

C++ Programming on Linux

CS 2308
Spring 2013

Jill Seaman

1

What is Linux?

- an operating system
- Unix-like
- Open source
- created in 1992 by Linus Torvalds
- can be installed on a wide variety of hardware
 - mobile phones
 - desktop/laptop computers (PCs)
 - mainframes
 - supercomputers

2

Using Linux

- Common user interfaces:
 - * Command line (\$ prompt)
 - User enters commands at the prompt
 - results displayed on following lines
 - * X Window System - graphical interface
 - Similar to MS Windows or Mac OS X
 - KDE: K Desktop Environment

3

Accessing Linux at Texas State

- Derr 231: Texas State CS Dept Linux Lab
- Requires a CS Dept Linux account
 - * use your netID and password
 - * <http://cs.txstate.edu/labs/LinuxAccounts.php>
- The lab machines start up in KDE (windows).
- To open a terminal window :
 - * Click on the kaleidoscope, select: System Tools > Terminal

4

Linux File System

- Common hierarchical system.
- Root directory of the system: /
- Directories can contain:
 - * Files
 - * Other Directories
- Each user has a home directory:
 - * /home/Students/js108

5

Basic Shell Commands

- To display the manual page for a linux command

```
[...]$man <command-name>
```

- To display a list of options that work with the command:

```
[...]$<command-name> --help
```

- To clear the screen

```
[...]$clear
```

6

Basic Shell Commands

- To display the current (working) directory

```
[...]$pwd  
/home/Students/js108
```

- To display a listing of the contents of the current directory

```
[...]$ls
```

- To see more info about the files in the directory

```
[...]$ls -l
```

7

Basic Shell Commands

- To display all the files, including the hidden ones

```
[...]$ls -a
```

- To display a listing of the contents of some other directory

```
[...]$ls /etc
```

- To change the current directory

```
[...]$cd /etc
```

8

Basic Shell Commands

- To create a new directory (in the current one)

```
[...]$mkdir projects
```

- To remove a directory (must be empty)

```
[...]$rmdir projects
```

- Some shortcuts

- * ~ is your home directory

- * .. is the parent directory

- * . is the current directory

```
[...]$cd ~/projects  
[...]$cd ..
```

9

Basic File Editing

- To use the nano editor to create a file and start editing it:

```
[...]$nano myFile.txt
```

- This begins an editor within the terminal window.
- You can type to enter text, navigate with the arrow keys, use the backspace/delete keys.
- Other commands, listed at bottom of window, are activated with the control key and a letter.

10

Basic File Editing

- When finished, press CTRL-X
- Follow the prompt: press Y to save

- You may also use other editors:

- * vim

- * emacs

- All of these editors run from within the terminal window.

11

More Editing Options

- There is also a text editor in KDE (the graphical interface)
- Find it in the menu system
- Files you create and save in the KDE text editor are stored to your linux home directory and can be accessed using the shell commands.
- Note: there is also a Firefox browser on the X Windows system.

12

Basic Shell Commands

Files

- To view the contents of a file (pick one)

```
[...] $more myFile.txt
[...] $less myFile.txt
[...] $cat myFile.txt
```

- To make a copy of a file

```
[...] $cp myFile.txt someFile.txt
[...] $cp myFile.txt ~/projects/anotherFile.txt
```

- To move or rename a file (or both)

```
[...] $mv myFile.txt ~/projects (keeps original name)
[...] $cd ~/projects
[...] $mv myFile.txt bFile.txt
```

13

Basic Shell Commands

Files

- To delete (remove) a file

```
[...] $rm myFile.txt
[...] $rm *.txt
```

- The file is gone, there is no trash can.

14

Compiling and Running C++ Programs

- Create a file containing a C++ program.

```
[...] $nano hello.cpp
```

- To compile the file using the gnu compiler:

```
[...] $g++ hello.cpp
```

(if you get compiler errors, fix in editor, run g++ again)

- To run the executable file:

```
[...] $./a.out (Not allowed to directly execute a file in the current directory)
```

15

Remote Access

from unix/linux shell

- Secure Shell allows you to securely connect to a remote computer within a command shell.

```
[...] $ssh js108@hercules.cs.txstate.edu
```

(You will be asked to enter your password)

- Current directory will be your home directory
- Can use all the standard linux commands
- Type exit to logout of the secure shell session

```
[...] $exit
```

16

Secure File Transfer

from unix/linux shell

- Secure FTP allows you to securely connect to a remote computer to transfer files.

```
[...]$sftp js108@hercules.cs.txstate.edu
```

(You will be asked to enter your password)

- `ls` will display files on remote machine
- use `get` to transfer a file to your local machine

```
sftp>get myFile.txt
```

- Type `exit` to logout of the secure ftp session

```
sftp>exit
```

17

Secure Shell on a Windows PC

- You can download the Secure Shell client (or puTTY) from the CS departmental download server

<http://downloads.cs.txstate.edu>

- Select `os` then `windows` then `remote_access`, then `SSHSecureShellClient-3.2.9.exe` (then install it).
- To run: double click on Secure Shell Client icon
- Click Quick Connect and enter a host machine:

`hercules.cs.txstate.edu`

athena
zeus
eros
hercules

- Enter username and password.

18

Secure FTP on a Windows PC

- If you are currently connected remotely thru the Secure Shell client and would like to transfer files with Secure FTP:
- click the Windows menu,
- then New File Transfer

- Note: you can also use Filezilla, a free app for windows or mac.

<http://filezilla-project.org>

19