Linux Commands II

Dr. Hyrum Carroll

August 30, 2016

Linux Commands I (review)

- `man` - show manual for a command, example: '`man ls`' (press '`q`' to exit the man page)
- `ls` - list directory, similar to `dir` on windows, example: '`ls /etc`', use '`ls -l /etc`' to see more detail
- `cd` - change directory, example: '`cd /etc/`'
- `touch` - update the timestamp on a file, example: '`touch foobar`'
- `cp` - copy a file or directory, example: '`cp source dest`' if you want to copy a directory use the `-R` option for recursive: '`cp -R /source /dest`'
- `mv` - move a file, example: '`mv source dest`'
- `rm` - remove a file, example: '`rm somefile`' to remove a directory you may need the `-R` option, you can also use the `-f` option which tells it not to confirm each file: '`rm -Rf dir/`'
- `mkdir` - make a directory, example: '`mkdir foobar`'
- `rmdir` - remove an empty directory, example: '`rmdir foobar`'

Linux Commands II Exercise

Write a command to do each of the following:

1. Display the name of the current directory
2. Display the path of `mkdir`
3. Execute the last command that started with "`mak`"
4. Execute the third to last command used
5. Print to the screen the contents of all of the `.txt` files in the current directory
6. Display the filename of just the last file from the previous command
7. View the contents of the file `researchIdeas.txt` one screen full at a time

Linux Commands II (review)

- `cat` - dumps files to the screen with no page breaks
- `more` - displays a file with page breaks after the screen fills up
- `less` - displays a file on the screen
- `pwd` - print working directory - lets you know where you are in the directory structure
- `*` - a wild card character - represents any string; e.g., '`more *`' will display all the files in a directory
- up and down arrows - goes to previous or next command - a short cut to avoid typing long strings
- `!` $\{partial\}$ - executes the last command with the name that started with the string $\{partial\}$
- `!$` - the last parameter on the previous line
- `which` - locate a command

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network
- `scp` - secure copy - copy a file or files across a network

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network
- `scp` - secure copy - copy a file or files across a network
- `sftp` - secure file transfer - transfer files to and from a remote computer through the network

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network
- `scp` - secure copy - copy a file or files across a network
- `sftp` - secure file transfer - transfer files to and from a remote computer through the network
- `du` - disk usage - the amount of space files/directories occupy

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network
- `scp` - secure copy - copy a file or files across a network
- `sftp` - secure file transfer - transfer files to and from a remote computer through the network
- `du` - disk usage - the amount of space files/directories occupy
- `sort` - sort lines - alphanumeric or string numeric (with `-n`

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network
- `scp` - secure copy - copy a file or files across a network
- `sftp` - secure file transfer - transfer files to and from a remote computer through the network
- `du` - disk usage - the amount of space files/directories occupy
- `sort` - sort lines - alphanumeric or string numeric (with `-n`
- `uniq` - display only unique lines

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network
- `scp` - secure copy - copy a file or files across a network
- `sftp` - secure file transfer - transfer files to and from a remote computer through the network
- `du` - disk usage - the amount of space files/directories occupy
- `sort` - sort lines - alphanumeric or string numeric (with `-n`
- `uniq` - display only unique lines
- `echo` - write arguments to the standard output

Linux Commands III

- `ssh` - secure shell - log into a remote computer through the network
- `scp` - secure copy - copy a file or files across a network
- `sftp` - secure file transfer - transfer files to and from a remote computer through the network
- `du` - disk usage - the amount of space files/directories occupy
- `sort` - sort lines - alphanumeric or string numeric (with `-n`
- `uniq` - display only unique lines
- `echo` - write arguments to the standard output

Examples:

- `ssh username@hostname`
  `ssh abc9z@herschel.cs.mtsu.edu`
- Same syntax as cp with "`username@hostname:`"
  `scp index.html`
  `abc9z@herschel.cs.mtsu.edu:/public_html/.`
- `sftp`: `sftp username@hostname`
  `sftp abc9z@herschel.cs.mtsu.edu`