

Linux 操作系统

课程回顾

李正华

苏州大学

2014 年 6 月 19 日

提纲

ubuntu 安装

Linux 命令行操作

进入 Linux 文本界面

登录、退出和关闭系统

使用 linux 命令

常用基本命令

目录操作命令

文件复制、删除、与重命名

文件查找命令

Shell 的使用

使用图形环境 XWindow

常用软件工具

管理用户和组

ubuntu 安装

- ▶ 建议 12.04.4
- ▶ <http://releases.ubuntu.com/12.04.4/>
- ▶ `ubuntu-12.04.4-desktop-amd64.iso` (64 位机器)
- ▶ 制作 U 盘镜像，安装过程见李中国老师课件
- ▶ kernel 版本应为 `3.11.xxx` (`cat /proc/issue`)

系统直接进入

- ▶ 修改配置文件，选择开机时系统直接进入文本模式
- ▶ RedHat: `/etc/inittab`
- ▶ ubuntu: `/etc/default/grub`
 - ▶ `GRUB_CMDLINE_LINUX_DEFAULT="quiet splash text"`

虚拟控制台

- ▶ Ctrl+Alt+F1-6

GUI 下的仿真终端

- ▶ Ctrl+Alt+T

登录和退出

▶ 文本界面的登录

- ▶ login: zhli
- ▶ Passwd: (光标不会移动!)
- ▶ 进入后: zhli@zhli:~\$
- ▶ su (sudo)
- ▶ root@zhli:/home/zhli#
- ▶ \$ 表示普通用户, # 表示超级用户 root

Linux 的运行级别

- ▶ 0: 停机, 关机
 - ▶ 1: 单用户模式
 - ▶ 2: 多用户模式, 但是没有 NFS (无网络功能)
 - ▶ 3: 完全多用户模式
 - ▶ 4: 未定义
 - ▶ 5: X11
 - ▶ 6: 重新启动
- ▶ `init 5`

关闭和重启

- ▶ 直接关掉电源可能会破坏文件系统!!
- ▶ `init 0; init 6`
- ▶ `shutdown`
- ▶ `halt`
- ▶ `reboot`

- ▶ `sync;sync;sudo shutdown now`
- ▶ `sync` 将任何在内存中缓冲的数据写到磁盘上, 包括修改了的超级块, `inode` 和延迟的读写

shutdown

- ▶ `shutdown [- rhck] time [messages]`
- ▶ `-r`: 关闭 linux 后立即重新引导, 即重启计算机, 同 `reboot`
- ▶ `-k`: 发送警告信号给所有用户, 不真正关机
- ▶ `-c`: `cancel`, 取消正在执行的关机程序
- ▶ `-h`: 关机后关闭电源, 同 `halt`

- ▶ `shutdown +1 "will shutdown after 1 minutes"`
- ▶ `shutdown 12:00 "will shutdown at 12:00"`

linux 命令的基本用法

- ▶ 基本语法: `$ Command Option Arguments`
- ▶ 命令和选项、参数之间必须有空格!
- ▶ 选项是包括一个或多个字母的代码, 前面有一个减号
- ▶ `ls`
- ▶ `ls -l`
- ▶ `ls -l /tmp`
- ▶ `ls -l /tmp /home`
- ▶ `cp a.txt b.txt`

ls I

- ▶ `ls [选项] [目录或文件]`
- ▶ `-a`: 显示所有子目录与文件，包括隐藏文件
- ▶ `-d`: 如果参数是目录，只显示其名称而不显示其下的内容
- ▶ `-F`: 以不同标记注明文件类型，如在目录文件后标记 `"/"`，可执行文件后标记 `"*"`，符号链接后面标记 `"@"`，等等
- ▶ `-i`: 在输出的第一列显示文件的索引节点号 `inode`
- ▶ `-l`: 长格式显示详细信息，每行列出的信息依次是：文件类型与权限、链接数、
- ▶ `-R`: 递归地显示指定目录的各个子目录中的文件
- ▶ `-t`: 按时间排序（和 `-c -u` 搭配使用）

ls II

- ▶ `-c`: 按照文件的修改时间排序
- ▶ `-u`: 按文件上次存取的时间排序

cat

- ▶ concatenate files and print on the standard output
- ▶ 查看文件内容，从键盘读取数据，合并文件等
- ▶ `cat [options] [filenames...]`
- ▶ `-b`: 不显示空行
- ▶ `-E`: 每一行行尾加上 “\$”
- ▶ `-T`: `Tab->“^I”`
- ▶ `-n`: 显示行号
- ▶ `-s`: 连续空行 -> 一个空行
- ▶ `-v`: 显示除了 Tab 和 Enter 以外的所有字符
- ▶ `-A`: 相当于 `-vET`

man

- ▶ 显示任何命令的在线手册页
- ▶ `man [options] command`
- ▶ 空格键 `whitespace`: 显示下一屏
- ▶ 回车键 `Enter`: 向下滚动一行
- ▶ `b`: 回滚一屏
- ▶ `f`: 前滚一屏
- ▶ `/pattern`: 向前搜索
- ▶ `n`: 查找下一次出现
- ▶ `h`: 列出所有的滚动键, `help`

man 选项

- ▶ `-M`: 指定搜索 man 手册页的路径: `MANPATH`
- ▶ `-P`: 指定所使用的分页程序, 默认使用 `/usr/bin/less -is`, 在环境变量 `MANPAGER` 中预设
- ▶ `-S`: 指定 man 命令的章节列表
- ▶ `-a`: 显示所有的手册页, 而不是只显示第一个
- ▶ `-f`: 只显示命令的功能, 而不是显示详细的手册内容
- ▶ `-w`: 不显示手册页, 只显示将被格式化和显示的文件所在位置

man 3 printf

1. Executable programs or shell commands
2. System calls (functions provided by the kernel)
3. Library calls (functions within program libraries)
4. Special files (usually found in /dev)
5. File formats and conventions eg /etc/passwd
6. Games
7. Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7)
8. System administration commands (usually only for root)
9. Kernel routines [Non standard]

mkdir

- ▶ `mkdir [选项] 目录名`
- ▶ `-p`: 可以是一个路径名称, 如果路径中的某些目录不存在, 则自动建好

rmmdir

- ▶ `rmmdir [选项] 目录名`
- ▶ 删除空目录
- ▶ `-p`: 递归删除目录，当子目录删除后其父目录也为空时，也一同被删除。
- ▶ `rm -r` 可以递归删除目录，无论是否为空，但是比较危险

cd

- ▶ cd [目录名]
- ▶ 将工作目录改为指定目录，默认到主目录
- ▶ 必须拥有对指定目录的执行和读权限
- ▶ 可以使用通配符

pwd

- ▶ 显示当前工作目录的绝对路径

cp

- ▶ `cp [options] 源文件或目录 目标文件或目录`
- ▶ `-a`: 常在复制目录时使用。保留链接、文件属性，并递归的复制目录，相当于 `-dpr`
- ▶ `-d`: 复制时保留链接
- ▶ `-f`: 删除已经存在的目标文件而不提示
- ▶ `-i`: 与 `-f` 相反，在覆盖目标文件之前提示，要求用户确认。
- ▶ `-p`: 除了复制源文件内容外，还将其修改时间和访问权限也复制到目标文件中
- ▶ `-r`: 递归复制
- ▶ `-l`: 不复制，只是生成硬链接文件

mv

- ▶ 重命名或者移动位置
- ▶ `mv [options] 源文件或目录目标文件或目录`
- ▶ `-i`
- ▶ `-f`

rm

- ▶ 删除目录或文件
- ▶ `rm [选项] 文件或目录`
- ▶ `-i`
- ▶ `-f`
- ▶ `-r` (删除目录时需要指定)

find I

- ▶ 用于查找文件
- ▶ `find [路径] [匹配表达式]`
- ▶ 可以和通配符一起使用（正则表达式也支持）
- ▶ `-name filename`: 查找指定名称的文件
- ▶ `-user username`
- ▶ `-group`
- ▶ `-print`: 显示查找的结果
- ▶ `-size n`: 查找大小为 `n` 块的文件，一块相当于 512 个字节。`nc` 表示 `n` 个字符。`"+n"` `"+nc"`
- ▶ `-inum n`: 查找 `inode` 为 `n` 的文件
- ▶ `-atime n`: 查找 `n` 天前被访问的文件，`"+n"` `"-n"`

find II

- ▶ `-mtime n`: 修改时间
- ▶ `-ctime n`: 类似与 `atime`, 但是检查的是文件 `inode` 被改变的时间, 即文件属性改变
- ▶ `-perm mod`: 查找给定权限的文件, 如 `777`
- ▶ `-newer file`: 查找比指定文件更新的文件, 即最后修改时间离现在最近。
- ▶ `-exec command`: 对匹配文件执行 `command` 命令, `command` 后面跟大花括号表示文件名, 必须以反斜杠和分号结尾。
- ▶ `-ok command`: 执行 `command` 的时候需要用户确认。
- ▶ `find / -name core -mtime +7`

find III

▶ `find . -iname '*.bak' -exec rm {}`

*Do not forget to enclose the pattern
in quotes in order to protect it from
expansion by the shell
(man find -name)*

Shells 是什么

- ▶ 硬件, linux 内核, shell, 用户命令行输入
- ▶ shell 可以看成是一个命令解释器, 当用户输入命令时, 先判断是否是内部命令或外部命令, 如果都不是, 报错; 如果存在, 系统调用传给 linux 内核。
- ▶ shell 本身是一种解释型的程序设计语言
- ▶ 进入文字界面时, 自动分配一个默认的终端:
`/etc/passwd; echo $SHELL`

shell 下的几个键

- ▶ ↑: 显示上一个命令
- ▶ ↓: 显示下一个命令
- ▶ ←, →
- ▶ backspace 退格

- ▶ history

- ▶ !997

shell 中的特殊字符 I

▶ 通配符

- ▶ ? * [] [!]

- ▶ 文件名前的圆点. 和路径中的斜线/必须显式匹配，例如
"*" 无法匹配 ".file"，".*" 可以匹配

▶ 引号

- ▶ 单引号 ('): 单引号括起来的字符都作为普通字符

- ▶ 双引号 (" "): 除了 \$、\、'、"，其它都是普通字符。
\$ 后面的指定的变量的值进行替换。 \ 转义

▶ 其他

- ▶ &

- ▶ ;

- ▶ >

- ▶ <

shell 中的特殊字符 II

- ▶ >>
- ▶ \
- ▶ |
- ▶ ||
- ▶ ~

shell 中的环境变量 I

- ▶ 什么是环境变量？定义运行环境的变量，可以影响用户在 Linux 系统中的行为
 - ▶ PATH
 - ▶ HOME
 - ▶ HISTSIZE (history)
 - ▶ LOGNAME
 - ▶ HOSTNAME
 - ▶ SHELL
 - ▶ LANG/LANGUAGE
 - ▶ PS1 (命令符号)
 - ▶ MAIL

 - ▶ JAVA_HOME; CLASSPATH

shell 中的环境变量 II

- ▶ `MANPATH; INFOPATH`
- ▶ `export MYPATH="/home/zhli/teach/linux"`
- ▶ `echo $MYPATH`
- ▶ 可以用 `env` 察看所有的环境变量
- ▶ 使用 `set` 显示所有本机定义的 shell 变量（注意不是环境变量）
- ▶ `unset` 可以清除变量（只针对环境变量吗？）
- ▶ `readonly MYPATH`（只读属性）
- ▶ 用户如何定制自己的环境变量？
 - ▶ `~/.profile` 或 `~/.bashrc`
 - ▶ 修改完后重新登录或 `source` 指定配置文件即可。

vi 编辑器 I

- ▶ 进入和退出

- ▶ `:q` `:q!` `:w` `:wq` `:x`

- ▶ vi 操作模式

- ▶ 两种模式：插入模式和命令模式。

- ▶ 由命令模式 -> 插入模式的方式

- ▶ `append: a A`

- ▶ `insert: i I`

- ▶ `open a new line: o O`

- ▶ vi 基本编辑

- ▶ `dd`

- ▶ `x X`

- ▶ `r R`

- ▶ `s` 删除光标处所在字元，并进入插入模式

vi 编辑器 II

- ▶ S 删除光标所处的行？并进入插入模式

- ▶ 光标移动：

- ▶ h j k l

- ▶ 0 \$ ^

- ▶ ctrl+f ctrl+b

- ▶ H M L B W E

- ▶ n- n+ (n 表示任一数字)

- ▶ fx Fx tx Tx (x 表示任一字符)

- ▶ ; 配合 fx&tx 使用，重复一次

- ▶ , 配合 fx&tx 使用，反方向重复一次

- ▶ / forward 查找

- ▶ ? backward 查找

- ▶ n 配合 /&? 使用，重复一次

- ▶ N 反方向重复

- ▶ 行编辑命令

vi 编辑器 IIII

- ▶ `d y p c` (可以和下面的字母组合使用)
- ▶ `e`: 光标所在位置到下一个词的前一个字符
- ▶ `w`: 光标所在位置到下一个词的第一个字符
- ▶ `b`: 光标所在位置到上一个词的第一个字符
- ▶ `$ 0 ^`
- ▶ `)`: 光标所在位置到下一个句子的第一个字符
- ▶ `(`: 光标所在位置到该句子的第一个字符
- ▶ `}`: 光标所在位置到该段落的最后一个字符
- ▶ `{`: 光标所在位置到该段落的第一个字符
- ▶ `yy 3yy dd cc`
- ▶ 选择文本: `ctrl+v v V` 结合光标移动
- ▶ 设置 `vi` 的环境变量
 - ▶ `:set` 可以设置 `vi` 的环境变量; `:set all` 可以察看所有的环境变量

vi 编辑器 IV

- ▶ `:set number/nonumber`
- ▶ `:set readonly`
- ▶ `:set showmode/noshowmode`

使用图形环境 X Window I

- ▶ X Window 是工作在 Unix/Linux 上的桌面系统。
- ▶ X Server, X Client, 通信管道
- ▶ 常见桌面环境: GNOME; KDE, CDE

常用软件工具 I

▶ 系统工具

- ▶ tar

- ▶ 软件包管理: RPM (Red Hat Package Manager), yum, .rpm; dpkg (Package Manager for Debian), apt-get, .deb

▶ 网络工具

- ▶ Web 浏览器: Mozilla Firefox

- ▶ 下载工具, ftp (FileZilla), ssh (PuTTY)

- ▶ 多媒体, adobe flash player; MPlayer

- ▶ 图形工具: 截图 (Ksnapshot); 图形处理 (Kpaint, Gimp)

- ▶ openoffice; libreoffice

管理用户 I

- ▶ `useradd userdel usermod passwd`
- ▶ `/etc/passwd /etc/shadow`

管理组 I

- ▶ `groupadd groupdel`
- ▶ `usermod -G group1 user1`
- ▶ `gpasswd -a user1 group1`
- ▶ `/etc/group`