

私有软件的问题:

软件私有的问题

1. **限制了软件的使用人数。**假设一个软件已经开发出来，那么这个软件的所有投资都已经付出（无论有多少人使用它）；从社会的角度来看，任何限制软件使用的行为都是对其投资的浪费。
2. **破坏了社会和谐。**假如你和你的邻居都想使用同一个软件，你购买了这个软件，你的邻居向你借，那你就陷入两难的境地。如果借，那么你违反了版权法，如果不借，你就违背邻里互助的美德。
3. **不允许用户修改代码。**这导致了用户无法根据自己的需求重写软件。如果一个现有的软件不能满足需求（很可能只要在原有的修改少量代码即可），用户要么自己重新写一个软件，要么就忍受现有软件的不完美。
4. **不利于软件的开发。**我们无法获得源代码（因为这属于商业秘密），所以我们每个人都无法借鉴现有的软件开发经验，至多我们只能站在周围人的肩膀上。



Free Software!

Wednesday, April 14, 2021

8/28

限制了软件的使用人数；
破坏了社会和谐；
不允许用户修改代码；
不利于软件的开发。

Linux 的特点:

1.2 Linux的特点

- 高效性和灵活性
- 不同的Linux发行包差别有限
- Linux能够完成关键业务
- Linux更加安全可靠
- Linux花费很小

1.5 Linux 新特点

- 处理同步编程的新方法
- 高性能的新内核:5.x.y (<https://www.kernel.org/>)
- 桌面环境更丰富(XWindow)
- 易操作的图形界面
- 网络软件趋于成熟
- 管理功能齐全



常用的 shell:

1.3 Linux的结构

• 内核（又称核心，kernel）

- 进程与内存管理
- 文件管理
- I/O设备管理

• 实用程序

- 命令解释器SHELL
- 汇编程序
- 编译器
- 编辑器
- 调试器
- ...

- SH
- BASH: Bourne Again SHell
- KSH
- CSH
- ...



Shell 的职责：

The Linux Shells

- `$echo $SHELL`
- `$cat /etc/shells`

Responsibilities of the Shell

1. Reading input and parsing the command line
2. Evaluating special characters, such as wildcards and the history character
3. Setting up pipes, redirection, and background processing
4. Handling signals
5. Setting up programs for execution

1. 读取输入并解析命令行
2. 评估特殊字符，例如通配符和历史字符
3. 设置管道、重定向和后台处理
4. 处理信号
5. 设置执行程序

Linux 的发行版本：

1.4 Linux发布版本

GNU/Linux Distribution

- Ubuntu
- RedHat
- openSUSE
- Fedora
- Debian GNU/Linux
- Slackware Linux
- Gentoo Linux
- ...

Linux 有几种安装类型：

GRUB 和 LILO：

2.2.3 安装步骤 -1

- 创建Linux分区
- 安装Linux软件
- 配置X Window系统操作界面
- 安装LILO(Linux LOader)或GRUB(推荐)
- 创建新的用户账户。

- GNU GRUB (GRand Unified Boot loader)
- LILO 没有交互式命令界面，而 GRUB 拥有。
- LILO 不支持网络引导，而 GRUB 支持。
- LILO 将关于可以引导的操作系统位置的信息物理上存储在 MBR 中。如果修改了 LILO 配置文件，必须将 LILO 第一阶段引导加载程序重写到 MBR。相对于 GRUB，这是一个更为危险的选择，因为错误配置的 MBR 可能会让系统无法引导。使用 GRUB，如果配置文件配置错误，则只是默认转到 GRUB 命令行界面。

运行级别：

运行级别



- *0: 关机级别
 - *1: 单用户运行级别，运行rc.sysinit和rc1.d目录下的脚本
 - *2: 多用户，但系统不会启动NFS，字符模式，在有些linux系统中，级别2为默认模式，具有网络功能，如ubuntu.debian
 - *3: 多用户，字符模式，系统启动具有网络功能，redhat常用运行级别
 - *4: 用户自定义级别
 - *5: 图形界面模式，redhat常用运行级别
 - *6: 重启级别
 - *S: 单用户运行级别，只运行rc.sysinit文件
- **runlevel**: runlevel 显示上次的运行级别和当前的运行级别，“N”表示没有上次的运行级别。man runlevel
- **init [0123456]**: (切换运行级别)即在 init 命令后跟一个参数，此参数是要切换到的运行级的运行级代号，如：用 init 0 命令关机；用 init 6 命令重新启动。

描述什么是管道，管道的好处是什么：

管道是一种进程通信的方式，在 linux 中用“|”代表管道操作，linux 利用管道将两个或多个命令联系起来，将管道左边的命令执行后标准输出的内容通过管道作为管道右边的命令的标准输入，以此类推。

管道的好处是可以将多个命令链接起来，实现多种命令联合使用，达到更加丰富的处理效果。

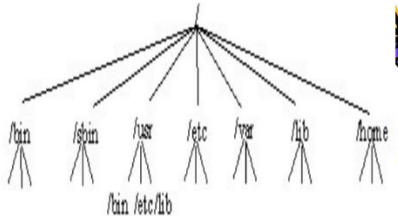
Linux 与 Dos 文件系统的区别：

6.1.1 Linux与DOS文件系统的区别



➤ Linux文件系统

- ✓ 单个目录树的结构
- ✓ 根是根目录“/”
- ✓ 往下连接各个分支



➤ DOS文件系统

- ✓ 每个分区为树根
- ✓ 多个分区，形成了多个树并列

Linux 有哪些文件类型：

6.2 Linux的文件

➤ 文件名

- ✓ 字母字符、下划线和数字组成，也可以加上句号和连字符
- ✓ 文件名和扩展名之间用句点隔开

➤ 文件类型

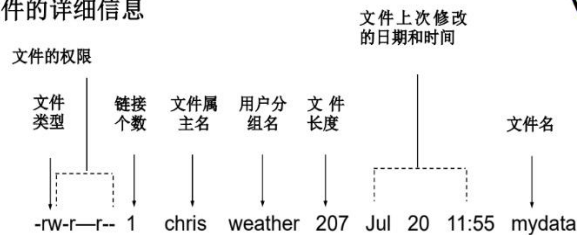
- ✓ 物理格式：字节流 (byte stream)
- ✓ 类型：file 命令查看
 - - 普通文件：纯文本；二进制；数据格式
 - d 目录文件
 - l 连接文件
 - b 块设备特殊文件
 - c 字符设备特殊文件
 - s 套接字文件 (socket)



Linux 文件属性中各字段的意义：

6.2.3 文件权限

➤ 文件的详细信息



- 三个安全级别：所有者级别、组访问级别、其他用户访问级别
- 文件和子目录：读（Read）、写（Write）、执行（eXecute）
- 访问权的分配：文件属主（owner/user）、用户分组（Group）、其他人（other）

Linux 根目录下一般有哪些目录？主要目录的含义与作用：

bin：存放所有二进制命令（用户）

sbin：系统管理员命令存放的目录

boot：linux 内核及引导系统程序所需的目录

dev：所有设备文件的目录

etc：二进制安装包配置文件默认路径，服务启动命令存放目录

home：普通用户的主目录

lib：库文件存放目录

proc：进程及内核信息存放目录

usr：系统存放程序的目录 ——bin：用户可执行文件目录（应用程序）

tmp：临时文件目录

var：存放内容常变动的文件目录

ftp 基本命令：

open:链接服务器；

操作服务器：

pwd:显示 linux 服务器路径；

cd:切换 linux 服务器路径；

ls:列出 linux 服务器当前目录下文件；

rm:删除 linux 服务器文件；

put:上传文件；（-r 循环递归文件夹）

get:下载文件。（-r 循环递归文件夹）

操作本地计算机：

lcd:切换 windows 路径；

lpwd:显示 windows 路径；

!dir:显示本地计算机当前目录文件。

关闭链接：

close:关闭链接，不退出 psftp；

quit/exit:关闭链接并退出 psftp。

进程与程序：

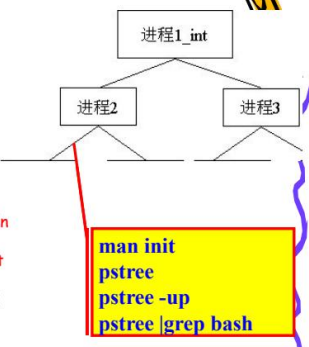
7.1 进程概述

- **进程（process）**：正运行的程序，即程序只有在被系统载入了内存并运行后才能够叫做进程。
- **程序**是磁盘文件，而进程则是内存中工作着的代码。

父子进程之间的关系：

7.1.2 进程间关系

- 复制机制来产生进程
- 进程关系：树形关系
- 每一个进程都记录了它的父进程和子进程的ID
- 进程结束之后退回到它的父进程
- 任何子进程都自动从父进程那里继承三个打开的设备文件：
 - 标准输入设备（键盘、鼠标）：**stdin**（**standard incoming**,标准输入）
 - 标准输出设备（显示器等）：**stdout**（**standard outputting**,标准输出）
 - 做标准错误输出（**stderr**, **standard error**）



进程状态：

进程状态 **man ps**

符号	含义
R	运行或准备运行
S	睡眠状态
I	空闲
Z	僵尸
D	不间断睡眠
W	进程没有驻留页
T	停止或跟踪

前后台进程：

7.2 进程类型

- **前台和后台进程**
 - 前台：一个程序控制着标准输入输出
 - 后台：一个程序不从标准输入接受输入，一般也不将结果输出到标准输出
- **守护进程**
 - 一般以后台进程的方式存在
 - 开机即被载入到系统中并常驻在系统中，直到关机时才结束，也被专称为“**Daemon**”，

/proc 目录:

- cmdline: 内核启动的命令行;
- cpuinfo: 有关系统 cpu 的信息;
- devices: 字符和块设备的主设备号, 以及分配到这些设备号的设备名称;
- filesystems: 可供使用的文件系统类型;
- interrupts: 系统的中断信息;
- loadavg: 以几个不同的时间间隔计算的系统平均负载;
- locks: 在打开文件上的加锁信息;
- meminfo: 内存状态信息;
- uptime: 系统运行的秒数和空闲的秒数;
- version: 运行的内核版本;

\$pid:

- fd: 目录, 包含进程打开文件的情况 (文件描述符);
- environ: 进程的环境变量;
- cwd: 进程当前的工作目录;
- exe: 实际运行程序的链接;
- cmdline: 进程完整的命令行信息。

库文件:

静态链接库就是在程序编译的时候就被加载进来, 这样的可执行文件会比较大一些, 还不能共享;

动态链接库是在程序执行的时候加载, 可共享, 占用空间较小。

-I: 先在所指定的目录查找, 然后再按常规的顺序去找。

-L: 指定编译时搜索库的路径。

-l***: 编译时告诉编译器库函数的路径。(默认标准路径)

makefile:

格式:

目标文件:依赖文件

gcc 编译语句 (可省略)

#: 注释

每行开头使用 tab

\$@:目标文件

\$^:所有依赖文件

\$<:第一个依赖文件

Make 默认执行顺序文件名: GNUMakefile、makefile、Makefile

gcc -M ***.c //自动链接依赖文件

gcc -MM ***.c //除标准库的头文件, 与-M 类似

gcc -M -MF 'name' ***.c //将依赖关系写入以 name 命名的文件中

gcc -M ***.c -MT 'name' //指定依赖规则中目标名为 name

shell 编程:

部分只读的环境变量

- \$0 程序名称
- \$1-9 前9个命令行参数的值
- \$* 所有命令行参数的值
- @\$ 所有命令行参数的值
- \$# 命令行参数的个数
- \$\$ 当前进程的进程ID
- \$? 最近使用命令的退出状态
- \$! 最近后台进程的进程ID
- PS1 shell命令提示符 `~/.bashrc`
- PS4 shell调试提示符

0 OK!
1 Operation not permitted
2 No such file or directory
13 Permission denied
127 command not found
130 terminated by Ctrl-C

常用环境变量：

常用的环境变量

PATH 决定了shell将到哪些目录中寻找命令或程序,系统就从这些目录里寻找可执行文件.具体见附件.

ROOTPATH 这个变量的功能和PATH相同,但它只罗列出超级用户 (root) 键入命令时所需检查的目录。

LDPATH(LD_LIBRARY_PATH) 这个变量包含了一系列用冒号隔开的目录,动态链接器将在这些目录里查找库文件.具体见附件.

HOME 当前用户主目录

HISTSIZE 历史记录数

LOGNAME 当前用户的登录名

HOSTNAME 指主机的名称

SHELL 当前用户Shell类型

LANGUAG 语言相关的环境变量,多语言可以修改此环境变量.

MAIL 当前用户的邮件存放目录.

PS1 基本提示符,对于root用户是#,对于普通用户是\$.

MANPATH 这个变量包含了一系列用冒号隔开的目录,命令man会在这些目录里搜索man页面.

INFODIR 这个变量包含了一系列用冒号隔开的目录,命令info将在这些目录里搜索info页面.

PAGER 这个变量包含了浏览文件内容的程序的路径 (例如less或者more) .

EDITOR 这个变量包含了修改文件内容的程序 (文件编辑器) 的路径.比如nano或者vi.

KDEDIRS 这个变量包含了一系列用冒号隔开的目录,里面放的是KDE相关的资料.

CONFIG_PROTECT 这个变量包含了一系列用空格隔开的目录,它们在更新的时候会被Portage保护起来.

CONFIG_PROTECT_MASK 这个变量包含了一系列用空格隔开的目录,它们在更新的时候不会被Portage保护起来

Shell structure:

test expression / [expression] **空格**

逻辑操作: 与 -a 或 -o 非 !

Test Operator for **integers**: int1 relop int2

Relop	Description
-gt	Greater than
-ge	Greater than or equal to
-eq	Equal to
-ne	Not equal to
-le	Less than or equal to
-lt	Less than

• The test built-in options for **files**

Option	Test Performed on file
-d filename	Exists and is a directory file
-f filename	Exists and is a regular file
-r filename	Exists and it readable
-s filename	Exists and has a length greater than 0
-u filename	Exists and has setuid bit set
-w filename	Exists and it writable
-x filename	Exists and it is executable
FILE1 -ot FILE2	FILE1 is older than FILE2

• **String** testing

Criteria	meaning
-n string	True if string has a length greater than zero (True if nozero)
-z string	True if string has a length of zero
String1 = string2	True if string1 is equal to string2
String1 != string2	True if string1 is not equal to string2

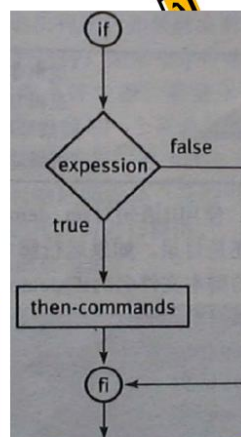
if ... then

• Structure

```
if expression
then
  then-command
fi
```

Example:

```
if test "$word1" = "$word2"
then
  echo "Match"
fi
```



if...then...else

- Structure

```
if test-command
then
  commands
else
  commands
fi
```

- You can use semicolon (;) ends a command the same way a NEWLINE does.

```
if [ ... ]; then
... ..
fi
```

```
if [ 5 = 5 ]; then echo "equal"; fi
```

if...then...elif

- Structure

```
if test-command
then
  commands
elif test-command
then
  commands
.
.
.
else
  commands
fi
```

for

- Structure

```
for loop-index
do
  commands
done
```

- Automatically takes on the value of each of command line arguments, one at a time. Which implies

 for arg in "\$@"

for... in

- Structure

```
for loop-index in argument_list
do
    commands
done
```

Example: **sh dirlist**

```
for file in *
do
    if [ -d "$file" ]; then
        echo $file
    fi
done
```

如何运行:
1. ls |bash dirlist
2. bash dirlist

while

- Structure

```
while test_command
do
    commands
done
```

Example:


```
while [ "$number" -lt 10 ]
do
    ... ..
    number=`expr $number + 1`
done
```

break and continue

case

- Structure

```
case test_string in
    pattern-1 )
        commands_1
        ;;
    pattern-2 )
        commands_2
        ;;
    ... ..
esac
```

 default case: catch all pattern

```
* )
```

- Special characters used in patterns



Pattern	Matches
*	Matches any string of characters.
?	Matches any single character.
[...]	Defines a character class. A hyphen specifies a range of characters
	Separates alternative choices that satisfy a particular branch of the case structure

命令解释:

ls	list directory contents
pwd	print name of current/working directory
rm	remove files or directories
mkdir	make directories
rmdir	remove empty directories
cp	copy files and directories
mv	move (rename) files
lpr/lp	print files (打印文件)
lpq	show printer queue status (打印机队列状态)
lprm	cancel print jobs (关闭打印)
echo	display a line of text displaying messages and values
cat	concatenate(链接) files and print on the standard output
head	output the first part of files
tail	output the last part of files
more	file perusal filter for crt viewing (分页查看)
less	opposite of more (允许回卷, 可以往回看)
locate	find files by name (查找数据库快照)
find	search for files in a directory hierarchy (实时查找)
grep	print lines matching a pattern
chmod	change file mode bits (permissions)
fdisk	manipulate disk partition table
parted	a partition manipulation program (分区维护程序)
mkfs	build a Linux filesystem
dd	convert and copy a file (转换和拷贝文件)
wc	print newline, word, and byte counts for each file
cmp	compare two files byte by byte (二进制文件)
diff	compare files line by line (文本文件)
sdiff	side by side merge of file differences
sort	sort lines of text files
uniq	report or omit repeated lines
sed	stream editor for filtering and transforming text
tac	concatenate and print files in reverse
rev	reverse lines characterwise
nl	number lines of files (显示行号)
stat	display file or file system status
touch	change file timestamps (时间戳)
ps	report a snapshot of the current process
pstree	display a tree of processes
top	display Linux processes (动态)
nice	run a program with modified scheduling priority (更改进程优先级)
kill	terminate a process
at atq atrm	queue, examine or delete jobs for later execution

crontab maintains crontab files for individual users
indent changes the appearance of a C program by inserting or deleting whitespace
size list section sizes and total size (.o/.exe)
ar create, modify, and extract from archives
nm list symbols from object files
gdb the GNU Debugger
chsh change default shell
id print real and effective user and group IDs
chown change file owner and group
chgrp change group ownership
set/env/printenv 显示所有变量
 set **** 给默认的 argument-list 赋值, 之后执行 shell 脚本使用 set 定义的 argument-list
declare 显示或设置 shell 变量
unset+var 删除变量
export+var 向后面的 shell 传递变量的值 (将变量变为环境变量)
readonly+var 设置变量只读
read read one line of standard input (最后一个变量获取剩余的全部输入内容)
shift+n 移动参数提前 n 个
expr evaluate expressions (乘法 *)
test check file types and compare values
let expressions (运算)
tee read from standard input and write to standard output and files(重定向到多处)
tr translate or delete characters
tar an archiving utility (存档和压缩)
compress/gzip/bzip2/zip/zcat compress files
alias 查看和定义别名 (alias 别名="****")
unalias 取消别名
write send a message to another user
talk talk to another user
msg display messages from other users
telnet user interface to TELNET protocol
ssh open SSH SSH client (remote login program)
links 以字符形式访问 web 网站
lynx 纯文本模式的网页浏览器, 不支持图形、音频、视频等多媒体信息
wget the non-interactive network downloader (非交互式下载)
w3m
ftp Internet file transfer program
ping send ICMP to network hosts
traceroute print the route packets trace to network host
ifconfig configure a network interface
netstat print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships
cal display a calendar
time time a simple command or give resource usage (计时或显示资源情况)

bc 一种任意精度的计算器
factor 分解因子
units 单位/进制转换
spell/ispell 检查拼写
look display lines beginning with a given string
fmt simple optimal text formatter (简单优化文本)
pr convert text file for printing
lp print files
cut remove sections from each line of files
 -c:以字符为单位分割 (1-4 -4 4-)
 -d:自定义分隔符, 默认制表符
 -f:显示某列
awk pattern scanning and processing language (扫描模式与语言处理)
paste merge lines of files (每个文件一列)
script make typescript of terminal session (记录会话)
w show who is logged on and what they are doing
who show who is logged on
last show a listing of last logged in users (所有用户的登录信息)
watch execute a program periodically, showing output fullscreen
dmesg print or control the kernel ring buffer (开机信息)
free display amount of free and used memory in the system(total=used+free+buff/cache)
vmstat report virtual memory statistics
lspci list all PCI devices
df report files system disk space usage
su run a command with substitute user and group ID
finger 查询其他使用者资料
passwd update user's authentication tokens (修改密码)
whereis locate the binary, source, and manual page files for a command
info read Info documents