

第二讲

Linux 操作系统 与程序开发环境

主要内容

- ❑ Linux 介绍
- ❑ Linux 系统的安装
- ❑ Linux 基本使用与管理
- ❑ Linux 基本命令
- ❑ Linux 程序开发环境

操作系统

❑ 计算机=硬件+软件

❑ 计算机软件大致可分为两类

- 系统软件

管理计算机资源，并为应用软件提供一个统一的平台

- 应用软件

在系统软件的基础上实现用户所需要的各种功能

❑ 操作系统（Operating System，简称OS）是最基本的系统软件，负责对电脑硬件直接控制及管理，功能一般包括处理器管理、存储管理、文件管理、设备管理和作业管理等。

常见操作系统

❑ Unix

- 1969 在 AT&T Bell 实验室诞生，主要用于服务器领域，大部分重要网络环节都是 Unix 构造
- 版本众多：AIX (IBM)、HP-UX (HP)、Solaris(SUN) ...
- UNIX 不是免费的，且不适用个人PC电脑

❑ MS-DOS

- 1981 年由微软公司为 IBM 个人电脑开发的
- 1985 — 1995 占据微机操作系统的统治地位

❑ Mac OS

- 1984 年由苹果公司推出
- 是首个在商用领域成功的图形用户界面
- 只能运行在苹果公司自己的电脑上

常见操作系统

❑ Windows

- 由微软公司于 1985 年发行
- 现已获得个人电脑操作系统的垄断地位

❑ Linux

- 1991 年发布，免费使用和自由传播的类 Unix 操作系统
- 支持多种硬件平台，包括个人电脑和超级并行计算机
- 有良好的用户界面，同时提供字符界面和图形界面
- 版本众多：Fedora、Ubuntu、...

❑ 其它操作系统：BSD、RTOS、Palm OS、...

Linux 起源

□ MINIX

1987 年，荷兰计算机科学家 A. Tanenbaum 专门写的简化的类 UNIX 系统 (mini-UNIX)，给入门者学习使用

□ Linux

由芬兰赫尔辛基大学的学生 Linus Torvalds 发起，目的是设计一个替代 Minix 的操作系统，可用于个人计算机，并且具有 Unix 操作系统的全部功能

Linux系统组成

❑ 内核

运行程序和管理各种硬件设备的核心程序

❑ Shell

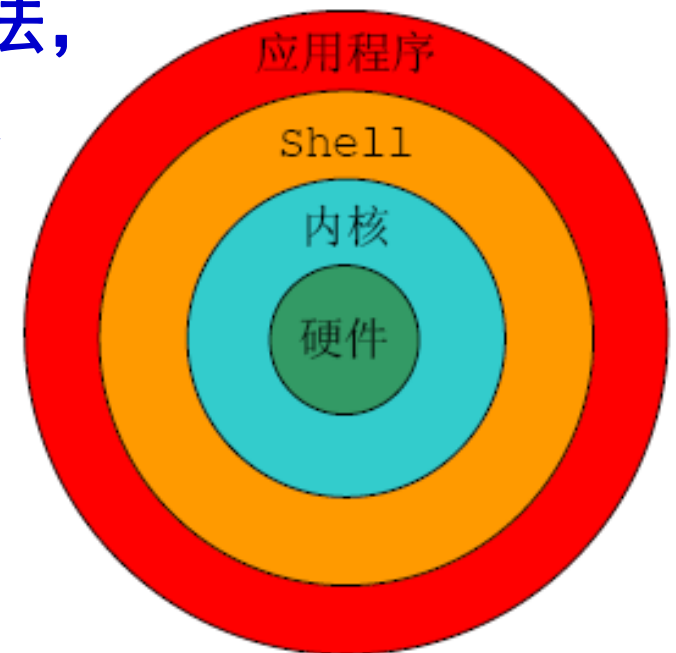
命令语言解释器，是用户与内核进行交互操作的接口

❑ 文件系统

文件存放在磁盘等存储设备上的组织方法，Linux 支持多种文件系统，如 ext3、fat、ntfs、iso9660、nfs、smb 等

❑ 应用软件

Linux 发行版都有一整套的应用程序集



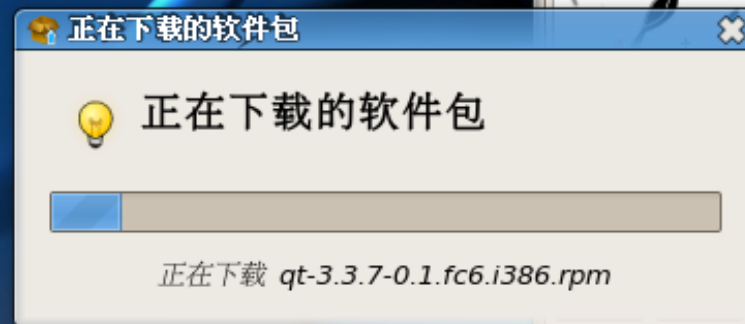
常见 Linux 发行版



- Red Hat 公司创建于1994年，是全球最大的开源技术厂家，其产品也是全世界应用最广泛的 Linux
- Red Hat 是目前唯一公开上市的开放源代码厂商
- 主要产品
 - 个人桌面版：Fedora (最新版为 Fedora 11)
 - 服务器免费版：CentOS
 - 企业版：RHEL

Fedora 截图





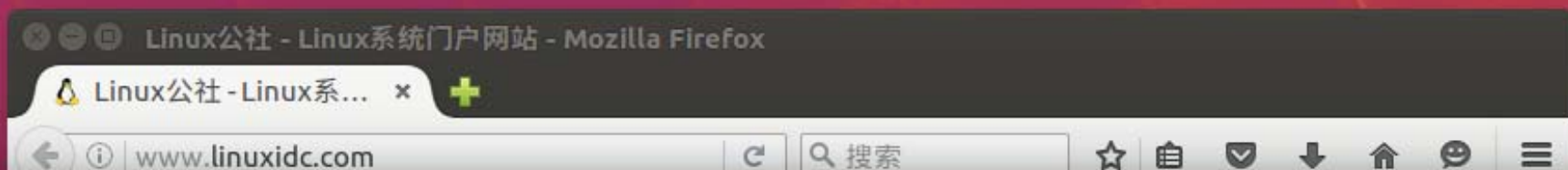
ubuntu



- Ubuntu 是一个相对较新的发行版，每六个月更新一次，被誉为对硬件支持最好最全面的Linux发行版之一
- 旨在创建一个可以为桌面和服务端提供一个最新且一贯的 Linux 系统
- Ubuntu 的版本号由发布的年份和月份组成，首次发布是在2004年10月，版本为4.10。当前版本于2018年4月发布，因此版本号为18.04
- 入门书《Ubuntu完美应用(第三版)》，关键在学以致用



Linuxidc.com



全部(A)

已安装(I)

更新(U)

搜索

编辑推荐

SpeedCrunch

MyPaint

Code::Blocks IDE

Artha

Geany

推荐的游戏

俄罗斯方块

Frozen-Bubble

消色块

数独

已安装

0 A.D.

软件分类

其它发行版



<http://www.debian.org>



<http://www.mandriva.com>



<http://http://www.novell.com/linux>



[http:// www.redflag-linux.com](http://www.redflag-linux.com)



<http://www.deepin.org/>

更多参见: <https://www.linux.org/pages/download/>

主要内容

- ❑ Linux 介绍
- ❑ Linux 系统的安装
- ❑ Linux 基本使用与管理
- ❑ Linux 基本命令
- ❑ Linux 程序开发环境

Fedora 安装

❑ 安装准备

- 划分一定的空间（5G 以上，建议 10G 左右）
- 获取安装盘（购买或下载后刻盘）

❑ 安装方式

- DVD安装、网络安装、本地安装
- 安装模式：图形模式和文本模式

❑ 安装过程 *step by step*

- 使用中文安装界面
- 分区时一定要手工分区
- 硬盘足够大时可安装所有软件包

主要内容

- ❑ Linux 介绍
- ❑ Linux 系统的安装
- ❑ Linux 基本使用与管理
- ❑ Linux 基本命令
- ❑ Linux 程序开发环境

Linux 使用基础

- **命令 (Command)** 给计算机的指令，通常使用键盘输入
- **命令行 (Command line)** 输入命令的地方
- **shell**: 是一种命令解释器，它提供了用户和操作系统之间的交互接口。当用户在命令行输入命令后，shell 进行解释，然后送往操作系统（内核）执行。
shell 可以执行 Linux 的系统**内部命令**，也可以执行**应用程序**。你还可以利用 shell **编程**，执行复杂的命令程序。
- **root**: 根用户、超级用户、管理员，它对整个系统有完全的访问权。只有超级用户才能更改系统的设置等。

当使用超级用户登录后，进行相关操作时需非常谨慎，否则可能会对系统或用户文件造成不可挽回的损失

Linux 使用基础

□ Linux 字符操作环境

- 终端和虚拟终端
- 用户可以通过终端访问系统资源
- 在图形界面下打开虚拟终端
- 远程登录（SSH Client）
- 退出终端： `exit`

注：Linux 区分大小写！

Linux 使用基础

- 用户登录后，将会进入一个系统指定的专属目录，即用户的主目录，该目录名通常为用户的登录帐号

例：用户 `student` 的主目录为 `/home/student/`

- 在创建用户时，系统管理员会给每个用户建立一个主目录，通常在 `/home/` 目录下
- 用户对自己主目录的文件拥有所有权，可以在自己的主目录下进行相关操作
- 每个用户名对应一个用户 ID 号（一个数字）
- 每个用户都被分配到一个指定的组 (group) 中

主要内容

- ❑ Linux 介绍
- ❑ Linux 系统的安装
- ❑ Linux 基本使用与管理
- ❑ Linux 基本命令
- ❑ Linux 程序开发环境

命令一般格式

command [选项] [文件或目录列表]

- 其中 选项 通常以 “-” 开始，如 **ls -l**
- 多个选项可用一个 “-”，如 **ls -l -a** 与 **ls -la** 相同
- 所有的命令从标准输入接受输入，输出结果显示在标准输出，而错误信息则显示在标准错误输出设备
- 可使用重定向功能对这些设备进行重定向
- 命令在正常执行结果后返回一个 0 值，如果命令出错，则返回一个非零值 (在shell中可用变量 **\$?** 查看)

更改口令

passwd

- ◆ 直接在命令行中输入 `passwd`，然后根据提示，先输入原来的口令，回车后，再输入新口令，并确认一次

提醒： 输入密码时，屏幕上不会有任何输出或提示！

- ◆ 为确保高安全性，Linux 对口令有一定的要求，而且通常需要在口令中加入一些特殊符号，如加号、减号等

yppasswd

- ◆ 修改 NIS 口令，网络用户需要使用该命令来修改密码

注： 在机房更改口令时需用 `yppasswd` 命令

获取帮助

man 命令

- 获取相关命令的帮助信息

info 命令

- 获取相关命令的更多信息

**举
例**

man **ls**

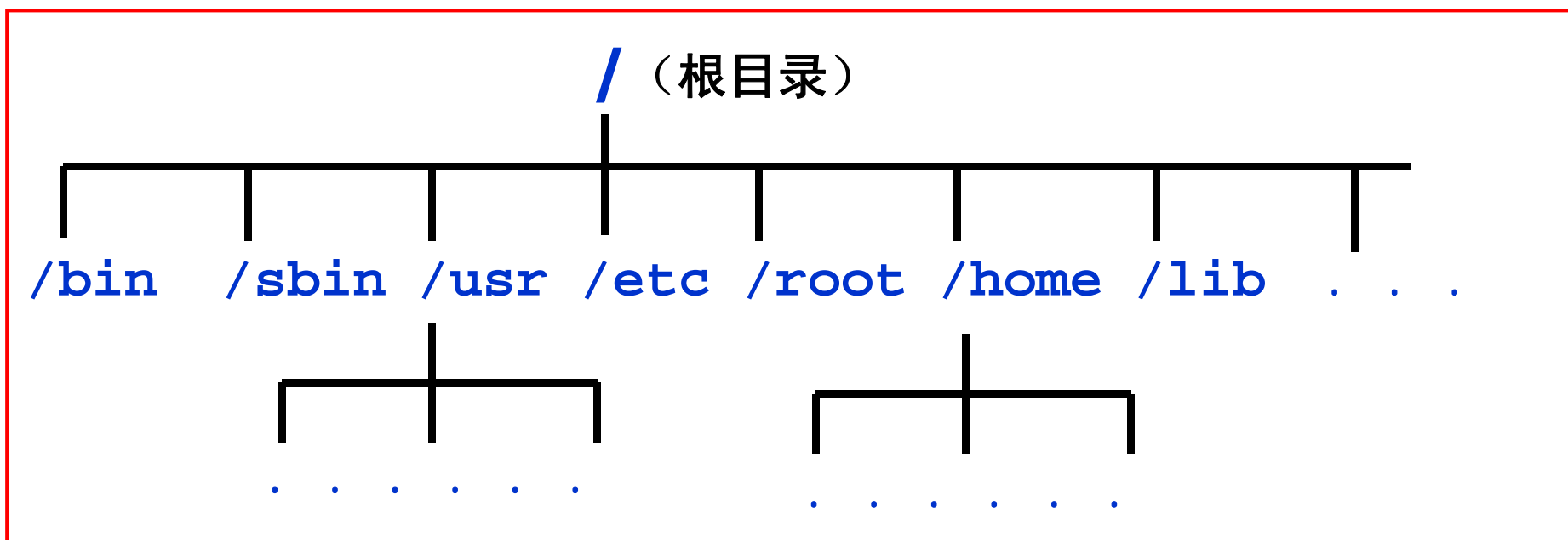
info **ls**

注：退出 **man** 或 **info** 按 **q** 即可

文件系统结构

❑ Linux 目录结构

- ◆ **目录树**结构：文件系统从一个**根目录**开始，根目录下可以有任意多个文件和子目录，子目录中又可以有任意多个文件和子目录
- ◆ Linux 的这种文件系统结构使得一个目录和它包含的文件/子目录之间形成一个**父子关系**

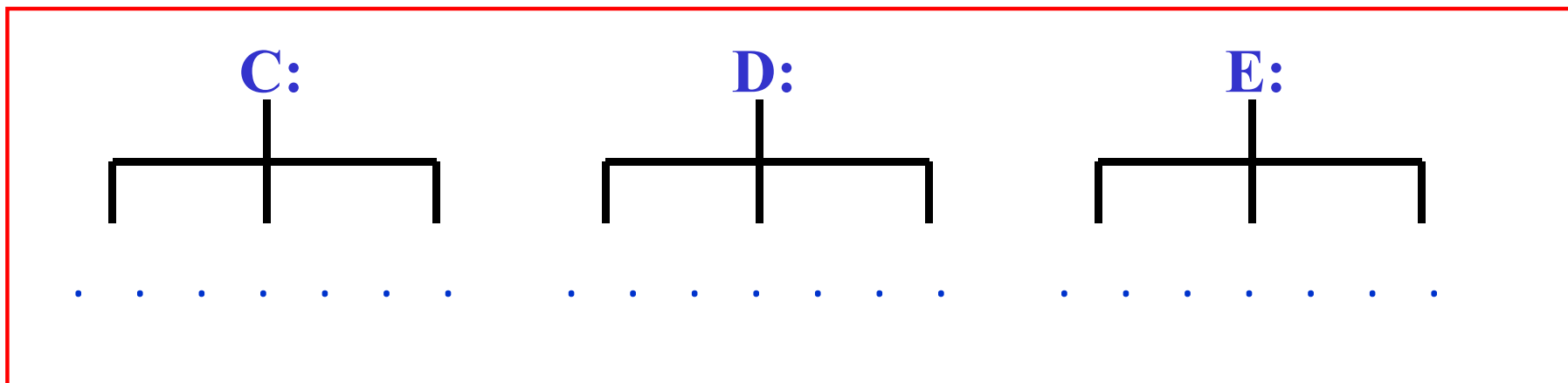


Linux 文件结构是以根目录 “ / ” 为起点，向下展开

文件系统结构

❑ Linux文件系统的结构与Windows的区别

◆ Windows 以分区为树根，有多个分区时形成多个树并列：



◆ Linux 所有文件系统都安装在一个**树根**上。进行分区时必须**先划分一个根分区**，然后**将其它的分区都挂载到这个根目录下**

当前工作目录

□ 当前工作目录

- ◆ 用户目前所处的目录
- ◆ 用户登录后进入的目录通常是自己的主目录
- ◆ 可用 `pwd` 命令查看用户的当前目录
- ◆ 可用 `cd` 命令来切换目录
- ◆ 一些特殊字符的特殊含义：
 - ✓ “`.`” 表示当前目录
 - ✓ “`..`” 表示当前目录的上一级目录（父目录）
 - ✓ “`-`” 表示用 `cd` 命令切换目录前所在的目录
 - ✓ “`~`” 表示用户主目录的绝对路径名

路径 (path)

□ 路径 (path)

- ◆ 文件或目录在文件系统中所处的位置
- ◆ 路径分为“绝对路径”和“相对路径”
 - 绝对路径：从根目录开始的完整的路径
 - 相对路径：相当于当前工作目录的路径
- ◆ 任何不以 / 和 ~ 开始的路径均为相对路径

举例

设用户当前所在的目录为： `/home/xym/Teaching/`
则该目录的子目录 `ParaComp/` 中的文件 `lect01.ppt`
的绝对路径为

`/home/xym/Teaching/ParaComp/`

相对路径为

`./ParaComp/`

Linux 常用命令

ls	显示目录内容	cat	查看文件内容
cd	切换工作目录	more	查看文件内容
pwd	显示工作目录	less	查看文件内容
cp	拷贝文件	gzip gunzip	文件压缩和解压
mkdir rmdir	创建/删除目录	ln	创建文件链接
mv	修改文件名	tar	文件打包
rm	删除文件	zip unzip	文件压缩和解压
chmod	改变文件的权限	dos2unix	格式转换

ls

❑ ls: 显示文件或目录信息

ls [选项] [目录或是文件]

- ◆ 对于目录，该命令将列出其中的所有子目录与文件
- ◆ 对于文件，ls 将输出其文件名以及所要求的其他信息
- ◆ 默认情况下，输出条目按字母顺序排序
- ◆ 当未给出目录名或文件名时，就显示当前目录的信息

ls

常用选项

- a 列出目录下的所有文件，包括以 . 开头的隐含文件
- l 列出文件的详细信息，通常称为“长格式”
- t 以时间排序
- d 输入参数是目录时，只显示该目录名，而不是列出该目录下的文件
- R 列出所有子目录下的文件
- S 以文件大小排序
- r 反序输出
- help 显示简短帮助信息
- version 输出版本信息

选项 `--help` 和 `--version` 几乎对所有命令均有效

ls

举例

```
ls
```

列出当前目录的内容

```
ls -l
```

以长格式列出当前目录下的文件

```
ls -al
```

以长格式列出当前目录下的所有文件，包括隐藏文件

```
ls -lt
```

按最后修改时间顺序，以长格式列出当前目录下的文件

cd / pwd

❑ cd: 改变工作目录

cd 目录

- ◆ 进入指定的目录，即将工作目录改为指定的目录

❑ pwd: 显示当前目录

pwd

- ◆ 告诉用户当前的工作目录，即当前在哪个目录下

cp

❑ cp: copy, 文件复制

cp [选项] 源文件或目录 目标文件或目录

- 把源文件复制为目标文件，若目标文件已存在，则被覆盖
- 把多个源文件复制到目标目录中
- 把源目录复制为指定的目标目录（目标目录不存在）
- 把源目录复制到指定的目标目录中

常用选项

- f 若目标文件与源文件同名，则直接覆盖而不提示
- i 交互式拷贝，即覆盖目标文件之前将给出提示
- p 保留源文件的修改时间和访问权限等属性
- r 递归，复制目录时必须加此选项

cp

举例

```
cp file1 /home/xym/data/
```

```
cp -i file1 /home/xym/file1.bak
```

将文件 `file1` 复制到 `/home/xym/` 目录下，并取名为 `file1.bak`。若不希望重新命名，可以使用下面的命令：

```
cp -i file1 /home/xym/
```

```
cp -pr /home/xym/bin/ /tmp/
```

将 `/home/xym/bin/` 目录中的所有文件及其子目录复制到 `/tmp/` 目录下，并保留所有文件的相关属性

mkdir / rmdir

❑ mkdir / rmdir: 创建和删除目录

`mkdir` 目录名

`rmdir` 目录名

举 例

```
mkdir data
```

在当前目录下创建一个名为 `data` 的目录

```
rmdir olddata
```

删除当前目录下的 `olddata` 目录，若该目录非空，则报错

`rmdir` 只能删除空目录，非空目录可用 `rm -r` 删除

mv

□ mv: 移动或更名现有的文件或目录

mv [选项] 源文件或目录 目标文件或目录

- ◆ 将源文件改名为目标文件，若目标文件已存在，则被覆盖
- ◆ 将多个源文件移至目标目录中
- ◆ 将源目录改名为指定的目标目录（目标目录不存在）
- ◆ 将源目录移至指定的目标目录中

常用
选项

- f 若目标文件与源文件同名，则直接覆盖而不提示
- i 在覆盖目标文件之前要求用户确认（交互式移动）
- u 如果要被覆盖的文件或目录比源文件要新，则不覆盖

mv

举例

```
mv file1 mydata/
```

将文件 `file1` 移至当前目录的子目录 `mydata/` 中

```
mv -i file1 mydata/file2
```

将文件 `file1` 移至 当前目录的子目录 `mydata/` 中，并改名为 `file2`

```
mv data1/ /home/zhang/data
```

- ✓ 若目标目录 `/home/zhang/data` 不存在，则将目录 `data1/` 移至 `/home/zhang/` 目录中，并改名为 `data`
- ✓ 若目标目录 `/home/zhang/data` 已存在，则 `data1/` 将被移至 `/home/zhang/data` 目录下
- ✓ 若 `/home/zhang/data` 中已存在目录 `data1/`，则报错

rm

❑ rm: 删除文件或目录

rm [选项] 文件或目录名

◆ 删除指定的文件或目录

常用选项

- f 强制删除，并且不给提示
- i 删除每个文件时给用户提示
- r 删除整个目录，包括文件和子目录，
用 rm 删除目录时必须使用该选项

举例

```
rm file1.txt
```

```
rm -i file1.txt
```

```
rm -rf mydata/
```

cat

❑ cat: 查看文件内容

`cat` [选项] 文件列表

- ◆ 功能之一: 显示文件内容
- ◆ 功能之二: 将两个或多个文件连接起来 (重定向)

常用选项

`-n` 在文件的每行前面显示行号

举例

```
cat -n file1
```

```
cat file1 file2 > file3
```

当文件内容较长时, 可用 `more` 或 `less` 等命令分屏显示

more / less

□ more / less: 查看文档内容

more 文件名

less 文件名

- ◆ 逐屏显示文本，按空格键显示下一屏，退出按 q
- ◆ more 显示文件内容时，只能往前翻
- ◆ less 可以往前翻，也可以向后翻
- ◆ less 可能无法正常显示中文，此时可用 more 或 cat

gzip / gunzip

❑ gzip/gunzip: 文件压缩与解压缩

gzip [选项] 文件列表

gunzip [选项] 文件列表

- Linux下常用的压缩和解压缩命令
- 压缩后 gzip 会在每个文件的后面添加扩展名 .gz
- 压缩后原文件会被自动删除
- 在 windows 下可以用 winzip 或 winrar 解压

gizip / gunzip

常用选项

- d 解压
 - l 对每个压缩文件，显示： 压缩文件的大小、未压缩文件的大小、压缩比、未压缩文件的名字等
 - r 递归
 - t 测试选项，检查压缩文件是否完整
 - v 显示文件名和压缩比
 - n n 为一个数字，代表压缩比，取值范围为 1~9
 - 1 或 --fast 表示最快压缩方法（低压缩比）
 - 9 或 --best 表示最好压缩方法（高压缩比）
- 系统缺省值为 6

gzip / gunzip

举 例

```
gzip file1.txt
```

压缩 `file1.txt`，压缩后取名为 `file1.txt.gz`，
原文件被删除

```
gzip -r9 /home/xym/data/
```

以最大压缩比，将目录 `/home/xym/data/` 下的所有文件进行**逐个**压缩，包括**子目录**下的文件，并在每个文件后添加扩展名 `.gz`

ln

❑ ln: 创建链接 (link)

ln [选项] 目标 [链接名]

- ◆ 为**目标**创建一个链接
- ◆ 创建链接，实际上是给系统中已有的文件或目录指定另外一个可用于访问它的名称（类似快捷方式）
- ◆ 删除链接，不会破坏原来的文件和目录
- ◆ 链接有两种：**硬链接**和**符号链接**，缺省是建立硬链接
- ◆ 若没有指定链接名，则链接名与原文件名相同

常用
选项

-s 建立符号链接

ln

举 例

```
ln -s file1.txt /home/zhang/
```

在 `/home/zhang/` 目录下建立一个指向 `file1.txt` 的符号链接，链接名为 `file1.txt`，若 `/home/zhang/` 目录中已存在名为 `file1.txt` 的文件，则系统会报错

```
ln -s /home/xym/bin /home/zhang/
```

在 `/home/zhang/` 目录下建立一个指向目录 `/home/xym/bin/` 的符号链接，链接名为 `bin`，若 `/home/zhang/` 目录中已存在名为 `bin` 的文件或目录，则报错

tar

❑ tar: 文件打包和解包

tar [主选项+辅选项] 文件或者目录

- ◆ 将多个文件或目录打包成一个文件
- ◆ **tar** 有主选项和辅选项，主选项是必须要有的，它告诉 **tar** 要做什么事情，辅选项是辅助使用的，可以选用

常用主选项

- c 创建新的档案文件（打包）
- r 把要存档的文件追加到档案文件的末尾
- t 列出档案文件的内容，查看已经备份了哪些文件
- u 更新档案中的已有文件，如果备份文件中没有要更新的文件，则将该文件追加到备份文件的最后
- x 从档案文件中释放文件（解包）

tar

常用 辅助 选项

- f** 使用档案文件或设备，这个选项通常是必选的
- v** 详细列出 **tar** 处理的文件信息
- z** 用 **gzip** 来压缩/解压缩文件包
- j** 用 **bzip2** 来压缩/解压缩文件包

举 例

```
tar -cvf data1.tar data1
```

```
tar -czvf data1.tgz data1
```

```
tar -xzvf data1.tgz
```

输出文件名
一定要给出

tar 的选项前的 符号 “-” 可以不加

zip / unzip

❑ zip / unzip: 压缩与解压缩 zip 文件

zip [选项] 文件列表
unzip 压缩文件

◆ 作用与windows下的 winzip 类似，压缩文件或目录

常用
选项

-e 对压缩后的文件加密
-r 递归，压缩目录时必须使用该选项
-u 更新文件

举
例

```
zip file1.zip file1
```

```
zip -r data1.zip data1/
```

```
unzip data1.zip
```


dos2unix

❑ dos2unix: 格式转换

dos2unix 文件

- 将 Windows 下的文本文件转化成 Linux 下的文本文件

举例

```
dos2unix hello_world.c
```

文件访问权限

❑ Linux 文件类型常见的有：

普通文件、目录、符号链接、设备文件等

❑ 访问权限

◆ 每个文件或目录都有访问权限，这些权限决定了谁能访问和如何访问这些文件或目录

◆ 通过设定权限，可以实现以下三种方式来限制访问权限：

- 只允许用户自己访问
- 允许与自己同组的用户访问
- 允许系统内的所有用户访问

文件访问权限

◆ 一个文件或目录可以有**读**、**写**和**执行**的权限。

当一个文件或目录被创建时，系统会自动赋予文件所有者的读和写的权限，这样就允许文件所有者能够阅读和修改该文件

◆ 一个文件或目录可以被**三种不同类型的用户**访问：**所有者**、**同组用户**和**其它用户**。

通常每个用户在创建时都被分配到一个特定的用户组，系统中存在许多不同的用户组。文件所有者可以通过修改文件的属性而授予其所在用户组的其他成员访问该文件的权限，同样也可以将自己的文件向系统内的所有用户开放。

文件访问权限

□ 查看文件和目录的属性: **ls -l**

```
[jypan@euler jypan]$ ls -l
total 1208
-rw-r--r--    1 jypan    tch      702019  6月  7 13:29 2ndECpde2.zip
-rw-r--r--    1 jypan    tch      472956  3月 15 11:52 2ndECpde.zip
drwx-----   2 jypan    tch       4096  5月 13 12:22 backup
drwxr-xr-x    3 jypan    tch       4096  5月  3 01:35 bin
lrwxrwxrwx    1 root     root        11 12月 27 2003 Desktop -> .Desk
```

The diagram illustrates the components of the `ls -l` command output. Red arrows point from specific fields in the command output to their corresponding labels in Chinese:

- `-rw-r--r--` points to **文件类型** (File type)
- `-rw-r--r--` points to **文件权限** (File permissions)
- `1` points to **链接个数** (Number of links)
- `jypan` points to **文件所有者** (File owner)
- `tch` points to **文件所有者所在的用户组** (User group of the file owner)
- `702019` points to **文件长度** (File size)
- `6月 7 13:29` points to **文件上次修改的时间和日期** (Time and date of last modification)
- `2ndECpde2.zip` points to **文件名** (Filename)

文件访问权限

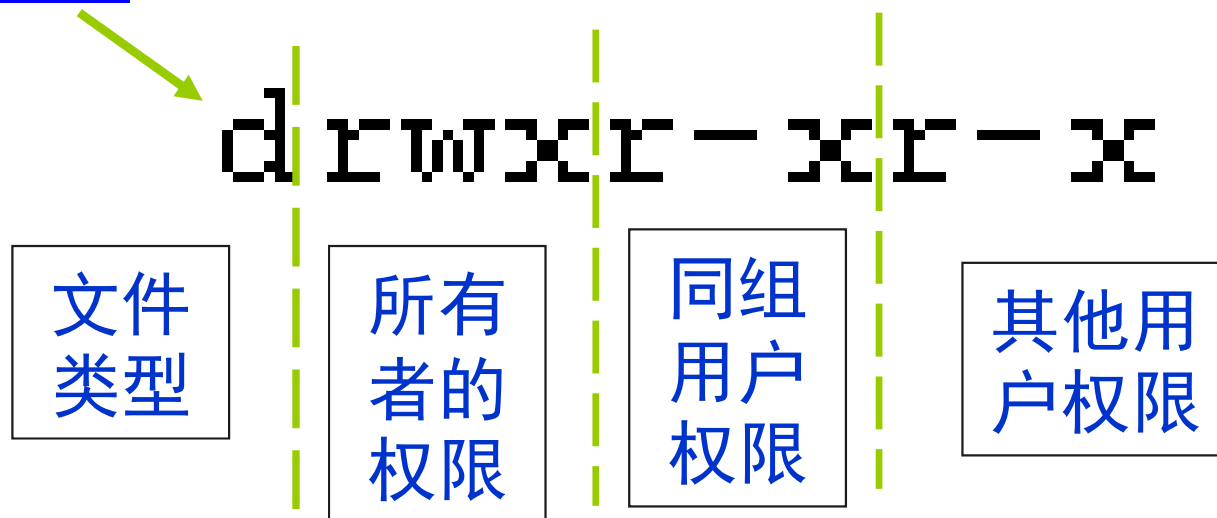
```
drwxr-xr-x    3 jypan    tch          4096  5月  3 01:35 bin
```

◆ 在显示的结果中，第一个字段中的第一个字符是表示文件的类型，具体含义为：

字符	代表的文件类型
-	普通文件
d	目录
l	链接文件
b、c	分别代表区块设备和其它外围设备，特殊类型文件
s、p	这些文件类型关系到系统的数据结构和管道，很少见

文件访问权限

drwxr-xr-x 3 jypan tch 4096 5月 3 01:35 bin



- 在显示的结果中，第一个字段的第 2~10 个字符是用来表示权限。这 9 个字符每 3 个一组，组成 3 套权限控制，
第一套控制文件所有者的访问权限
第二套控制所有者所在用户组的其他成员的访问权限
第三套是控制系统内其他用户的访问权限

文件访问权限

d rwx r - x r - x

- 每套控制中三个字符按顺序排列，所代表的含义如下：

r	Read (读取)：对文件而言，具有阅读文件内容的权限， 对目录而言，具有浏览目录内容的权限。
w	Write (写入)：对文件而言，具有修改文件内容的权限， 对目录而言，具有移动、删除目录内文件的权限。
x	eXecute (执行)：对文件而言，具有执行文件的权限， 对目录而言，具有进入目录的权限。
-	表示不具有该权限，或取消该权限

访问权限举例

例：

-rwx------

普通文件

系统内其它用户没有读、写和执行权限

用户组其它成员没有读、写和执行权限

用户有读、写和执行权限

只有文件所有者具有读、写和执行权限

访问权限举例

- rw-r--r--** 普通文件，用户有读、写权限，组用户和其他用户只有读权限。
- drwx--x--x** 目录，用户有读、写和执行权限，组用户和其他用户只有执行权限。
- drwx-----** 目录，用户有读、写和执行权限，组用户和其他用户不能访问。
- rwxr-xr-x** 普通文件，用户有读、写和执行权限，组用户和其他用户有读和执行权限。

每个用户都拥有自己的专属目录（主目录），通常放置在 **/home** 目录下，这些专属目录的默认权限通常为

rwx-----

修改访问权限

- ❑ 如何更改已有文件或目录的访问权限？

`chmod`

- ❑ `chmod` 有两种用法：

- ◆ 文字设定法

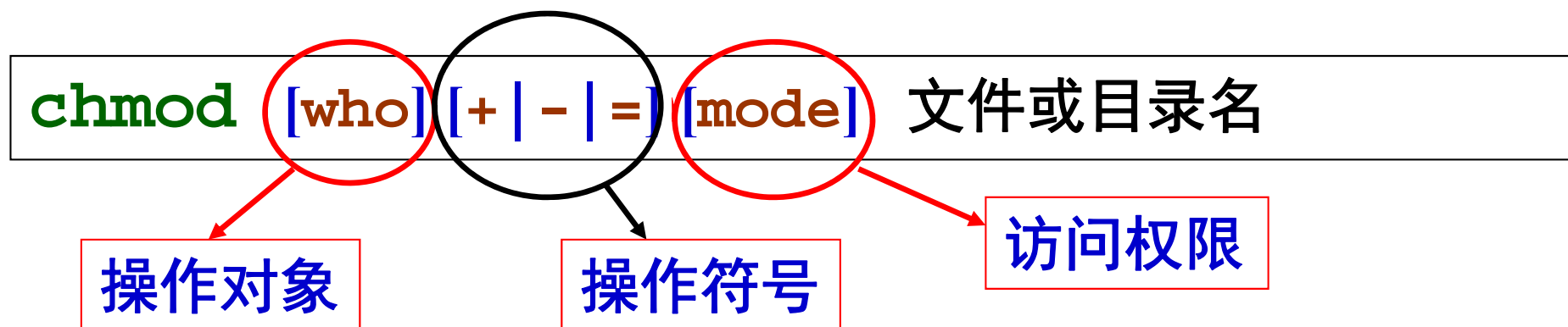
使用字母和操作符表达式来修改或设定文件的访问权限

- ◆ 数字设定法

利用数字来设定文件的访问权限

chmod

◆ chmod 文字设定法



● 操作对象 **who** 可是下述字母中的任一个或者它们的组合：

- ✓ **u** 表示“用户（**user**）”，即文件或目录的所有者
- ✓ **g** 表示“同组（**group**）用户”
- ✓ **o** 表示“其他（**others**）用户”
- ✓ **a** 表示“所有（**all**）用户”
- ✓ 如果没有指定操作对象 **who**，则缺省值为 **a**

chmod

- 操作符号可以是：

- ✓ + 添加某个权限。
- ✓ - 取消某个权限。
- ✓ = 赋予给定权限并取消其他所有权限（如果有的话）

- 访问权限可用下述字母的任意组合

- ✓ r 可读
- ✓ w 可写
- ✓ x 可执行

- 文件或目录名

以空格分开的要改变权限的文件列表，支持通配符

chmod

举 例

```
chmod o-rx mydata/
```

```
chmod o=rx mydata/
```

```
chmod go=r mydata/
```

```
chmod +x myname.txt
```

等价于: `chmod a+x myname.txt`

```
chmod +w myname.txt ?
```

等价于: `chmod u+w myname.txt`

chmod

◆ 数字设定法

`chmod num 文件或目录名`

- 我们必须先了解如何用数字表示的属性：

- 0 表示没有权限
- 1 表示可执行权限
- 2 表示可写权限
- 4 表示可读权限

然后将其相加，即可得到一个表示访问权限的数字。

- 一个文件的访问权限可以用三个数字来表示，分别代表用户、同组用户和其它用户的访问权限

chmod

权限	对应数字
r	4
w	2
x	1
-	0

-rw-r--r--  **644**

drwx--x--x  **711**

drwx-----  **700**

-rwxr-xr-x  **755**

chmod

举 例

```
chmod 644 myname.txt
```

设定文件 `myname.txt` 的属性为: `-rw-r--r--`

```
chmod 750 myname.txt
```

设定文件 `myname.txt` 的属性为: `-rwxr-x---`

```
chmod 700 mydata/
```

设定目录 `mydata` 的属性为: `-rwx-----`

Linux Shell 介绍

- ❑ 许多Linux 发行版的缺省 Shell 为 **Bash**
- ❑ Shell 的一些基本使用技巧
 - 自动补全功能: **Tab**
 - 历史记录: **history**、上下箭头键
 - 别名: **alias**
 - 重定向和管道: **> >> < << |**
 - 环境变量
 - 通配符: *** ? []**
- ❑ 正则表达式、文本过滤、Shell 编程（略）

进程管理

- **ps**: 查看正在运行的进程

-A	显示所有进程
-f	完全显示
-l	长格式显示，可查看各个进程的优先权值
u	增加显示用户名，进程起始时间，CPU 和内存占用百分比等信息
-u	查看指定用户的进程
r	显示正在运行的进程

- 后台进程: **&**
- 终止一个进程: **Ctrl+c** 或 **kill**

主要内容

- ❑ Linux 介绍
- ❑ Linux 系统的安装
- ❑ Linux 基本使用与管理
- ❑ Linux 基本命令
- ❑ Linux 程序开发环境

VIM

□ VIM 介绍

- vim 是Linux下最常用的文本编辑器之一
- vim 是一个纯字符编辑器
- vim 没有菜单，只有命令
- 在线帮助： [vimtutor](#)

□ VIM 的启动

vi 文件名

- 文件若存在，则打开该文件，否则将新建一个文件

VIM

□ VIM 的工作模式

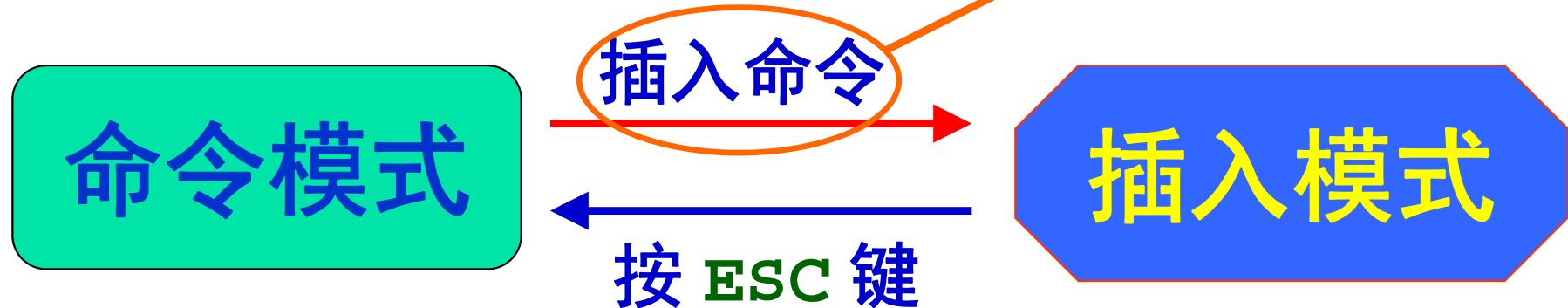
→ 命令模式

该模式下，所有的输入都被看成是命令

→ 插入模式

在该模式下可以输入文件内容

□ 工作模式的切换



- 启动 vim 编辑器时，首先进入的是命令模式
- 在任何模式下，按 ESC 键 便可返回到命令模式

VIM

□ VIM 的退出

→ 先保存，后退出

→ 具体步骤：

第一步：切换成命令模式

第二步：使用下面的命名保存文件或退出 vim

- **:w** 保存当前编辑文件，但并不退出vim
- **:q** 退出 vim
- **:wq** 表示存盘并退出vim
- **:q!** 放弃所作的修改而直接退出vim

VIM

□ VIM 中光标的移动

- 可使用方向键或命令，但不能使用鼠标

- 删除文本

插入模式: Delete、Backspace

命令模式: x、dd、重复因子

Linux 编程环境

□ C 语言/ Fortran 程序的编译

例: `gcc -O2 -o hello hello.c`
`gfortran -O2 -o hello hello.f95`

- O2 编译时对进行一定的优化
- o 指定编译生成的可执行文件的文件名
(若不指定, 缺省为 `a.out`)

□ 运行

`./hello`

`./a.out`