



Linux (大数据、JavaEE、
Python 开发通用版)

尚硅谷

Linux开山篇

内容介绍和学习方法

● 本套Linux课程内容

基础篇

linux入门

vm和Linux的安装

Linux目录结构

实操篇

远程登录
(XShell Xftp)

Vi和Vim编辑器

开机、重启和
用户登录注销

用户管理

实用指令

定时任务调度

磁盘分区、挂载

网络配置

进程管理

RPM和YUM



● 本套Linux课程内容

Linux之JavaEE定制篇

Linux搭建JavaEE环境

- 1) JDK安装
- 2) Eclipse 的安装
- 3) Tomcat的安装
- 4) MySQL的安装

Linux之大数据定制篇

Shell编程

- 1) 概述
- 2) Shell脚本执行方式
- 3) Shell变量
- 4) 运算符
- 5) 条件判断
- 6) 流程控制
- 7) 函数
- 8) 定时维护MySQL数据库

Linux之Python定制篇

Python专业开发平台-Ubuntu

Ubuntu下开发Python开发环境

APT软件管理和远程登录

- Linux学习方向

linux运维工程师

linux嵌入式工程师

linux下开发项目

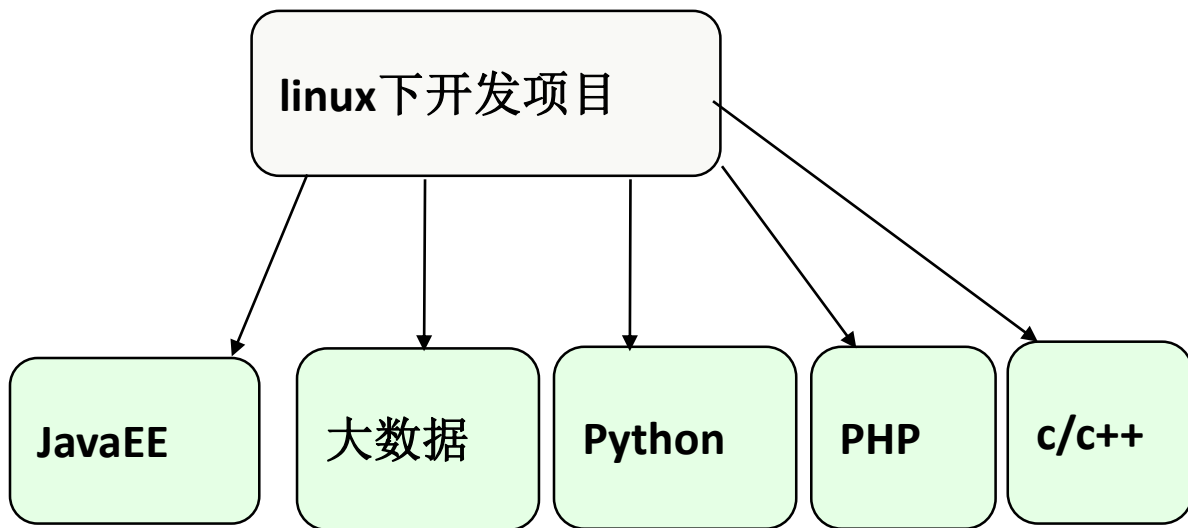
JavaEE

大数据

Python

PHP

c/c++



● Linux的应用领域

个人桌面领域的应用

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 [fɪ'dɔ: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。



● Linux的应用领域

服务器领域

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**。



● Linux高手进阶之路

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

- 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) 适当的囫圇吞枣
- 7) Linux不是编程，重点是实际操作，各种常用指令要玩的溜

Linux基础篇

Linux入门

● Linux的介绍

概述:

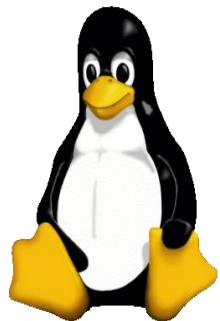
1. linux怎么读?
2. **linux**是一个开源、免费的**操作系统**，其稳定性、安全性、处理多并发已经得到业界的认可，目前很多企业级的项目都会部署到Linux/unix系统上。
3. 常见的操作系统(win7、IOS、Android、Mac)



- Linux的介绍

概述:

4. Linux 吉祥物



Tux

- **Linux**的介绍

概述:

5.Linux的创始人

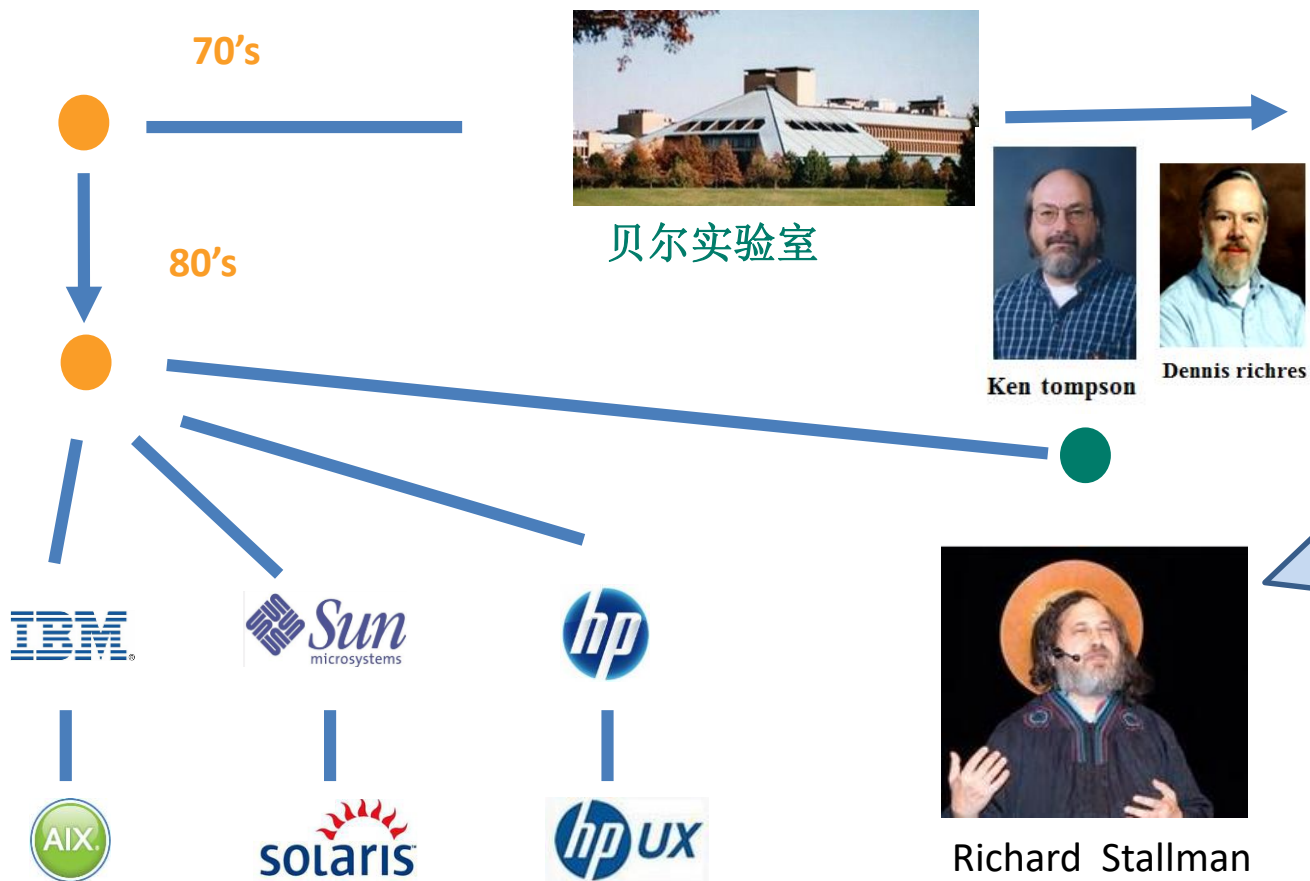


Linus Torvalds

6.Linux主要的发行版:

Ubuntu(乌班图)、**RedHat**(红帽)、**CentOS**、**Debian**[蝶变]、**Fedora**、**SuSE**、**OpenSUSE**

Linux和Unix的关系-unix是怎么来的



- 在自由的时代用户应该免费享有对软件源代码阅读、修改的权利。
- 软件公司可以靠提供服务和训练获得盈利。



Richard Stallman

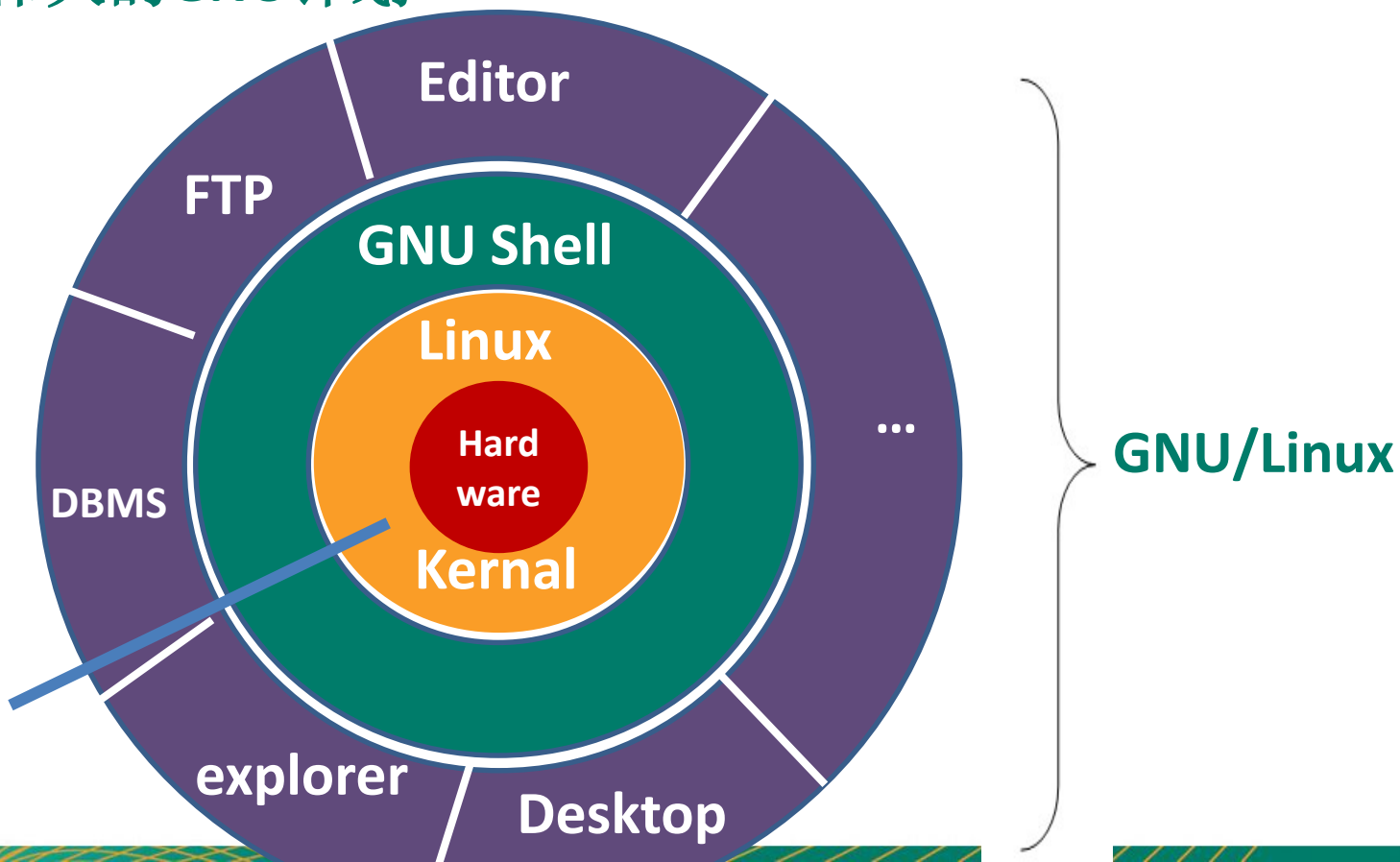


Linus Torvalds

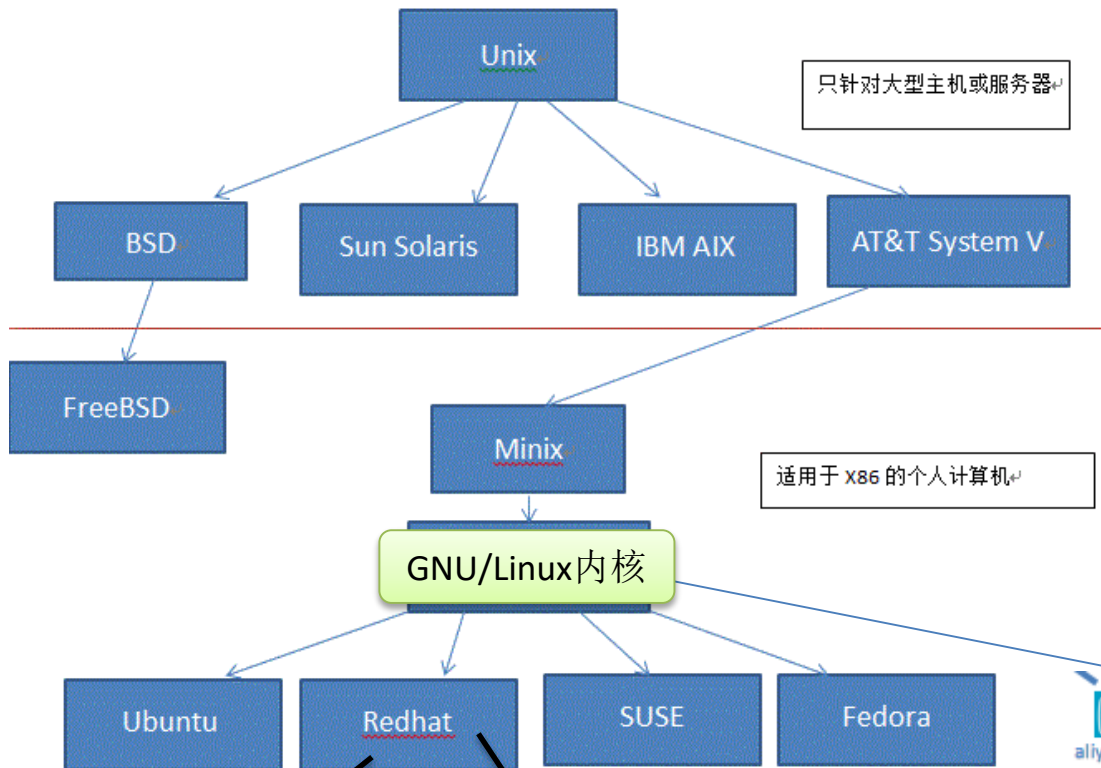
和Unix的关系-Linux是怎么来的 伟大的GNU计划



通俗版



Linux和Unix的关系



Ken Thompson



Dennis Riches



Linus Torvalds



● Linux和Windows比较(了解)

比较

Window

Linux

免费与收费

收费且很贵

Linux 免费或少许费用。

软件与支持

数量和质量的优势，不过大部分为收费软件；由微软官方提供支持和服

开源自由软件，用户可以修改定制和再发布，由于基本免费没有资金支持，部分软件质量和体验欠缺；有全球所有的Linux开发者和自由软件社区提供支持。

安全性

三天两头打补丁安装系统安全更新，还是会中病毒木马；

要说 Linux 没有安全问题，那当然是不可能的，这一点仁者见仁智者见智，相对来说肯定比 Windows 平台要更加安全

使用习惯

普通用户基本都是纯图形界面下操作使用，依靠鼠标和键盘完成一切操作，用户上手容易入门简单；

兼具图形界面操作和完全的命令行操作，可以只用键盘完成一切操作，新手入门较困难，需要一些学习和指导，一旦熟练之后效率极高。

可定制性

封闭的，系统可定制性很差；

开源，可定制化非常强。

应用场景

桌面操作系统主要使用的是window。

支撑百度，谷歌，淘宝等应用软件和服务的，是后台成千上万的Linux服务器主机。世界上大部分软件和服务都是运行在Linux之上的。

Linux基础篇

VM和Linux的安装

- 安装vm和Centos

基本说明:

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



- 安装vm和Centos

Centos下载地址:

CentOS下载地址:

- 网易镜像: <http://mirrors.163.com/centos/6/isos/>
- 搜狐镜像: <http://mirrors.sohu.com/centos/6/isos>

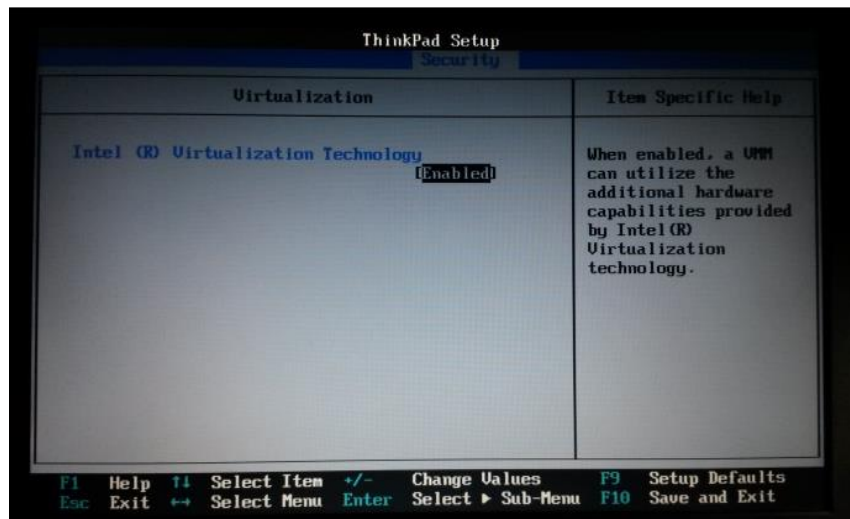
- 安装vm和Centos

VM安装的步骤



尚硅谷大数据技术之安

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)



- 安装vm和Centos

CentOS安装的步骤

- 1) 创建虚拟机
- 2) 开始安装系统 (CentOS6.8)



尚硅谷大数据技术之Linux安装VM Cent



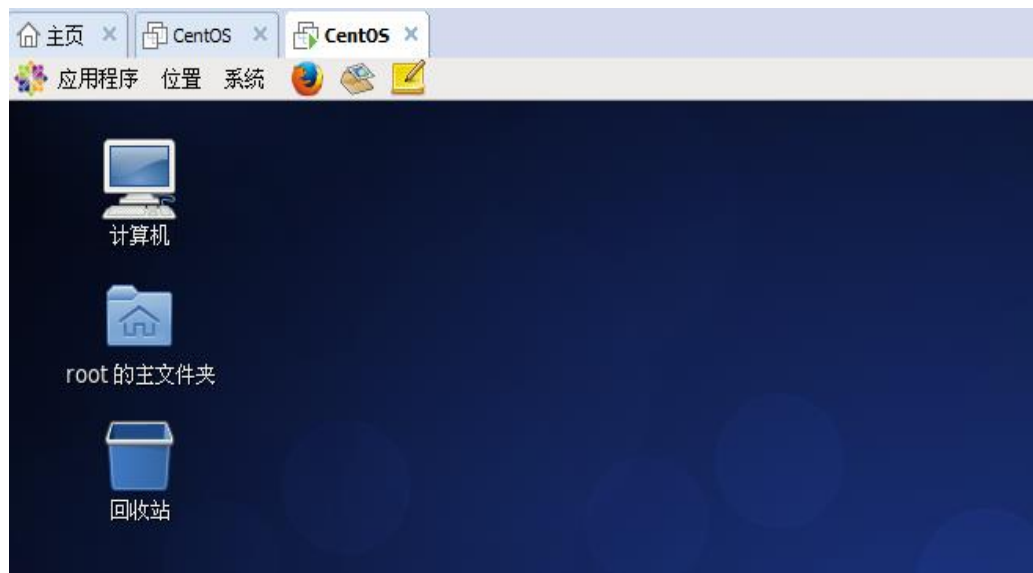
- 安装vm和Centos

CentOS安装技术难点-网络设置

- 安装vm和Centos

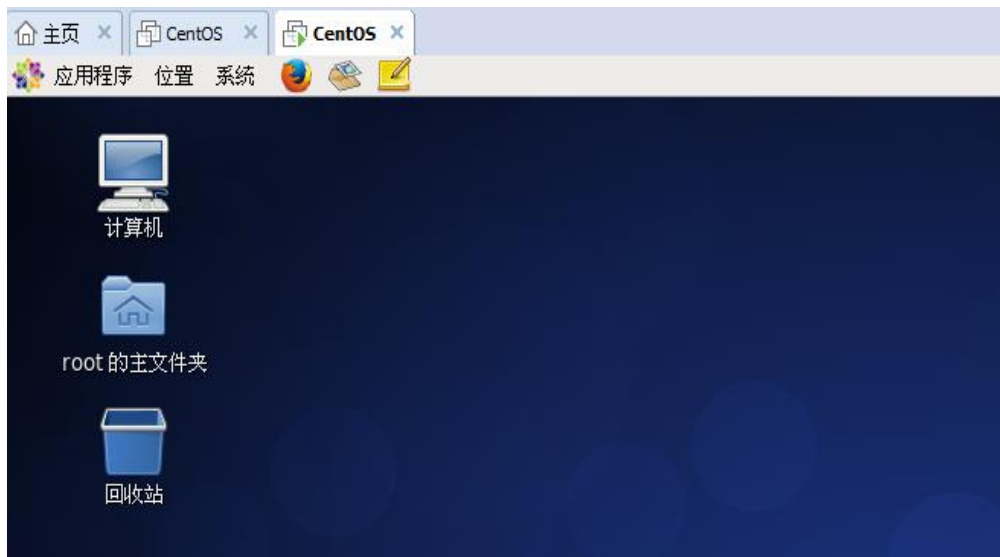
CentOS安装的步骤

3) 安装成功后的界面



- 安装虚拟机和CentOS的练习

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， 请各位同学完成安装。



- 安装vmtools

介绍

vmtools 安装后，可以让我们在windows下更好的管理vm虚拟机

- 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

● 安装vmtools

安装vmtools的步骤:

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

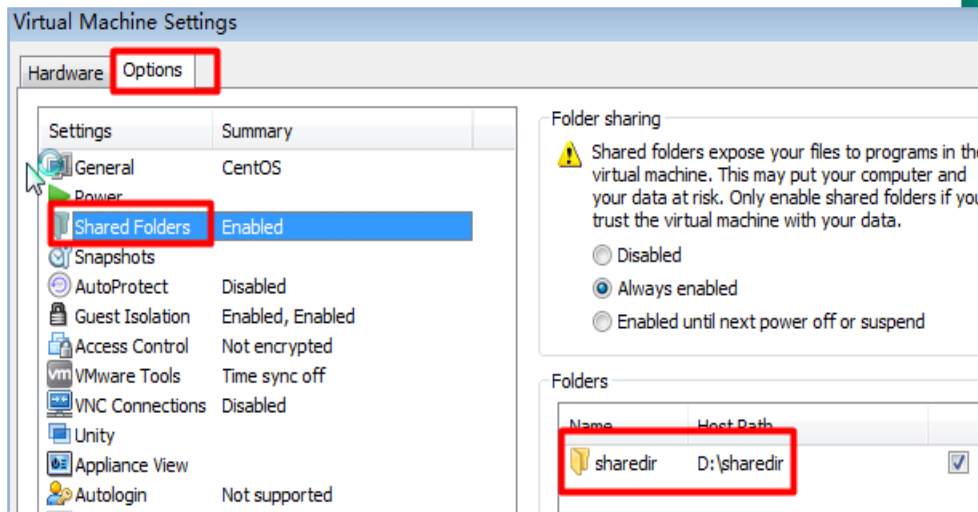
● 设置共享文件夹

基本介绍:

- 1) 为了方便，可以设置一个共享文件夹，比如d:/share
- 2) windows 和 contos 就可以共享文件了，但是在实际公司开发中，文件的上传下载是需要使用远程方式完成的，
- 3) 远程方式登录，我们后面会具体讲解

具体步骤:

- 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

请安装vmtools:

- 1) 并且设置一个共享文件夹c:/myDir，该文件编写一个hello.txt 文件
- 2) 在linux 这边打开，可以看到hello.txt文件

Linux基础篇

Linux的目录结构



- **Linux目录结构**

基本介绍:

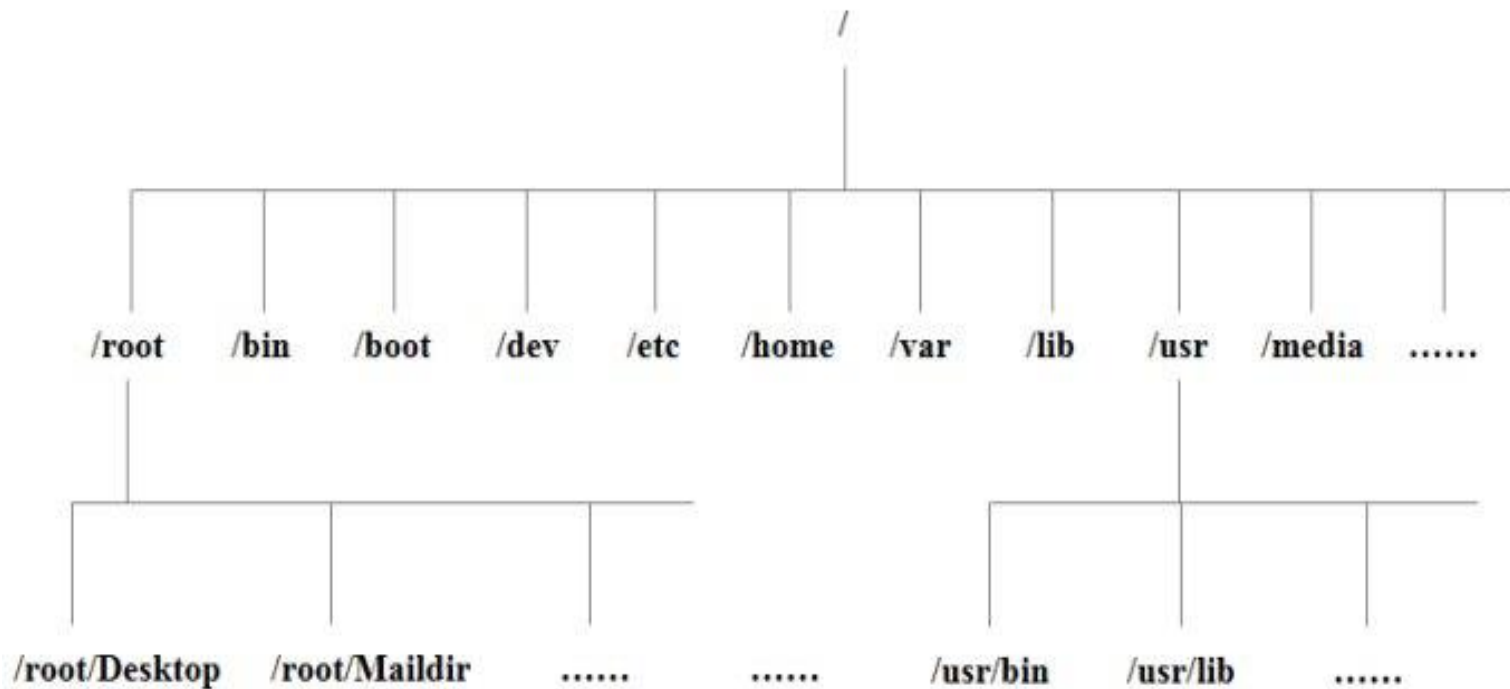
linux的文件系统是采用级层式的树状目录结构，在此结构中的最上层是根目录“/”，然后在此目录下再创建其他的目录。

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。**

• Linux目录结构

基本介绍:





● Linux目录结构

具体的目录结构:

- `/bin` [重点] (`/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` [重点]
 - 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` [重点]
 - 该目录为系统管理员, 也称作超级权限者的用户主目录。



● Linux目录结构

具体的目录结构:

➤ /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 [重点]

- 所有的系统管理所需要的配置文件和子目录 my.conf

➤ /usr [重点]

- 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。



● Linux目录结构

➤ /boot [重点]

- 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

➤ /proc

- 这个目录是一个虚拟的目录，它是系统内存的映射，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

➤ /tmp

- 这个目录是用来存放一些临时文件的。



● Linux目录结构

➤ /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 [重点]

- 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 [重点]

- 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。 d:/myshare



● Linux目录结构

➤ /opt

- 这是给主机额外安装软件所摆放的目录。如安装ORACLE数据库就可放到该目录下。默认为空。

➤ /usr/local [重点]

- 这是另一个给主机额外安装软件所安装的目录。一般是通过编译源码方式安装的程序。

➤ /var [重点]

- 这个目录中存放着在不断扩充着的东西，习惯将经常被修改的目录放在这个目录下。包括各种日志文件。

➤ /selinux [security-enhanced linux] 360

- SELinux是一种安全子系统,它能控制程序只能访问特定文件。

Linux实操篇

远程登录到Linux服务器

● 为什么需要远程登录Linux

说明: 公司开发时候, 具体的情况是这样的

- 1) linux服务器是开发小组共享的.
- 2) 正式上线的项目是运行在公网的.
- 3) 因此程序员需要远程登录到centos进行项目管理或者开发.
- 4) 画出简单的网络拓扑示意图(帮助理解)
- 5) 远程登录客户端有 Xshell5, Xftp5, 我们学习使用 Xshell5 和 Xftp, 其它的远程工具大同小异.

- 远程登录Linux-Xshell5

介绍:

说明: Xshell 是目前最好的远程登录到Linux操作的软件, 流畅的速度并且完美解决了中文乱码的问题, 是目前程序员首选的软件。

Xshell^[1] 是一个强大的安全终端模拟软件, 它支持SSH1, SSH2, 以及Microsoft Windows 平台的TELNET 协议。

Xshell可以在Windows界面下用来访问远端不同系统下的服务器, 从而比较好的达到远程控制终端的目的。





- 远程登录Linux-Xshell5

安装配置和使用:

具体看老师的演示和操作!



- 远程上传下载文件Xftp5

介绍:

是一个基于 [windows](#) 平台的功能强大的[SFTP](#)、[FTP](#) 文件传输软件。使用了Xftp 以后，windows 用户能安全地在 [UNIX/Linux](#) 和 Windows PC 之间传输文件。(示意图)



- 远程登录Linux-Xftp

Xftp5安装配置和使用:

具体看老师的演示和操作！

细节讨论

如何处理Xftp中文乱码问题。



- **Xftp5和XShell5的使用**

练习要求:

1. 通过Xftp5软件可以连接到linux ,并传输一个文件 到 /root目录
2. 使用xshell可以连接到linux ,并且可以重新启动一把 linux系统 **【reboot】**

Linux实操篇

Vi和Vim编辑器

- vi和vim的基本介绍

所有的 Linux 系统都会内建 vi 文本编辑器。

Vim 具有程序编辑的能力，可以看做是Vi的增强版本，可以主动的以字体颜色辨别语法的正确性，方便程序设计。代码补完、编译及错误跳转等方便编程的功能特别丰富，在程序员中被广泛使用。





- **vi和vim常用的三种模式**

正常模式:

以 **vim** 打开一个档案就直接进入一般模式了(这是默认的模式)。在这个模式中， 你可以使用『上下左右』按键来移动光标，你可以使用『删除字符』或『删除整行』来处理档案内容，也可以使用『复制、贴上』来处理你的文件数据。

插入模式:

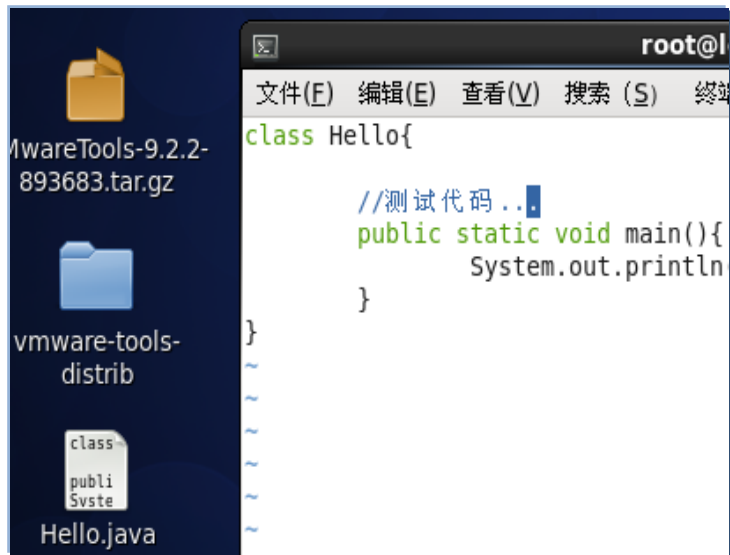
按下 **i, l, o, O, a, A, r, R** 等任何一个字母之后才会进入编辑模式, 一般来说按 **i** 即可.

命令行模式

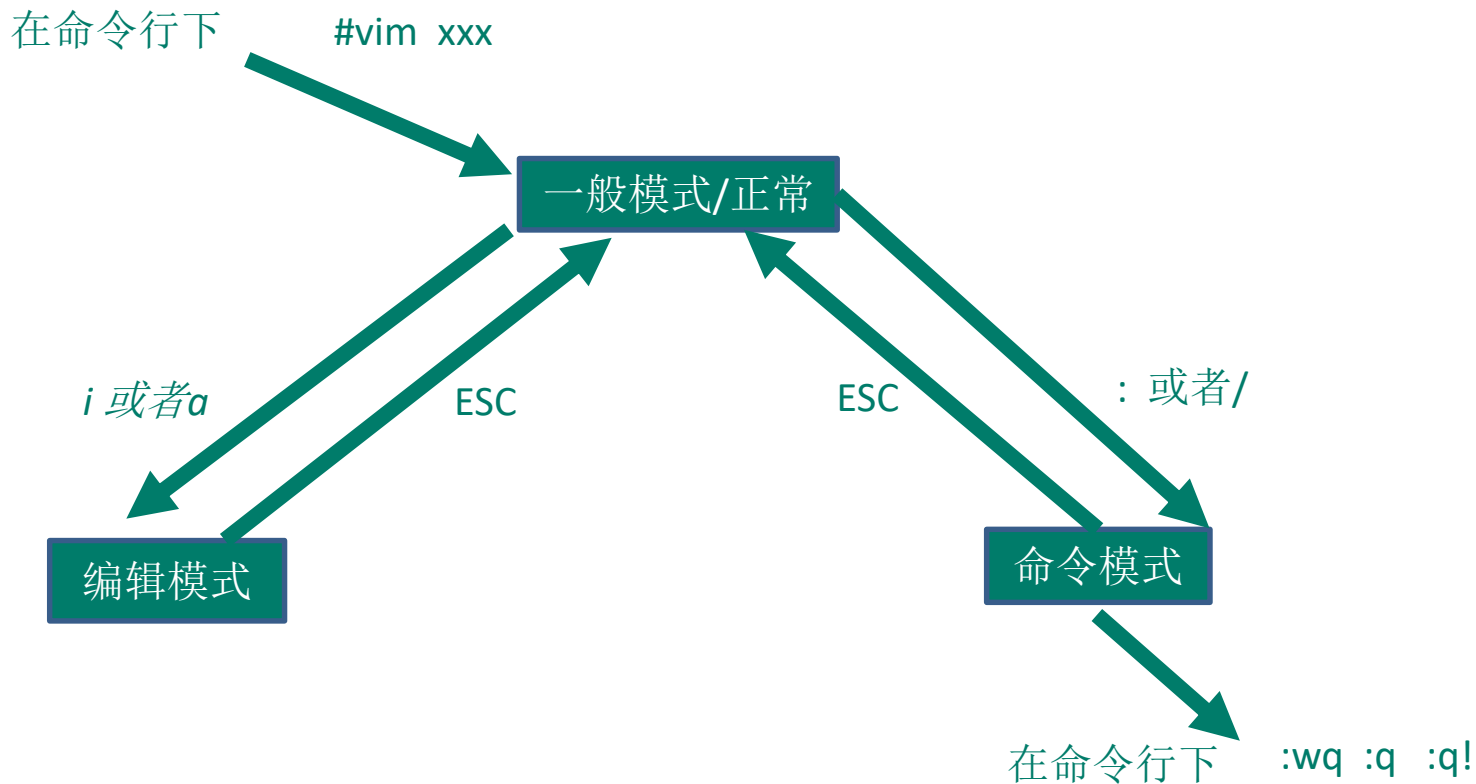
在这个模式当中， 可以提供你相关指令，完成读取、存盘、替换、离开 **vim**、显示行号等的动作则是在此模式中达成的！

- vi和vim基本使用

使用 vim 开发一个 Hello.java 程序, 保存.步骤说明和演示:



- vi和vim模式的相互切换



● vi和vim快捷键

快捷键使用练习:



vi 和 vim 的常用快

- 1) 拷贝当前行 yy, 拷贝当前行向下的5行 5yy, 并粘贴。
- 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]
- 4) 设置文件的行号, 取消文件的行号.[命令行下 :set nu 和 :set nonu]
- 5) 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]
- 6) 在一个文件中输入 "hello", 然后又撤销这个动作 u
- 7) 编辑 /etc/profile 文件, 并将光标移动到 20行 shift+g
- 8) 更多的看整理的文档

vi / vim 键盘图

Esc

命令
模式

~ 转换 大小写	! 外部 过滤器	@. 运行 宏	# prev ident	\$ 行末	% 括号 匹配	^ "软" 行首	& 重复 :s	* next ident	(句首) 下一 句首	"soft" bol down	+ 后一行 行首
· 跳转到 标注	1	2	3	4	5	6	7	8	9	0 "硬" 行首	- 前一行 行首	= 自动 ³ 格式化
Q 切换到 ex模式	W 下一 单词	E 词尾	R 替换 模式	T back 'till	Y 拷贝 行	U 撤消 行内命令	I 到行首 插入	O 分段 (前)	P 粘贴 (前)	{ 段首	}	段尾
q 录制 宏	w 下一 单词	e 词尾	r 替换 字符	t 'till	y 拷贝 ^{1,3}	u 撤消 命令	i 插入 模式	o 分段 (后)	p 粘贴 ¹ (后)	[杂项]	杂项
A 在行末 附加	S 删除行 并插入	D 删除 至行末	F 行内字符 反向查找	G 文尾/ 行号	H 屏幕 顶行	J 合并 两行	K 帮助	L 屏幕 底行	:	ex 命令	" 寄存器 ¹ 标识	行首/ 列
a 附加	s 删除字符 并插入	d 删除 ^{1,3}	f 行内字符 查找	g 附加 命令 ⁶	h ←	j ↓	k ↑	l →	:	重复 ; t/T/f/F	' 跳转到标 注的行首	\ 未使用!
Z 退出 ⁴	X 退格	C 修改 至行末	V 可视 行模式	B 前一 单词	N 查找 上一处	M 屏幕 中间行	< 反缩进 ³	> 缩进 ³	?	向前 搜索		
Z 附加 命令 ⁵	X 删除 (字符)	c 修改 ^{1,3}	V 可视 模式	b 前一 单词	n 查找 下一处	m 设置 标注	反向 , t/T/f/F	重复 命令	/	向后 搜索		

动作

移动光标, 或者定义操作的范围

命令

直接执行的命令,
红色命令进入编辑模式

操作

后面跟随表示操作范围的指令

extra

特殊功能,
需要额外的输入

q

后跟字符参数

w,e,b命令

小写(b): quux(**f**oo, **b**ar, **b**az);
大写(B): quux(**F**oo, **B**ar, **B**az);

主要ex命令:

:w (保存), :q (退出), :q! (不保存退出)
:e f (打开文件 f),
:%s/x/y/g ('y' 全局替换 'x'),
:h (帮助 in vim), :new (新建文件 in vim)

其它重要命令:

CTRL-R: 重复 (vim),
CTRL-F/-B: 上翻/下翻,
CTRL-E/-Y: 上滚/下滚,
CTRL-V: 块可视模式 (vim only)

可视模式:

漫游后对选中的区域执行操作 (vim only)

备注:

- (1) 在 拷贝/粘贴/删除 命令前使用 "x (x=a..z,") 使用命令的寄存器('剪贴板') (如: "ay\$ 拷贝剩余的行内容至寄存器 'a')
- (2) 命令前添加数字 多遍重复操作 (e.g.: 2p, d2w, 5i, d4j)
- (3) 重复本字符在光标所在行执行操作 (dd = 删除本行, >> = 行首缩进)
- (4) ZZ 保存退出, ZQ 不保存退出
- (5) zt: 移动光标所在行至屏幕顶端, zb: 底端, zz: 中间
- (6) gg: 文首 (vim only), gf: 打开光标处的文件名 (vim only)

- **vi和vim课堂练习**

- 1) 看使用vim 编写一个Hello.java
- 2) 可以通过javac 来编译，通过 java 来运行 【**暂不编译**，创建好即可】
- 3) 把老师课堂讲的各种快捷键自己练习一把即可

Linux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 -h now`

立即进行关机

`shutdown -h 1`

"hello, 1 分钟后会关机了"

`shutdown -r now`

现在重新启动计算机

`halt`

关机，作用和上面一样。

`reboot`

现在重新启动计算机

`sync`

把内存的数据同步到磁盘。

注意细节:

不管是重启系统还是关闭系统，首先要运行`sync`命令，把内存中的数据写到磁盘中

● 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

Linux实操篇

用户管理



● 基本介绍

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。

● 添加用户

基本语法

useradd 用户名

应用案例

1) 案例1: 添加一个用户 xiaoming

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



- 指定/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

应用案例

- 1) 删除用户xiaoming，但是要保留家目录
- 2) 删除用户以及用户主目录

细节说明

- 1) 是否保留家目录的讨论？



● 查询用户信息指令

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



- 查看当前用户/登录用户

基本语法

whoami/ who am I

- 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。

新增组

- 指令

groupadd 组名

- 案例演示

- 用户组

- 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

- 增加一个用户 `zwj`, 直接将他指定到 `wudang`

- 用户组

修改用户的组

- 指令(基本语法)

`usermod -g 用户组 用户名`

- 案例演示

- 用户和组的相关文件

- /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

Linux基础篇

实用指令



● 指定运行级别

基本介绍:

运行级别说明:

- 0: 关机
- 1: 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

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

命令: `init [012356]`

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。

- 指定运行级别

面试题:

如何找回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，直接进入到了 命令行终端(3)



● 帮助指令

man 获得帮助信息

- 基本语法
man [命令或配置文件]（功能描述：获得帮助信息）
- 应用实例
案例：查看ls命令的帮助信息

help指令

- 基本语法
help 命令（功能描述：获得shell内置命令的帮助信息）
- 应用实例
案例：查看cd命令的帮助信息

百度帮助更直接

虽然上面两个都可以来获取指令帮助，但是需要英语功底，如果英语不太好的，我还是推荐大家直接百度靠谱。



- 文件目录类

pwd 指令

- 基本语法
 pwd (功能描述: 显示当前工作目录的绝对路径)
- 应用实例
 案例: 显示当前工作目录的绝对路径

ls指令

- 基本语法
 ls [选项] [目录或是文件]
- 常用选项
 -a: 显示当前目录所有的文件和目录, 包括隐藏的。
 -l: 以列表的方式显示信息
- 应用实例
 案例: 查看当前目录的所有内容信息



● 文件目录类

cd 指令

- 基本语法
cd [参数] (功能描述: 切换到指定目录)
- 常用参数
绝对路径和相对路径
cd ~ 或者 cd : 回到自己的家目录
cd .. 回到当前目录的上一级目录
- 应用实例
案例1: 使用绝对路径切换到root目录
案例2: 使用相对路径到/root 目录
案例3: 表示回到当前目录的上一级目录
案例4: 回到家目录

- 文件目录类

mkdir指令

mkdir指令用于创建目录

- 基本语法
 - mkdir [选项] 要创建的目录
- 常用选项
 - p：创建多级目录
- 应用实例

案例1:创建一个目录 /home/dog

案例2:创建多级目录 /home/animal/tiger

- 文件目录类

rmmdir指令

rmmdir指令删除空目录

- 基本语法

rmmdir [选项] 要删除的空目录

- 应用实例

案例1:删除一个目录 /home/dog

- 使用细节

rmmdir 删除的是空目录，如果目录下有内容时无法删除的。

提示：如果需要删除非空目录，需要使用 `rm -rf` 要删除的目录



- 文件目录类

touch指令

touch指令创建空文件

- 基本语法

touch 文件名称

- 应用实例

案例1: 创建一个空文件 hello.txt



- 文件目录类

cp指令

cp 指令拷贝文件到指定目录

- 基本语法
 - cp [选项] source dest
- 常用选项
 - r : 递归复制整个文件夹
- 应用实例
 - 案例1: 将 /home/aaa.txt 拷贝到 /home/bbb 目录下
 - 案例2: 递归复制整个文件夹，举例
- 使用细节
 - 强制覆盖不提示的方法: \cp

- 文件目录类

rm指令

rm 指令移除文件或目录

- 基本语法

rm [选项] 要删除的文件或目录

- 常用选项

-r : 递归删除整个文件夹

-f : 强制删除不提示

- 应用实例

案例1: 将 /home/aaa.txt 删除

案例2: 递归删除整个文件夹 /home/bbb

- 使用细节

强制删除不提示的方法: 带上 -f 参数即可



- 文件目录类

mv指令

mv 移动文件与目录或重命名

- 基本语法

mv oldNameFile newNameFile (功能描述: 重命名)

mv /temp/movefile /targetFolder (功能描述: 移动文件)

- 应用实例

案例1: 将 /home/aaa.txt 文件 重新命名为 pig.txt

案例2: 将 /home/pig.txt 文件 移动到 /root 目录下



- 文件目录类

cat指令

cat 查看文件内容

- 基本语法

- cat [选项] 要查看的文件

- 常用选项

- n : 显示行号

- 应用实例

- 案例1: /etc/profile 文件内容，并显示行号

- 使用细节

- cat 只能浏览文件，而不能修改文件，为了浏览方便，一般会带上 管道命令 | more

• 文件目录类

more指令

more指令是一个基于VI编辑器的文本过滤器，它以全屏幕的方式按页显示文本文件的内容。
more指令中内置了若干快捷键，详见操作说明

- 基本语法
 more 要查看的文件
- 操作说明
- 应用实例
 案例: 采用more查看文件
 /etc/profile

操作	功能说明
空白键 (space)	代表向下翻一页；
Enter	代表向下翻『一行』；
q	代表立刻离开 more ，不再显示该文件内容。
Ctrl+F	向下滚动一屏
Ctrl+B	返回上一屏
=	输出当前行的行号
:f	输出文件名和当前行的行号

● 文件目录类

less指令

less指令用来分屏查看文件内容，它的功能与more指令类似，但是比more指令更加强大，支持各种显示终端。less指令在显示文件内容时，并不是一次将整个文件加载之后才显示，而是根据显示需要加载内容，**对于显示大型文件具有较高的效率。**

- 基本语法

- less 要查看的文件

- 操作说明

- 应用实例

- 案例: 采用less查看一个大文件文件

- `/opt/金庸-射雕英雄传txt精校版.txt`

操作	功能说明
空白键	向下翻动一页;
[pagedown]	向下翻动一页
[pageup]	向上翻动一页;
/字符串	向下搜寻『字符串』的功能; n: 向下查找; N: 向上查找;
?字符串	向上搜寻『字符串』的功能; n: 向上查找; N: 向下查找;
q	离开 less 这个程序;



● 文件目录类

echo指令

echo输出内容到控制台。

- 基本语法
echo [选项] [输出内容]
- 应用实例
案例: 使用echo 指令输出环境变量
案例: 使用echo 指令输出 hello,world!

head指令

head用于显示文件的开头部分内容，默认情况下head指令显示文件的前10行内容

- 基本语法
head 文件 (功能描述: 查看文件头10行内容)
head -n 5 文件 (功能描述: 查看文件头5行内容, 5可以是任意行数)
- 应用实例
案例: 查看/etc/profile 的前面5行代码

- 文件目录类

tail指令

tail用于输出文件中尾部的内容，默认情况下tail指令显示文件的前10行内容。

- 基本语法

- 1) tail 文件 (功能描述: 查看文件头10行内容)
- 2) tail -n 5 文件 (功能描述: 查看文件头5行内容, 5可以是任意行数)
- 3) tail -f 文件 (功能描述: 实时追踪该文档的所有更新)

- 应用实例

案例1: 查看/etc/profile 最后5行的代码

案例2: 实时监控 mydate.txt , 看看到文件有变化时, 是否看到, 实时的追加日期

- 文件目录类

- > 指令 和 >> 指令

- > 输出重定向和 >> 追加

- 基本语法

- 1) ls -l > 文件

- (功能描述: 列表的内容写入文件a.txt中(覆盖写))

- 2) ls -al >> 文件

- (功能描述: 列表的内容追加到文件aa.txt的末尾)

- 3) cat 文件1 > 文件2

- (功能描述: 将文件1的内容覆盖到文件2)

- 4) echo "内容">> 文件

- 应用实例

- 案例1: 将 /home 目录下的文件列表 写入到 /home/info.txt 中

- 案例2: 将当前日历信息 追加到 /home/mycal 文件中



• 文件目录类

In 指令

软链接也成为符号链接，类似于windows里的快捷方式，主要存放了链接其他文件的路径

- 基本语法

`ln -s [原文件或目录] [软链接名]`（功能描述：给原文件创建一个软链接）

- 应用实例

案例1: 在/home 目录下创建一个软连接 linkToRoot，连接到 /root 目录

案例2: 删除软连接 linkToRoot

```
[root@hsp home]# ln -s /root/ myroot
[root@hsp home]# ll
总用量 0
lrwxrwxrwx. 1 root root 6 3月  7 01:28 myroot -> /root/
```

- 细节说明

当我们使用pwd指令查看目录时，仍然看到的是软链接所在目录。

- 文件目录类

history指令

查看已经执行过历史命令,也可以执行历史指令

- 基本语法

history （功能描述：查看已经执行过历史命令）

- 应用实例

案例1: 显示所有的历史命令

案例2: 显示最近使用过的10个指令。

案例3: 执行历史编号为5的指令

- 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)
- 2) date +%Y (功能描述: 显示当前年份)
- 3) date +%m (功能描述: 显示当前月份)
- 4) date +%d (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



● 时间日期类

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 2020-11-11 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

选项	功能
-name<查询方式>	按照指定的文件名查找模式查找文件
-user<用户名>	查找属于指定用户名所有文件
-size<文件大小>	按照指定的文件大小查找文件。

- 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200m的文件 (+n 大于 -n小于 n等于)

● 搜索查找类

locate指令

locate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

● 搜索查找类

grep指令和 管道符号 |

grep 过滤查找，管道符，“|”，表示将前一个命令的处理结果输出传递给后面的命令处理。

- 基本语法

grep [选项] 查找内容 源文件

- 常用选项

选项	功能
-n	显示匹配行及行号。
-i	忽略字母大小写

- 应用实例

案例1: 请在 `hello.txt` 文件中，查找 "yes" 所在行，并且显示行号

- 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



● 压缩和解压类

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 mypackage.zip 解压到 /opt/tmp 目录下

● 压缩和解压类

tar 指令

tar 指令 是打包指令，最后打包后的文件是 .tar.gz 的文件。

- 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

选项	功能
-c	产生.tar打包文件
-v	显示详细信息
-f	指定压缩后的文件名
-z	打包同时压缩
-x	解包.tar文件

- 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

Linux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

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 zhangfei



- 组的创建

基本指令

groupadd 组名

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，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 `fruit`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

将 `zwj` 这个用户从原来所在组，修改到 `wudang` 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:

```
-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 $rwx=4+2+1=7$

1

文件: 硬连接数或 目录: 子目录数

root

用户

root

组

1213

文件大小(字节), 如果是文件夹, 显示 4096字节

Feb 2 09:39

最后修改日期

abc

文件名

● 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。

文件 类型	属主 权限			属组 权限			其他用户 权限		
0	1	2	3	4	5	6	7	8	9
d	rwX			r-X			r-X		
目录 文件	读	写	执行	读	写	执行	读	写	执行

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。
- 2) 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



- 修改权限-**chmod**

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

- 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：



● 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



- 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 `shaolin` (少林)
- 2) 请将 `/home/kkk` 目录下所有的文件和目录的所在组都修改成 `shaolin`(少林)

- 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

(2) 创建用户

(3) 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.

● 课后练习

练习文件权限管理[课堂练习]

建立两个组（神仙,妖怪）

建立四个用户(唐僧,悟空，八戒，沙僧)

设置密码

把悟空，八戒放入妖怪 唐僧 沙僧 在神仙

用悟空建立一个文件（`monkey.java` 该文件要输出 `i am monkey`）

给八戒一个可以 `r w` 的权限

八戒修改 `monkey.java` 加入一句话(`i am pig`)

唐僧 沙僧 对该文件没有权限

把 沙僧 放入妖怪组

让沙僧 修改 该文件 `monkey`, 加入一句话 ("`我是沙僧，我是妖怪!`");

● 课后练习

- 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

Linux实操篇

定时任务调度



● crond 任务调度

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

-e	编辑crontab定时任务
-l	查询crontab任务
-r	删除当前用户所有的crontab任务



● crond 任务调度

快速入门

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

参数细节说明

• 5个占位符的说明

项目	含义	范围
第一个“*”	一小时当中的第几分钟	0-59
第二个“*”	一天当中的第几小时	0-23
第三个“*”	一个月当中的第几天	1-31
第四个“*”	一年当中的第几月	1-12
第五个“*”	一周当中的星期几	0-7（0和7都代表星期日）

● crond 任务调度

参数细节说明

• 特殊符号的说明

特殊符号	含义
*	代表任何时间。比如第一个“*”就代表一小时中每分钟都执行一次的意思。
,	代表不连续的时间。比如“0 8,12,16 * * * 命令”，就代表在每天的8点0分，12点0分，16点0分都执行一次命令
-	代表连续的时间范围。比如“0 5 * * 1-6命令”，代表在周一到周六的凌晨5点0分执行命令
/n	代表每隔多久执行一次。比如“/10 * * * * 命令”，代表每隔10分钟就执行一遍命令

● crond 任务调度

参数细节说明

· 特定时间执行任务案例

时间	含义
45 22 * * * 命令	在22点45分执行命令
0 17 * * 1 命令	每周1 的17点0分执行命令
0 5 1,15 * * 命令	每月1号和15号的凌晨5点0分执行命令
40 4 * * 1-5 命令	每周一到周五的凌晨4点40分执行命令
*/10 4 * * * 命令	每天的凌晨4点，每隔10分钟执行一次命令
0 0 1,15 * 1 命令	每月1号和15号，每周1的0点0分都会执行命令。注意：星期几和几号最好不要同时出现，因为他们定义的都是天。非常容易让管理员混乱。

● crond 任务调度

应用实例

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

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

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

crond 相关指令:

- 1) `conrtab -r`: 终止任务调度。
- 2) `crontab -l`: 列出当前有那些任务调度
- 3) `service crond restart` [重启任务调度]

Linux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:

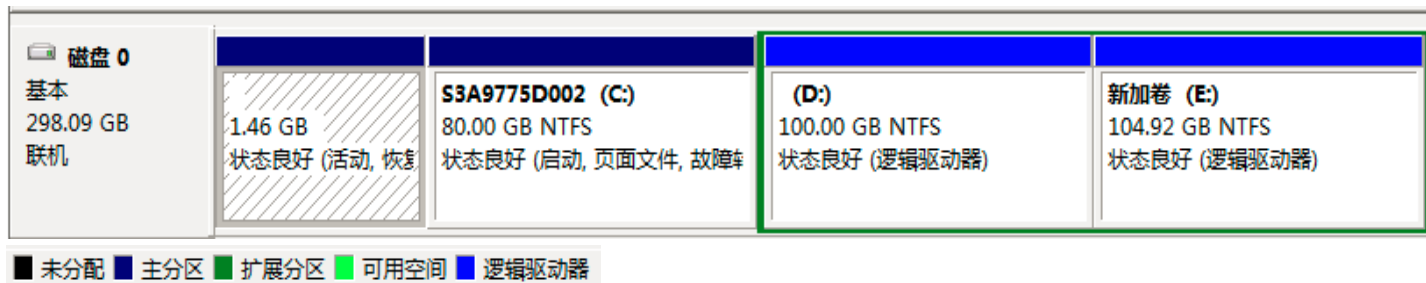
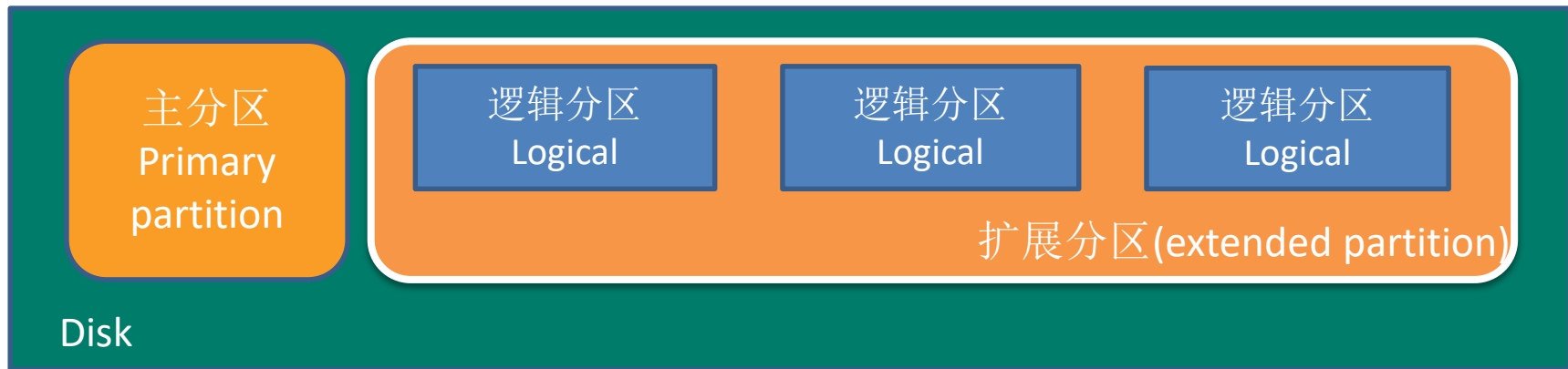
- 1.最多支持四个主分区
- 2.系统只能安装在主分区
- 3.扩展分区要占一个主分区
- 4.MBR最大只支持2TB，但拥有最好的兼容性

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

- 分区基础知识

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



图1.2i



● Linux分区

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。



● Linux分区

查看所有设备挂载情况

命令：lsblk 或者 lsblk -f

```
[root@hsp ~]# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sr0         11:0    1 1024M  0 rom
sda          8:0     0   20G  0 disk
├─sda1       8:1     0  200M  0 part /boot
├─sda2       8:2     0    2G  0 part [SWAP]
└─sda3       8:3     0 17.8G  0 part /
```

```
[root@hsp ~]# lsblk -f
NAME        FSTYPE LABEL UUID                                MOUNTPOINT
sr0
sda
├─sda1 ext4          ebbec079-ba40-486c-b03b-588cfc191460 /boot
├─sda2 swap          6a7e0a69-969c-4f07-9c72-782e3b299f47 [SWAP]
└─sda3 ext4          4402a96e-bfe1-4baa-9dd9-3968dd02adc1 /
```

- 挂载的经典案例

说明：

下面我们以增加一块硬盘为例来熟悉下磁盘的相关指令和深入理解磁盘分区、挂载、卸载的概念。

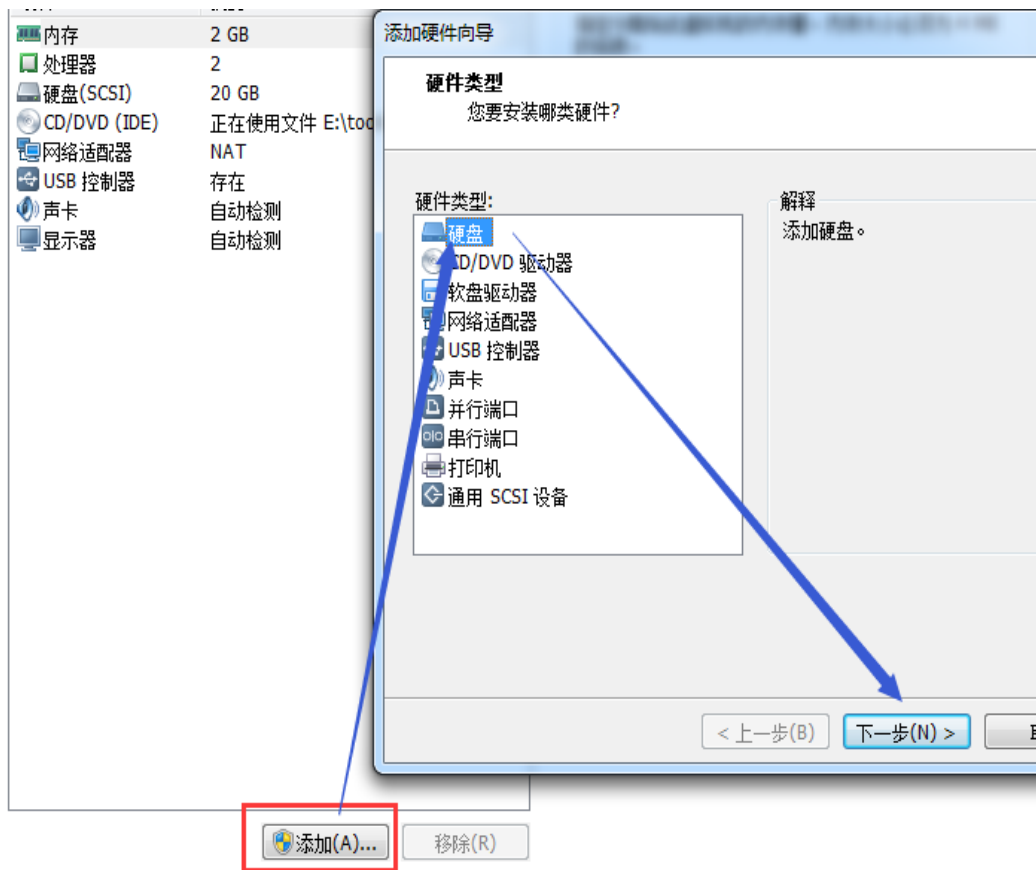
如何增加一块硬盘

- 1) 虚拟机添加硬盘
- 2) 分区
- 3) 格式化
- 4) 挂载
- 5) 设置可以自动挂载。

● 挂载的经典案例

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）！





● 挂载的经典案例

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`。

```
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Selected partition 4
First cylinder (2233-2610, default 2233): 回车, 默认全部剩余空间
Using default value 2233
Last cylinder, +cylinders or +size{K,M,G} (2233-2610, default 2610): 同上
Using default value 2610
```


- 挂载的经典案例

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

● 挂载的经典案例

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`
- 用命令行挂载重启后会失效

虚拟机增加硬盘步骤5

永久挂载: 通过修改/etc/fstab实现挂载
添加完成后 执行 `mount -a` 即刻生效

```
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/sdb1 /newdisk ext4 defaults 0 0
UUID=2563489-2961-4b79-aa31-9d6e346f2b15 / ext4 defaults 0 0
UUID=27f8ef79-d6e0-42b2-95ca-20a348b46666 /boot ext4 defaults 1 2
UUID=506af6cc-fb37-4279-af1c-72b2a7adc992 swap swap defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
```

- 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



- 磁盘情况查询

查询指定目录的磁盘占用情况

- 基本语法

`du -h /目录`

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
- 3) 统计/home文件夹下文件的个数，包括子文件夹里的
- 4) 统计文件夹下目录的个数，包括子文件夹里的
- 5) 以树状显示目录结构

```
[root@hsp aaa]# tree /home/  
/home/  
├── aaa  
│   ├── abc.txt  
│   └── bbb  
│       └── ttt.txt  
├── myroot -> /root/  
└── newdisk
```


Linux实操篇

网络配置



- **Linux网络配置原理图**

画出原理图，帮助理解

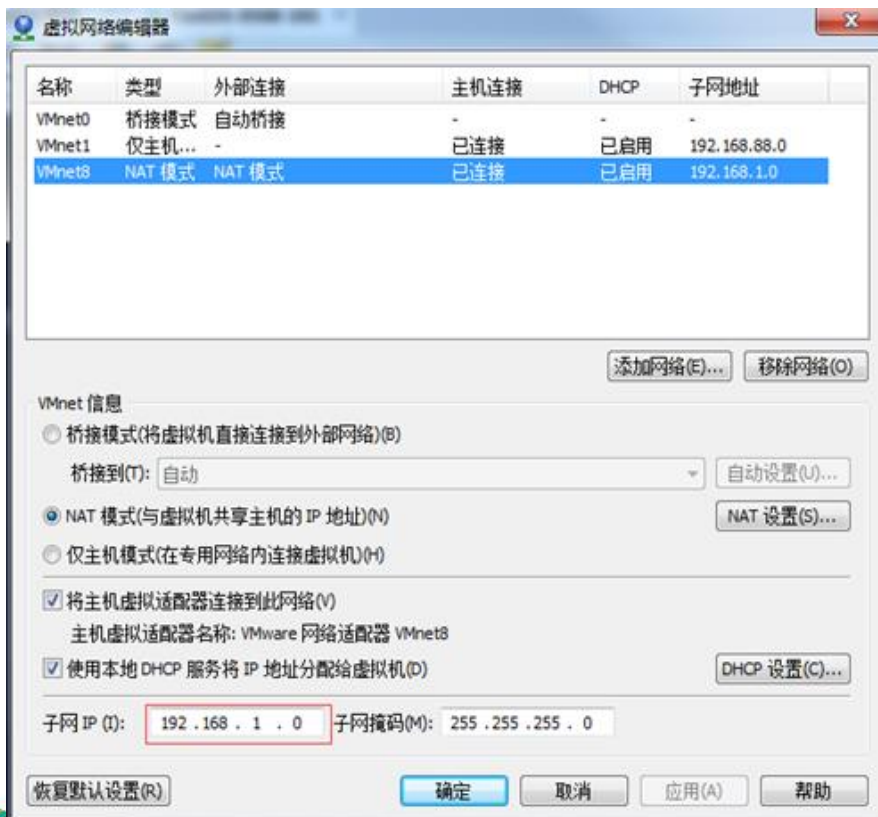
- 查看网络IP和网关

查看虚拟网络编辑器



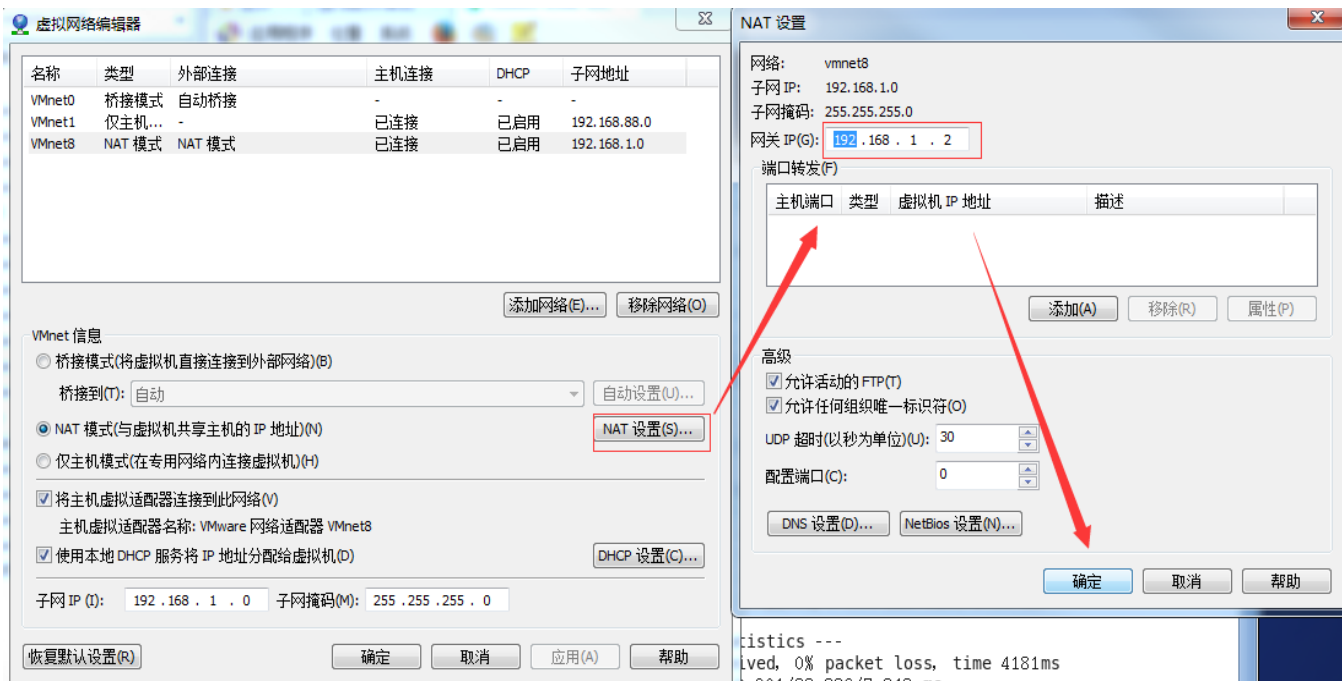
● 查看网络IP和网关

修改ip地址



● 查看网络IP和网关

查看网关



虚拟网络编辑器

名称	类型	外部连接	主机连接	DHCP	子网地址
VMnet0	桥接模式	自动桥接	-	-	-
VMnet1	仅主机...	-	已连接	已启用	192.168.88.0
VMnet8	NAT 模式	NAT 模式	已连接	已启用	192.168.1.0

添加网络(E)... 移除网络(O)

VMnet 信息

☐ 桥接模式(将虚拟机直接连接到外部网络)(B)

桥接到(T): 自动 自动设置(U)...

☒ NAT 模式(与虚拟机共享主机的 IP 地址)(N) NAT 设置(S)...

☐ 仅主机模式(在专用网络内连接虚拟机)(H)

☒ 将主机虚拟适配器连接到此网络(V)

主机虚拟适配器名称: VMware 网络适配器 VMnet8

☒ 使用本地 DHCP 服务将 IP 地址分配给虚拟机(D) DHCP 设置(C)...

子网 IP (I): 192.168.1.0 子网掩码(M): 255.255.255.0

恢复默认设置(R) 确定 取消 应用(A) 帮助

NAT 设置

网络: vmnet8

子网 IP: 192.168.1.0

子网掩码: 255.255.255.0

网关 IP(G): 192.168.1.2

端口转发(F)

主机端口	类型	虚拟机 IP 地址	描述
------	----	-----------	----

添加(A) 移除(R) 属性(P)

高级

☒ 允许活动的 FTP(T)

☒ 允许任何组织唯一标识符(O)

UDP 超时(以秒为单位)(U): 30

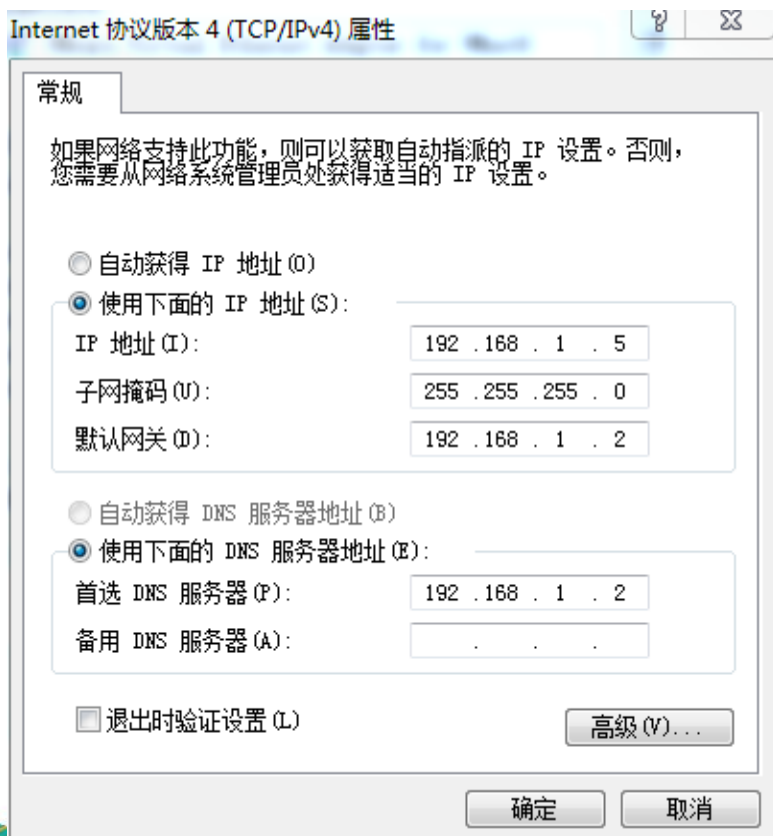
配置端口(C): 0

DNS 设置(D)... NetBios 设置(N)...

确定 取消 帮助

● 查看网络IP和网关

查看windows环境的中VMnet8网络配置 (ipconfig指令)





- **ping** 测试主机之间网络连通性

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

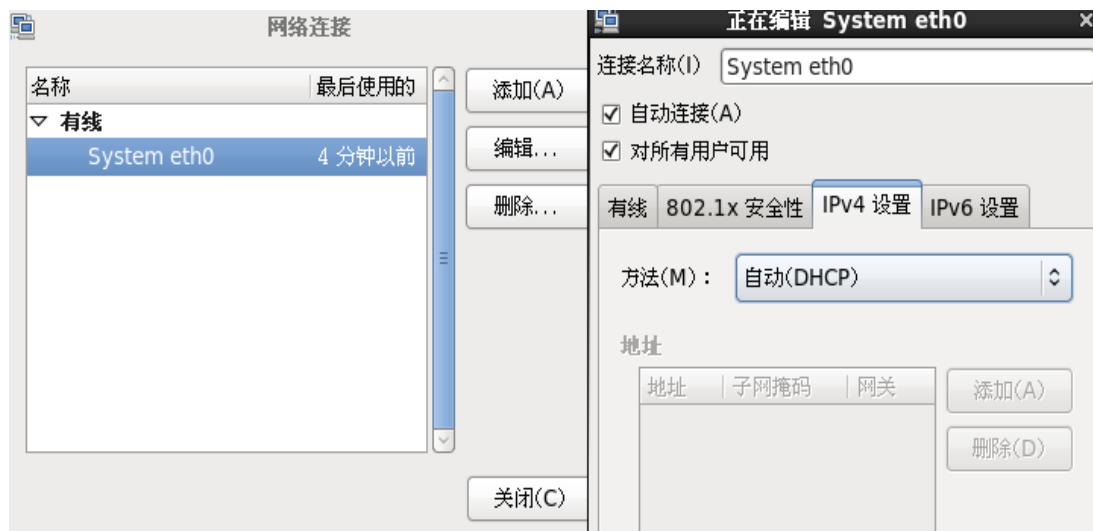
测试当前服务器是否可以连接百度

[root@hadoop100 桌面]# ping www.baidu.com

● linux网络环境配置

第一种方法(自动获取):

说明: 登陆后, 通过界面的来设置自动获取ip



特点: 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。

- **linux网络环境配置**

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



- **linux网络环境配置**

第二种方法(指定固定的ip):

- ifcfg-eth0文件说明

```
DEVICE=eth0          #接口名（设备,网卡）
HWADDR=00:0C:2x:6x:0x:xx #MAC地址
TYPE=Ethernet        #网络类型（通常是Ethernet）
UUID=926a57ba-92c6-4231-bacb-f27e5e6a9f44 #随机id
#系统启动的时候网络接口是否有效（yes/no）
ONBOOT=yes
# 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）
BOOTPROTO=static
#IP地址
IPADDR=192.168.184.130
#网关
GATEWAY=192.168.184.2
#域名解析器
DNS1=192.168.184.2
```

- 重启网络服务或者重启系统生效

service network restart 、 reboot

Linux实操篇

进程管理(重点)

● 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。直到关机才结束。

● 显示系统执行的进程

基本介绍

ps命令是用来查看目前系统中，有哪些正在执行，以及它们执行的状况。可以不加任何参数。

ps 显示的信息选项：

字段	说明
PID	进程识别号
TTY	终端机号
TIME	此进程所消 CPU 时间
CMD	正在执行的命令或进程名

ps -a: 显示当前终端的所有进程信息

ps -u: 以用户的格式显示进程信息

ps -x: 显示后台进程运行的参数

● 显示系统执行的进程

ps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

- **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

● 显示系统执行的进程

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



● 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

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

常用选项:

-p : 显示进程的PID

-u : 显示进程的所属用户

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

● 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】

service管理指令:

service 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。

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。
- 3) 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。



- 服务(service)管理

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



• 服务(service)管理

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：





- 服务(service)管理

一个思考题

如果不小心将默认的运行级别设置成 0 或者 7，怎么处理？
进入单用户模式，修改成正常的即可。。。

- 服务(service)管理

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`

2) `chkconfig 服务名 --list`

3) `chkconfig --level 5 服务名 on/off`



- 服务(service)管理

chkconfig指令

- **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
- 2) 案例2：请查看sshd服务的运行状态
- 3) 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？
- 4) 案例4：当运行级别为5时，关闭防火墙。
- 5) 案例5：在所有运行级别下，关闭防火墙
- 6) 案例6：在所有运行级别下，开启防火墙

- **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:

选项	功能
-d 秒数	指定 <code>top</code> 命令每隔几秒更新。默认是3秒在 <code>top</code> 命令的交互模式当中可以执行的命令:
-i	使 <code>top</code> 不显示任何闲置或者僵死进程。
-p	通过指定监控进程ID来仅仅监控某个进程的状态。

● 动态监控进程

交互操作说明：

操作	功能
P	以CPU使用率排序，默认就是此项
M	以内存的使用率排序
N	以PID排序
q	退出top

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):



● 监控网络状态

查看系统网络情况netstat

- 基本语法
netstat [选项]
- 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
 - 请查看服务名为 sshd 的服务的信息。

检测主机连接命令ping:

是一种网络检测检测工具，它主要是用检测远程主机是否正常，或是两部主机间的介质是否为断、网线是否脱落或网卡故障。

如: ping 对方ip地址

Linux实操篇

RPM 与 YUM

- rpm包的管理

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。



- rpm包的管理

rpm包的简单查询指令:

查询已安装的rpm列表 `rpm -qa|grep xx`

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。。



● rpm包的管理

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

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

- rpm包的管理

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

- 细节讨论

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

2) 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`

- rpm包的管理

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) 演示卸载和安装firefox浏览器

- yum

介绍:

Yum 是一个[Shell](#)前端软件包管理器。基于[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。



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:

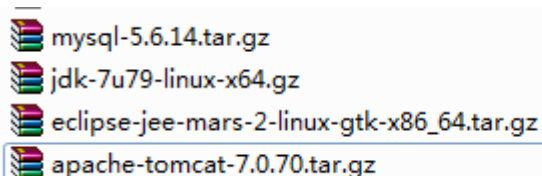
案例：请使用yum的方式来安装firefox

Linux之JavaEE定制篇

搭建JavaEE环境

● 概述

如果需要在Linux下进行JavaEE的开发，我们需要安装如下软件：



- mysql-5.6.14.tar.gz
- jdk-7u79-linux-x64.gz
- eclipse-jee-mars-2-linux-gtk-x86_64.tar.gz
- apache-tomcat-7.0.70.tar.gz

● 安装JDK

安装步骤

- 0) 先将软件通过xftp5 上传到 /opt 下
- 1) 解压缩到 /opt
- 2) 配置环境变量的配置文件vim /etc/profile
- 3) JAVA_HOME=/opt/jdk1.7.0_79
- 4) PATH=/opt/jdk1.7.0_79/bin:\$PATH
- 5) export JAVA_HOME PATH

测试是否安装成功

编写一个简单的Hello.java 输出"hello,world!"

● tomcat的安装

步骤：

- 1) 解压缩到/opt
- 2) 启动tomcat `./startup.sh`
- 3) 开放端口 `vim /etc/sysconfig/iptables`

测试是否安装成功：

在windows、Linux 下 访问 <http://linuxip:8080>

Apache Tomcat/7.0.70

If you're seeing this, you've success



Recommended Reading:

[Security Considerations HOW-TO](#)

[Manager Application HOW-TO](#)

[Clustering/Session Replication HOW-TO](#)

● Eclipse的安装

步骤：

- 1) 解压缩到/opt
- 2) 启动eclipse，配置jre和server
- 3) 编写Hello world 程序并测试成功！
- 4) 编写jsp 页面,并测试成功！



hello,world!!

● mysql5.6的安装

安装的步骤和文档

[说明: 因为mysql安装时间很长, 所以在授课时, 可以考虑最先安装mysql]
相关的安装软件在课件

注意: 先删除一下 **Mysql 相关的软件**..



尚硅谷centos6.8下安装mysql.do

Linux之大数据定制篇

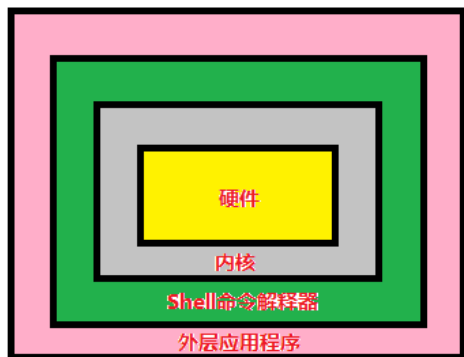
大数据定制篇-Shell编程

● 为什么要学习Shell编程

- 1) Linux运维工程师在进行服务器集群管理时，需要编写Shell程序来进行服务器管理。
- 2) 对于JavaEE和Python程序员来说，工作的需要，你的老大会要求你编写一些Shell脚本进行程序或者是服务器的维护，比如编写一个定时备份数据库的脚本。
- 3) 对于大数据程序员来说，需要编写Shell程序来管理集群。

● Shell是什么

Shell是一个命令行解释器，它为用户提供了一个向Linux内核发送请求以便运行程序的界面系统级程序，用户可以用Shell来启动、挂起、停止甚至是编写一些程序。



● Shell脚本的执行方式

脚本格式要求

- 1) 脚本以#!/bin/bash开头
- 2) 脚本需要有可执行权限

编写第一个Shell脚本

- 需求说明
创建一个Shell脚本，输出hello world!

脚本的常用执行方式

- 方式1(输入脚本的绝对路径或相对路径)
 - 1) 首先要赋予helloworld.sh 脚本的+x权限
 - 2) 执行脚本
- 方式2(sh+脚本)
说明：不用赋予脚本+x权限，直接执行即可。

● Shell的变量

Shell的变量的介绍

- 1) Linux Shell中的变量分为，系统变量和用户自定义变量。
- 2) 系统变量：\$HOME、\$PWD、\$SHELL、\$USER等等
比如：echo \$HOME 等等..
- 3) 显示当前shell中所有变量：set

shell变量的定义

• 基本语法

- 1)定义变量：变量=值
- 2)撤销变量：unset 变量
- 3) 声明静态变量：readonly变量，注意：不能unset

• 快速入门

案例1：定义变量A

案例2：撤销变量A

案例3：声明静态的变量B=2，不能unset

案例4：可把变量提升为全局环境变量，可供其他shell程序使用

● Shell的变量

shell变量的定义

• 定义变量的规则

- 1) 变量名称可以由字母、数字和下划线组成，但是不能以数字开头。
- 2) 等号两侧不能有空格
- 3) 变量名称一般习惯为大写

• 将命令的返回值赋给变量

- 1) `A=`ls -la`` 反引号，运行里面的命令，并把结果返回给变量A
- 2) `A=$(ls -la)` 等价于反引号



● 设置环境变量

基本语法

- 1) `export 变量名=变量值` （功能描述：将shell变量输出为环境变量）
- 2) `source 配置文件` （功能描述：让修改后的配置信息立即生效）
- 3) `echo $变量名` （功能描述：查询环境变量的值）

快速入门

- 1) 在`/etc/profile`文件中定义`TOMCAT_HOME`环境变量
- 2) 查看环境变量`TOMCAT_HOME`的值
- 3) 在另外一个shell程序中使用 `TOMCAT_HOME`

注意：在输出`JAVA_HOME` 环境变量前，需要让其生效

`source /etc/profile`

● 位置参数变量

介绍

当我们执行一个shell脚本时，如果希望获取到命令行的参数信息，就可以使用到位置参数变量

比如：`./myshell.sh 100 200`，这个就是一个执行shell的命令，可以在myshell 脚本中获取到参数信息

基本语法

`$n`（功能描述：`n`为数字，`$0`代表命令本身，`$1-$9`代表第一到第九个参数，十以上的参数，十以上的参数需要用大括号包含，如`${10}`）

`$*`（功能描述：这个变量代表命令行中所有的参数，**`$*`把所有的参数看成一个整体**）

`$@`（功能描述：这个变量也代表命令行中所有的参数，**不过`$@`把每个参数区分对待**）

`$#`（功能描述：这个变量代表命令行中所有参数的个数）

位置参数变量

● 应用实例

案例：编写一个shell脚本 `positionPara.sh`，在脚本中获取到命令行的各个参数信息。



● 预定义变量

基本介绍

就是shell设计者事先已经定义好的变量，可以直接在shell脚本中使用

基本语法

\$\$ (功能描述：当前进程的进程号 (PID))

\$! (功能描述：后台运行的最后一个进程的进程号 (PID))

\$? (功能描述：最后一次执行的命令的返回状态。如果这个变量的值为0，证明上一个命令正确执行；如果这个变量的值为非0（具体是哪个数，由命令自己来决定），则证明上一个命令执行不正确了。)

应用实例

在一个shell脚本中简单使用一下预定义变量

● 运算符

基本介绍

学习如何在shell中进行各种运算操作。

基本语法

1) “\$((运算式))”或“\${运算式}”

2) `expr m + n`

注意`expr`运算符间要有空格

3) `expr m - n`

4) `expr *, /, %` 乘, 除, 取余

• 应用实例

案例1: 计算 $(2+3) \times 4$ 的值

案例2: 请求出命令行的两个参数[整数]的和



- 条件判断

判断语句

- 基本语法

[condition] (注意condition前后要有空格)

#非空返回true，可使用\$?验证 (0为true，>1为false)

- 应用实例

[atguigu]

[]

[condition] && echo OK || echo notok

返回true

返回false

条件满足，执行后面的语句



● 条件判断

判断语句

• 常用判断条件

1) 两个整数的比较

= 字符串比较

-lt 小于

-le 小于等于

-eq 等于

-gt 大于

-ge 大于等于

-ne 不等于

2) 按照文件权限进行判断

-r 有读的权限

-w 有写的权限

-x 有执行的权限

3) 按照文件类型进行判断

-f 文件存在并且是一个常规的文件

-e 文件存在

-d 文件存在并且是一个目录

4) 应用实例

案例1: "ok"是否等于"ok"

判断语句:

案例2: 23是否大于等于22

判断语句:

案例3: /root/shell/aaa.txt 目录中的文件是否存在

判断语句:

● 流程控制

if 判断

• 基本语法

```
if [ 条件判断式 ];then  
    程序
```

```
fi
```

或者

```
if [ 条件判断式 ]  
then  
    程序  
elif [条件判断式]  
then  
    程序
```

```
fi
```

注意事项：（1）[条件判断式]，中括号和条件判断式之间必须有空格（2）推荐使用第二种方式

• 应用实例

案例：请编写一个shell程序，如果输入的参数，大于等于60，则输出 "及格了"，如果小于60,则输出 "不及格"

● 流程控制

case语句

• 基本语法

case \$变量名 in

"值1")

如果变量的值等于值1，则执行程序1

;;

"值2")

如果变量的值等于值2，则执行程序2

;;

...省略其他分支...

*)

如果变量的值都不是以上的值，则执行此程序

;;

esac

• 应用实例

案例1：当命令行参数是 1 时，输出 "周一", 是2 时，就输出"周二"， 其它情况输出 "other"

- 流程控制

for循环

- 基本语法1

for 变量 in 值1 值2 值3...

do

程序

done

- 应用实例

案例1：打印命令行输入的参数

- 基本语法2

for ((初始值;循环控制条件;变量变化))

do

程序

done

- 应用实例

案例1：从1加到100的值输出显示 [这里可以看出\$* 和 \$@ 的区别]

- 流程控制

while循环

- 基本语法1

while [条件判断式]

do

程序

done

- 应用实例

案例1：从命令行输入一个数n，统计从 $1+..+n$ 的值是多少？

● read读取控制台输入

基本语法

`read(选项)(参数)`

选项:

-p: 指定读取值时的提示符;

-t: 指定读取值时等待的时间（秒），如果没有在指定的时间内输入，就不再等待了。。

参数

变量: 指定读取值的变量名

应用实例

案例1: 读取控制台输入一个num值

案例2: 读取控制台输入一个num值，在10秒内输入。



● 函数

函数介绍

shell编程和其它编程语言一样，有系统函数，也可以自定义函数。系统函数中，我们这里就介绍两个。

系统函数

- **basename**基本语法

功能：返回完整路径最后 / 的部分，常用于获取文件名

`basename [pathname] [suffix]`

`basename [string] [suffix]` （功能描述：`basename`命令会删掉所有的前缀包括最后一个（`'/'`）字符，然后将字符串显示出来。

选项：

`suffix`为后缀，如果`suffix`被指定了，`basename`会将`pathname`或`string`中的`suffix`去掉。

- 应用实例

案例1：请返回 `/home/aaa/test.txt` 的 `"test.txt"` 部分

- 函数

系统函数

- **dirname**基本语法

功能：返回完整路径最后 / 的**前面的部分**，常用于返回路径部分

dirname 文件绝对路径 （功能描述：从给定的包含绝对路径的文件名中去除文件名（非目录的部分），然后返回剩下的路径（目录的部分））

- 应用实例

案例1：请返回 /home/aaa/test.txt 的 /home/aaa

- 函数

自定义函数

- 基本语法

```
[ function ] funname[()]\n{\n    Action;\n    [return int;]\n}
```

调用直接写函数名: funname [值]

- 应用实例

案例1: 计算输入两个参数的和, getSum

● Shell编程综合案例

需求分析

- 1) 每天凌晨 2:10 备份 数据库 atguiguDB 到 /data/backup/db
- 2) 备份开始和备份结束能够给出相应的提示信息
- 3) 备份后的文件要求以备份时间为文件名，并打包成 .tar.gz 的形式，比如：
2018-03-12_230201.tar.gz
- 4) 在备份的同时，检查是否有10天前备份的数据库文件，如果有就将其删除。

```
[root@hsp db]#  
[root@hsp db]# ll  
总用量 8  
-rw-r--r--. 1 root root 1032 3月 12 22:53 2018-03-12_225338.tar.gz  
-rw-r--r--. 1 root root 187 3月 12 23:02 2018-03-12_230201.tar.gz  
[root@hsp db]#
```

Linux之Python定制篇

Python定制篇-专业的Python开发平台Ubuntu

● Ubuntu介绍



Ubuntu（友帮拓、优般图、乌班图）是一个以桌面应用为主的开源GNU/Linux操作系统，Ubuntu 是基于 GNU/Linux，支持x86、amd64（即x64）和ppc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/>

- 安装Ubuntu

安装的步骤



尚硅谷Python安装Ubunt

● 安装Ubuntu

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root 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，就可以切换到root了。提示符`$`代表一般用户，提示符`#`代表root用户。
- 3) 输入 `exit` 命令，退出root并返回一般用户
- 4) 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]。

```
atguigu@ubuntu:~$ python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license()" for more information
>>> print "hello,world"
hello,world
>>> exit()
```

```
atguigu@ubuntu:~$ python3
Python 3.5.2 (default, Nov 17 2016, 17:05:23)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license()" for more information
>>> print("hello,world!")
hello,world!
>>> exit
Use exit() or Ctrl-D (i.e. EOF) to exit
```

在Ubuntu下开发一个Python程序

- 1) vim hello.py [编写hello.py]
- 2) python3 hello.py [运行hello.py]

Linux之Python定制篇

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。

ubuntu 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

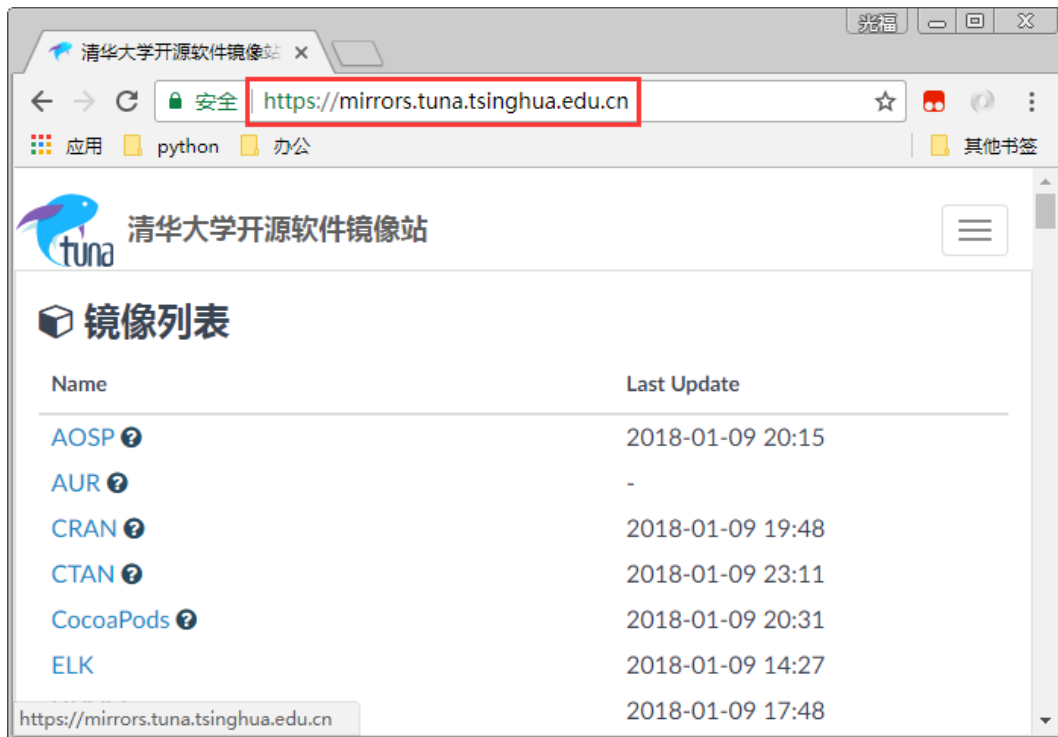
● 更新Ubuntu软件下载地址

原理介绍

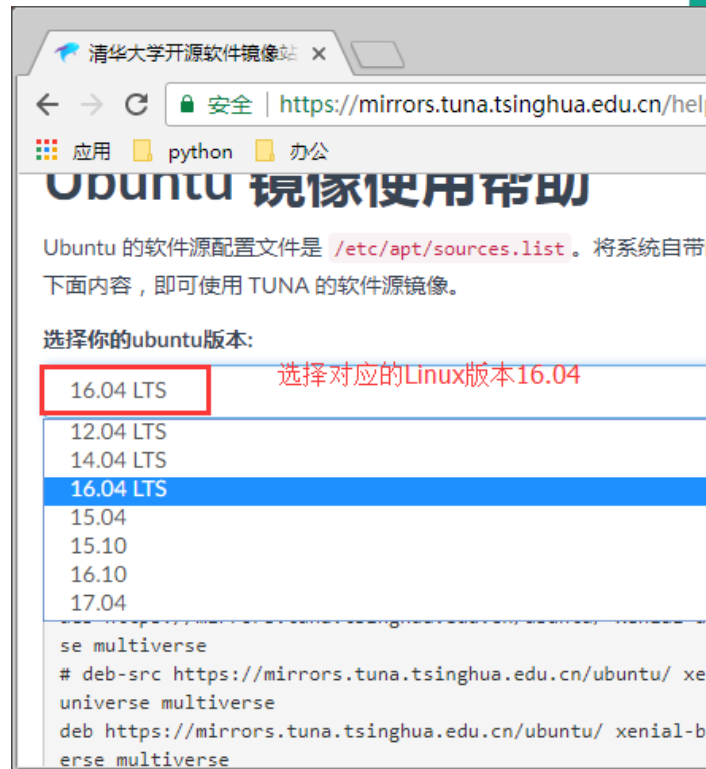
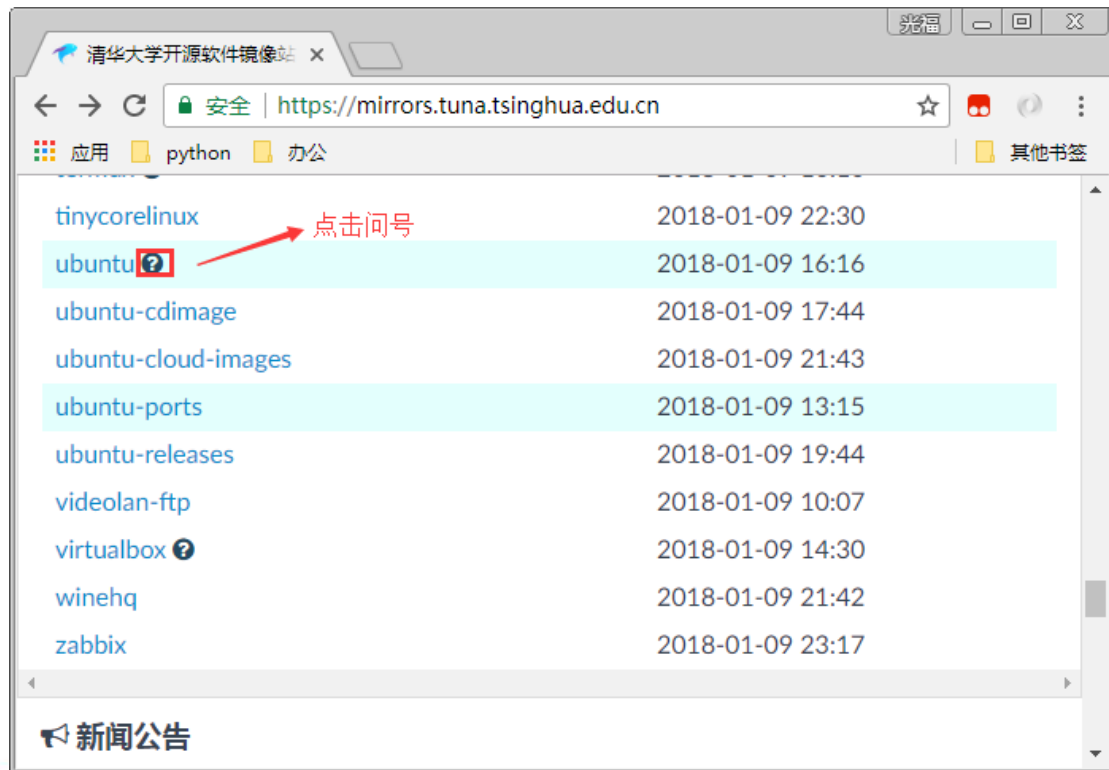
(画出示意图)

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



● 更新Ubuntu软件下载地址 寻找国内镜像源



● 更新Ubuntu软件下载地址

备份Ubuntu默认的源地址

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

```
atguigu@ubuntu:/etc/apt$ pwd
/etc/apt
atguigu@ubuntu:/etc/apt$ ls
apt.conf.d      sources.list      trusted.gpg
preferences.d   sources.list.d    trusted.gpg~
sources.bak     sources.list.save trusted.gpg.d
atguigu@ubuntu:/etc/apt$ cat sources.list
# deb cdrom:[Ubuntu 16.04.2 LTS _Xenial Xerus_ - Release amd64 (20170
215.2)]/ xenial main restricted

# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrad
e to
# newer versions of the distribution.
deb http://us.archive.ubuntu.com/ubuntu/ xenial main restricted
# deb-src http://us.archive.ubuntu.com/ubuntu/ xenial main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://us.archive.ubuntu.com/ubuntu/ xenial-updates main restrict
ed
# deb-src http://us.archive.ubuntu.com/ubuntu/ xenial-updates main re
stricted

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the
```

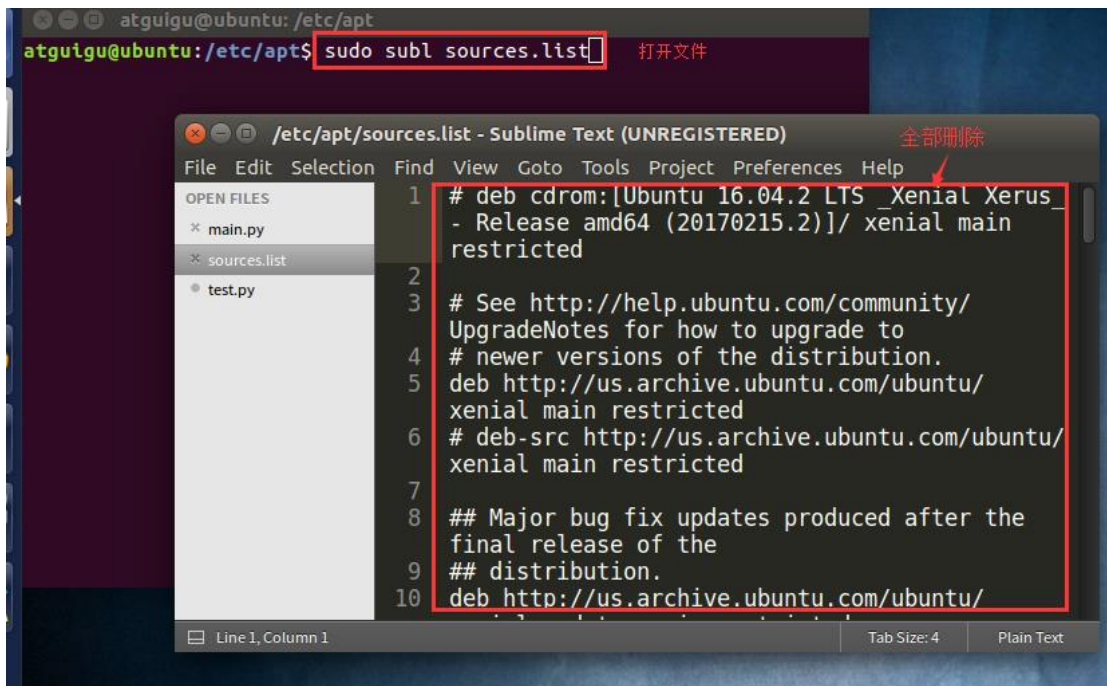
查看地址，国外的地址慢，换成清华大学的，但是先备份，拷贝一份

```
atguigu@ubuntu:/etc/apt$ ls
apt.conf.d      sources.list      trusted.gpg
preferences.d   sources.list.d    trusted.gpg~
sources.bak     sources.list.save trusted.gpg.d
atguigu@ubuntu:/etc/apt$ cp sources.list sources.
cp: 无法创建普通文件'sources.list.backup': 权限不
atguigu@ubuntu:/etc/apt$ sudo cp sources.list sou
atguigu@ubuntu:/etc/apt$ ls
apt.conf.d      sources.list      sources.list.
preferences.d   sources.list.backup trusted.gpg
sources.bak     sources.list.d    trusted.gpg~
atguigu@ubuntu:/etc/apt$
```

备份成功，现在可以修改so

- 更新Ubuntu软件下载地址

更新源服务器列表



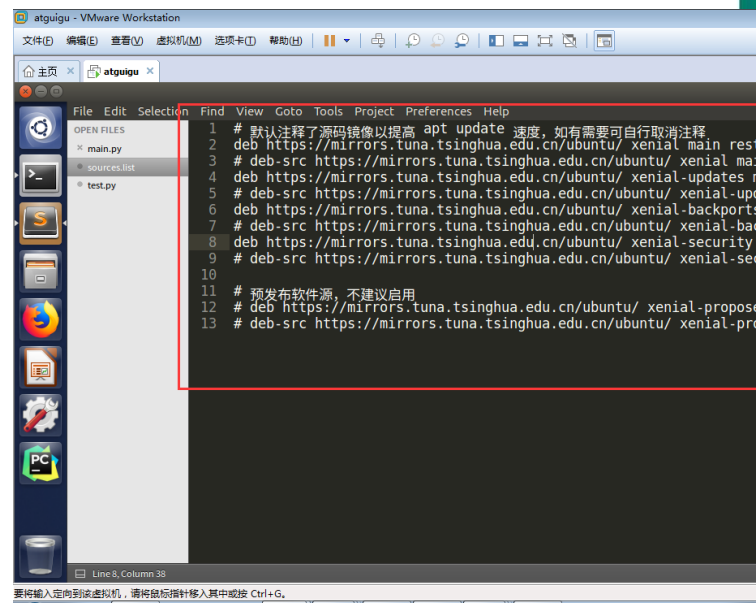
The image shows a terminal window and a Sublime Text editor. In the terminal, the command `sudo subl sources.list` is entered, with a red box highlighting the command and a red arrow pointing to the terminal output. The Sublime Text editor displays the contents of the `/etc/apt/sources.list` file, which is highlighted with a red box. The file contains the following text:

```
1 # deb cdrom:[Ubuntu 16.04.2 LTS _Xenial Xerus_  
- Release amd64 (20170215.2)]/ xenial main  
restricted  
2  
3 # See http://help.ubuntu.com/community/  
UpgradeNotes for how to upgrade to  
4 # newer versions of the distribution.  
5 deb http://us.archive.ubuntu.com/ubuntu/  
xenial main restricted  
6 # deb-src http://us.archive.ubuntu.com/ubuntu/  
xenial main restricted  
7  
8 ## Major bug fix updates produced after the  
final release of the  
9 ## distribution.  
10 deb http://us.archive.ubuntu.com/ubuntu/
```

A red arrow points to the top right corner of the Sublime Text window, labeled "全部删除" (Delete All).



● 更新Ubuntu软件下载地址 更新源服务器列表





● 更新Ubuntu软件下载地址

更新源

更新源地址: `sudo apt-get update`

```
atguigu@ubuntu:~$ pwd
/home/atguigu
atguigu@ubuntu:~$ sudo apt-get update 更新源
获取:1 http://archive.ubuntu-kylin.com/ubuntu-kylin xenial InRelease [18.1 kB]
命中:2 https://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial InRelease
获取:3 https://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates InRelease [102 kB]
获取:4 https://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-backports InRelease [102 kB]
获取:5 https://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-security InRelease [102 kB]
忽略:6 http://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/3.4 InRelease
命中:7 http://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/3.4 Release
获取:8 https://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/main amd64 Packages [689 kB]
获取:9 https://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/main i386 Packages [646 kB]
获取:10 https://mirrors.tuna.tsinghua.edu.cn/ubuntu xenial-updates/main amd64 DEP-11 Metadata [307 kB]
```



- **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



- 使用ssh远程登录Ubuntu

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 - 切换到root用户

```
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

atguigu@ubuntu:~$ ll
总用量 132
drwxr-xr-x 17 atguigu atguigu 4096 3月 11 07:41 ./
drwxr-xr-x  3 root    root    4096 3月 10 22:13 ../
-rw-----  1 atguigu atguigu  776 3月 11 08:23 .bash_history
-rw-r--r--  1 atguigu atguigu  220 3月 10 22:13 .bash_logout
-rw-r--r--  1 atguigu atguigu 3771 3月 10 22:13 .bashrc
drwx----- 12 atguigu atguigu 4096 3月 11 08:25 .cache/
```



- 使用ssh远程登录Ubuntu

从linux系统客户机远程登陆linux系统服务机

- 基本语法：

ssh 用户名@IP

例如：ssh atguigu@192.168.188.130

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



谢谢！ 欢迎收看！