STAT:5400 (22S:166)
Computing in Statistics

Introduction

Lecture 2
August 24, 2016

Kate Cowles
374 SH, 335-0727
kate-cowles@uiowa.edu

Structure of Linux environment: like an upside-down tree

- directories
- subdirectories
- files

Home directories

- smart for multi-user system to provide each user private place to store files
  - called "home directory" in Linux
- you are only *regular* user who can access files in your home directory
  - *system administrator* also can
  - we will see later how to use `chmod` command to give access to other people
- when you log in to Linux, you automatically end up with your home directory as the current directory

The "shell"

- shell: program that sends commands typed at the keyboard to the operating system to perform
- several shells available in UNIX and Linux
  - C shell (csh)
    - developed for Berkeley UNIX
  - T C shell (tcsh)
    - default in our Linux network
    - enhanced version of C shell
  - Bourne Shell (sh) and its extensions
    - Bourne Again Shell (bash)
    - highly programmable Korn shell (ksh)
- you can run a shell other than default shell simply by typing its name
  - e.g., if you start up in tcsh and wish to switch to ksh, type `ksh`, and a Korn shell will start up

shell command to list the contents of a directory

- `ls` command lists the contents of a Linux directory
  - unless options are used, `ls` gives just names of files and subdirectories
  - all options start with an "-"
  - example: `ls -l` gives "long listing"
  - here's part of its output for the contents of my directory for this course

```
drwx------  2 kcowles faculty   8192 Aug 20 14:09 hw
drwx------  2 kcowles faculty     96 Aug 20 14:21 labs2005
drwx------  2 kcowles faculty     96 Aug 11 14:43 lects
drwx------  2 kcowles faculty   8192 Aug 22 13:24 lects2005
-rw-------  1 kcowles faculty   1013 Jul 30 16:24 online.resources
-rw-------  1 kcowles faculty      8 Aug 22 12:46 questionnaire.aux
-rw-------  1 kcowles faculty   1492 Aug 22 12:46 questionnaire.dvi
-rw-------  1 kcowles faculty   5348 Aug 22 12:46 questionnaire.log
-rw-------  1 kcowles faculty   1497 Aug 22 12:46 questionnaire.tex
```

  - first character of an entry is almost always either
    - ∗ "-" — entry is a file
    - ∗ "d" — entry is a directory

  - next nine characters show the security mode (explained in next section)
  - username of owner of file
  - group owner of file
    - ∗ subset of people with accounts on the system to which the owner belongs
    - ∗ we may choose to give them special access to this file
  - size of file in bytes
  - date and time when file was last modified
  - filename

File security in Linux

- many PC users unfamiliar with the idea of file security
- Windows didn't use to need security because it is a single-user OS
- Linux is a multi-user OS, so it has security to prevent people from accessing each other's confidential files
- Linux computers are not vulnerable to viruses and worms that infect a computer by altering files or writing new ones

Security characters in long listing

- three sets of three characters
  - first set for user
  - second set for group
  - third set for other (everyone in the world)
- letters r, w, and x mean different operations one can perform on a file
  - r — you can read the file's contents
  - w — you can write or change the file's contents
  - x — you can execute the file (given only for programs and directories)
- 9 security characters as a group are called the *security mode* of the file

Changing file permissions: “chmod”

- `chmod` stands for “change mode”
- first argument specifies which set(s) (user, group, or other) of 3 characters you want to modify
- second argument is a `+` (if you wish to add permissions) or a `-` (if you want to take them away
- third argument is which permission(s) you want to change

- examples:
  chmod g+r questionnaire.tex
  - would give the group (in this case `faculty` read permission for this file

  chmod a+rx hw
  - would give “all” (user, group, and other) read and execute permission for the directory `hw`

Using “wildcards” to save typing

- wildcard allows you to specify more than one file in one command
- `*` matches any number of characters
- to execute command on all files in the current directory, specify `*` as the filename; e.g.

  ```
  chmod o-rx *
  ```
- to execute a command on all the files with filename `questionnaire` regardless of their extension, use `questionnaire.*` as the filename
- other filename character is `?`, which matches exactly 1 character

Limitation of Linux security

- to be able to give special access permissions to a certain group of people, must get system administrator to create a group containing those people
- impossible to give different sets of access permissions to a different groups of people because any file or directory can have only one group owner
- fix will be implemented in future version of Linux

Learning which groups you are in

- enter

  ```
  groups
  ```

Printing

- basic Linux print commands
  - `lpr` — send file to printer
  - `lpq` — display print queue
  - `lprm` — remove file from print queue
- examples:
  - `lpr thoughts`
    sends file named “thoughts” to default printer
  - `lpr -Pp346 thoughts`
    sends same file to a printer called `p346`
    names of printers are determined by system administraor

lpq

- `lpq` is used to check status of print jobs
- `lpq` without any arguments produces status of all print jobs in default print queue

```
[kcowles@p-lnx402 ~]$ lpq
p374 is ready
no entries
```

```
[kcowles@p-lnx402 lects2005]$ lpr root.ps
[kcowles@p-lnx402 lects2005]$ lpr lect01.ps
[kcowles@p-lnx402 lects2005]$ lpq
p374 is ready and printing
Rank    Owner   Job     File(s)                         Total Siz
active  kcowles 127     root.ps                         99328 byt
1st     kcowles 128     lect01.ps                       116736 by
```

- can use `-P` option to get status of print jobs in a different queue

```
lpq -P p346
p346 is ready
no entries
```

Canceling print jobs with lprm

- `lprm` is short for “line printer remove”
- use the *job number* to specify which one to cancel

```
[kcowles@p-lnx402 lects2005]$ lprm 127
[kcowles@p-lnx402 lects2005]$ lpq
p374 is ready and printing
Rank    Owner   Job     File(s)                         Total Siz
active  kcowles 128     lect01.ps                       116736 by
```

Canceling other kinds of jobs

- Use `ps` to list running jobs
- `u` option restricts to your own (the user’s) jobs

```
[kcowles@p-lnx402 ~]$ ps u
USER       PID %CPU %MEM   VSZ  RSS TTY      STAT START   TIME COMMAND
kcowles   4277  0.0  0.0 44720 1468 pts/0    Ss   09:01   0:00 -csh
kcowles   4299  0.0  0.2 63740 5860 pts/0    S+   09:01   0:00 pine
kcowles   4307  0.0  0.0 44720 1476 pts/1    Ss   09:10   0:00 -csh
kcowles   4374  0.0  0.1 58936 3096 pts/1    S+   09:36   0:01 vim lect2
kcowles   4575  2.0  0.0 44508 1424 pts/2    Ss   10:05   0:00 -csh
kcowles   4596  0.0  0.0  4580  708 pts/2    R+   10:05   0:00 ps u
```

- `pid` is the process i.d.
- use `kill <pid>` to cancel a process that you are unable to terminate in a normal way
  - `kill` tries to allow the process to clean up after itself as it shuts down
  - sometimes a process is so out-of-control that it ignores the signal to die
  - `kill -9 <pid>` is command to “kill immediately”

Using man pages to learn about commands

- entire contents of Linux manual is online
- `man` command
- use with `-k` option to find all commands that contain a word of interest in their short description in the manual; e.g.
  - `man -k find`