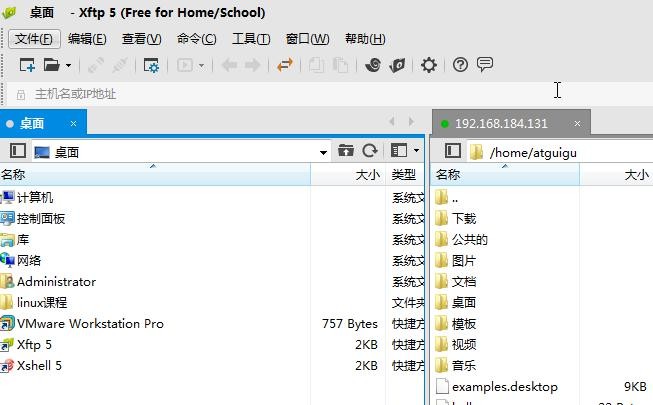
执行上面的指令，就启动了 sshd 服务。会监听端口 22



19.6.2 在 Windows 使用 XShell5/XFTP5 登录 Ubuntu

前面我们已经安装了 XShell5，直接使用即可。

注意：使用 atguigu 用户登录，需要的时候再 su - 切换成 root 用户



19.6.3 从 linux 系统客户机远程登陆 linux 系统服务机

首先，我们需要在 linux 的系统客户机也要安装 openssh-server

* + - * 基本语法：

ssh 用户名@IP

例如：ssh [atguigu@192.168.188.131](mailto:atguigu@192.168.188.131)

使用 ssh 访问，如访问出现错误。可查看是否有该文件 ～/.ssh/known\_ssh 尝试删除该文件解决。

* + - * 登出

登出命令：exit 或者 logout

