CLI Basics



Help and Documentation



Basic Help

- Cli commands are generally typed in this format: <command> <options> <arguments>
- Some commands have built in help "option", usually -h, or -help, or --help
- Comments in config and example configuration files
- For more detailed help read the manual

Example:

user@system: ls --help

```
Usage: ls [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of –cftuvSUX nor ––sort is specified.
Mandatory arguments to long options are mandatory for short options too.
                             do not ignore entries starting with .
  -a, --all
  -A, --almost-all
                            do not list implied . and ..
      --author
                             with -1, print the author of each file
                             print C-style escapes for nongraphic characters
  -b, --escape
      --block-size=SIZE
                             scale sizes by SIZE before printing them; e.g.,
                               '--block-size=M' prints sizes in units of
                               1,048,576 bytes; see SIZE format below
  -B, --ignore-backups
                             do not list implied entries ending with ~
                             with -lt: sort by, and show, ctime (time of last
```



Manual (man pages)

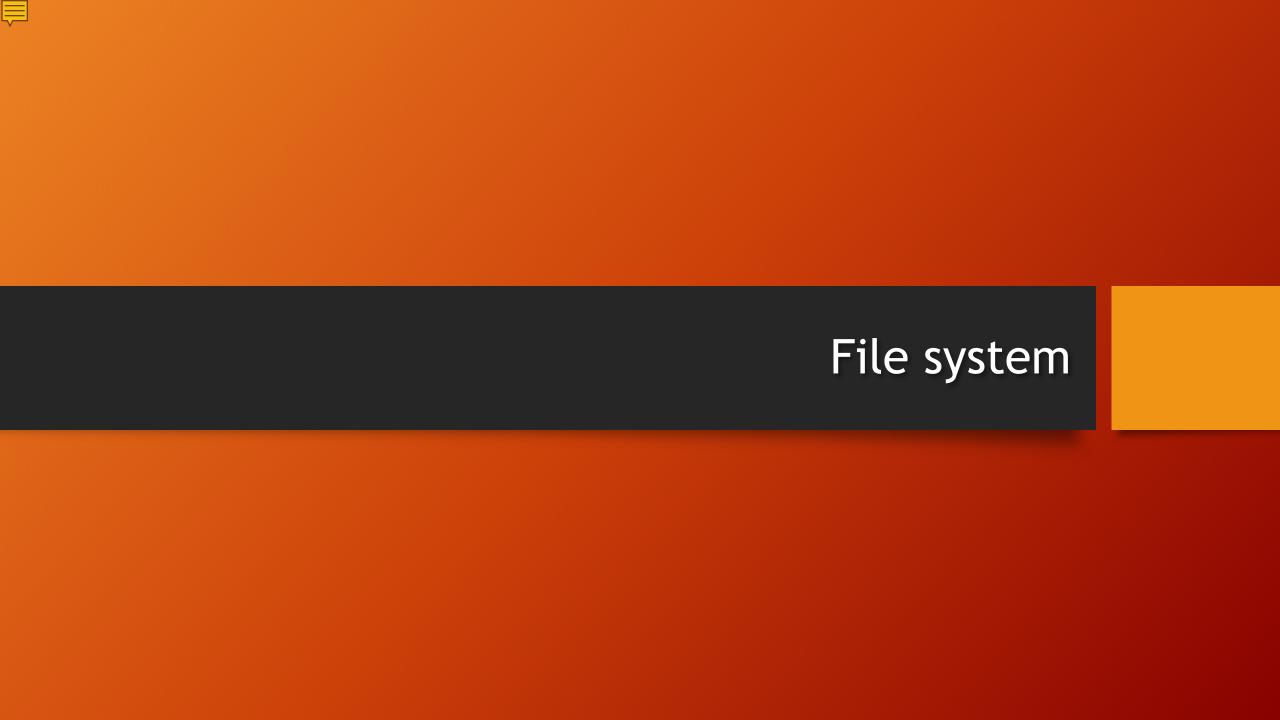
Section	Description
1	General commands
2	System calls
3	Library functions, covering in particular the C standard library
4	Special files (usually devices, those found in /dev) and drivers
5	File formats and conventions
6	Games and screensavers
7	Miscellanea
8	System administration commands and daemons

- Unix manuals were originally printed in multiple volumes and provided to customer
 - Broken down to sections
 - For example User commands are in section 1
- "man pages" are now software documentation included in the OS
 - Preferred way of looking up commands and documentation



Online Documentations

- Info was to be an easier to navigate version of man-pages
 - Try "info" command in your VM to see what it does
 - Since web and internet is widely available the project has stalled
- Many Unix/Linux vendors and GNU/Linux communities maintain web-based documentations for their distributions
 - Example: https://manpages.ubuntu.com
- Specific applications may have their own manual
 - Example: Ubuntu Server Guide at https://ubuntu.com/server/docs
 - Provides guides on how to setup various services, tools, and various configurations
 - Apache Documentation: http://httpd.apache.org/docs/





Files and paths

- Everything in the file system is under the "root" /
- typical types of files: regular file, directory, symbolic link
 - See: https://linuxconfig.org/identifying-file-types-in-linux
 - Remember: file types in Linux are NOT images, document, dll, etc...
 - A directory is just a special file
 - Contains a list of files that are inside the directory
- Two ways of addressing a file:
 - Absolute Path
 - /home/User/Documents/file1
 - Relative Path (relative to current working directory)
 - ./Documents/file1



File and Directory Permissions

- Read (r), Write (w), Execute (x) permissions can be specified for each file in the file system
- The r, w, and x can also be assigned for owner, group, and others on each file
- If your user account does not have permission to read, write, or execute a file, system will deny access.
- Root account has access to all files!
 - Limit the use of your root account, use your regular user for all regular tasks, do not "sudo" or "su" unnecessarily



Common commands

- List directory content: ls
- Change Directory: cd
- Make directory: mkdir
- Remove files and directories: rm
- Create a new empty file or change timestamp on existing file: touch
- Copy: cp
- Move: mv



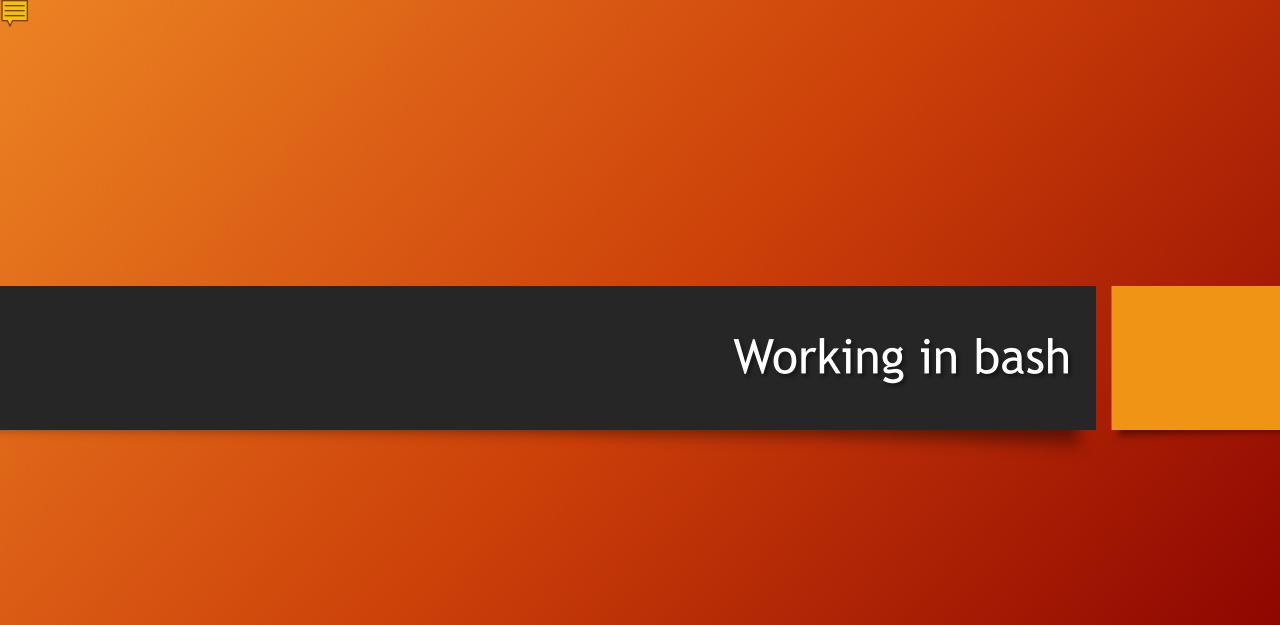
Navigation using shortcuts

- ~ (tilde) refers to user's home directory
 - This refers to a variable called \$HOME in bash
 - Example: cd ~
 - ~<username> to get to that users' home directory
- . (dot) refers to current directory
 - Example: cd ./Documents
- .. (dot dot) refers to one directory above
 - Example: cd ..



More file commands

- More and less!
- Cat
- Grep
- Echo





Bash Trickery!

- Tab key to auto-complete command, path, or filename
- Up and down arrow keys for command history
- Special characters:
 - * (asterisk) Matches zero or more characters
 - ls *.txt
 - ? (question mark) Matches any single character
 - ls ????
 - [] (square brackets) Matches a single character in a range
 - ls [a][0-9]*
 - \$ (dollar sign) Identifies a variable
 - ~ (tilde character) Represents the user home directory
 - cd ~/Documents

```
user@hostname:~$ touch a1.txt
user@hostname:~$ touch a874.txt
user@hostname:~$ ls [a-z][0-9]*
a1.txt a874.txt
user@hostname:~$ rm a*
user@hostname:~$ ls [a-z][0-9]*
ls: ... No such file or directory
```



Bash Trickery!

Quoting special characters

- \ (backslash): turns off the special meaning of its following character
 - echo \\$PATH outputs \$PATH and will ignore function of \$
- " (double quotes): cause most metacharacter special meanings to be ignored
 - echo "date"
- ' (single quotes): negate the translation of all special characters
 - Is '?' outputs actual filename matching "?"
- ` (backquote): enable command substitution to occur
 - echo `date` outputs the date



Command Redirection

- Redirecting input from or output to other sources such as file, network, device, etc. (other than Standard input or keyboard)
- redirection Symbols
 - < (take input from ...)
 - > (output to ...)
 - >> (output to and append to an existing file)
 - Pipe: (|) takes output of a command and connects it to the input of another command
 - ex. ls -la | more