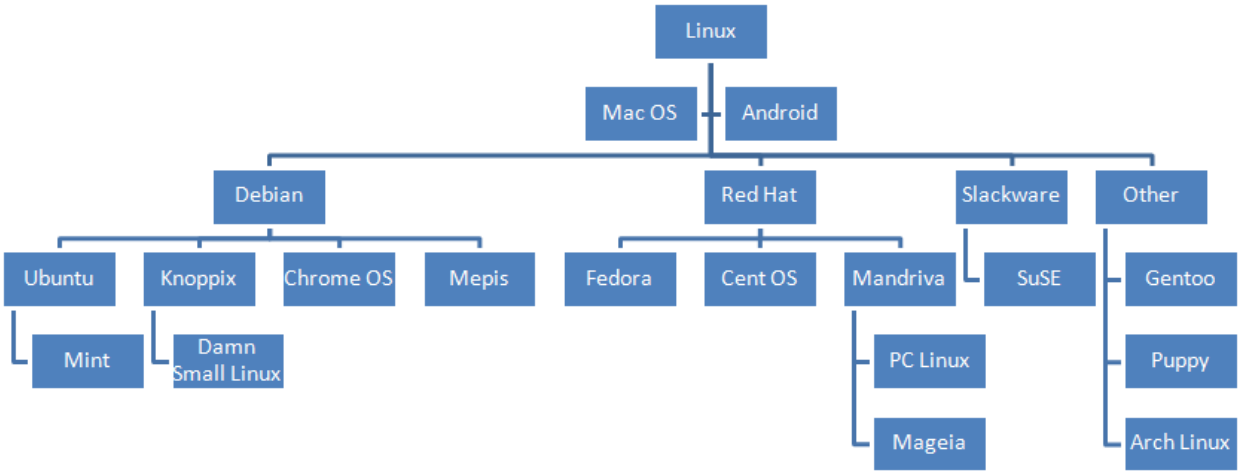
Distribution Basics:



Common Network Ports & Matching Linux Server Application:

Port Number	Protocol	Server Application
22	SSH	OpenSSH
23	TELNET	telnetd
25	SMTP	PostFix, Sendmail
53	DNS	BIND, named, unbound
67	BOOTP	dnsmasq, dhcpd
80	HTTP	Apache, nginx
443	HTTPS	Apache, nginx

Other Important Server Apps:

File Server

- NFS (Network File Server) – common on many UNIX-style operating systems
- Samba/CIFS – Common Internet File System, open-source implementation of Microsoft's Server Message Block protocol

Database Server

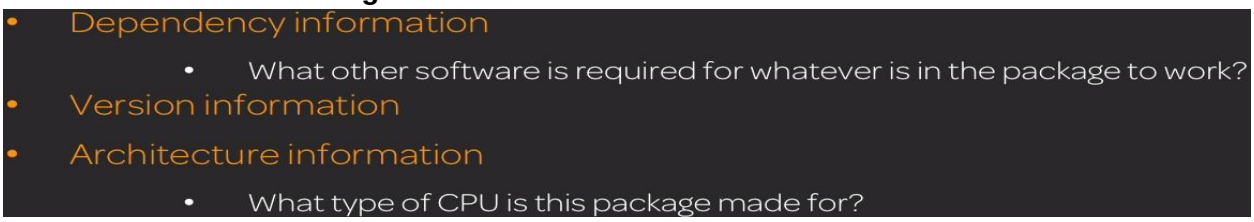
- MySQL/MariaDB – MySQL (now owned by Oracle) is one of the world's most popular open-source database management servers. MariaDB is a clone of MySQL not owned by Oracle
- PostgreSQL – a bit more advanced open-source database management system

CUPS

- Common Unix Printing System – open-source print server, primarily developed by Apple

What is Package Management?:

What Constitutes a Package?:



Package Management:

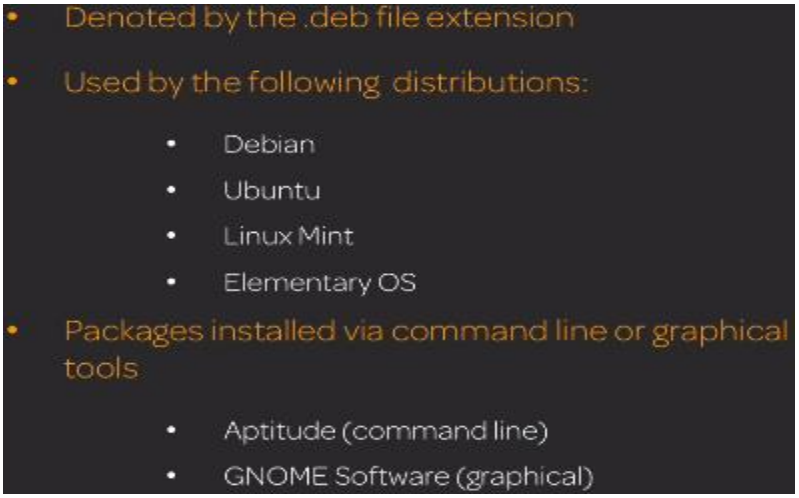
Installing from Source:

- Slack
- Gentoo

Pre-Built Packages:

- RPM

- .deb



yum (RHEL):

- yum install <package name> === installs a package from a repository
- yum update <package name> === checks & prompts for updates
- yum remove <package name> === uninstalls package
- yum search <repo> === queries a repo for a package

rpm (RHEL):

- rpm -ivh <rpm file name.rpm> === installs an rpm file
- rpm -qi <package name> === queries an rpm file for info
- rpm -U <package name> === updates an rpm file
- rpm -e <package name> === erases an rpm file

APT (Debian):

- apt install <package> === install package from a repo
- apt update <package> === checks & prompts for updates
- apt remove <package> === uninstalls a package
- apt-cache search === queries local apt cache for a package

dpkg (Debian):

- dpkg -i <package> === installs
- dpkg -l <package> === list all installed packages
- dpkg -r <package> === uninstalls

CLI Syntax: ls

ls - list the contents of a directory and their attributes

ls -l = long listing, showing file and directory ownership, permissions, and sizes

ls -lh = the 'h' switch shows size in human readable format, must be used with 'l' switch

ls -a = show all files and folders, including hidden ones

ls -R = list directories recursively

ls -S = sort files by size with the largest at the top

ls -t = sort by last time modifed displaying the newest first, most useful with the -l switch

whoami - display your currently logged in user

su - substitute user, change to another user account on the system

exit - leave a shell environment that you are logged in to

init 6 - legacy command for rebooting a system

init 0 - legacy command for shutting a system down

halt, poweroff - shuts down (halts) a system

shutdown - can be used to poweroff, reboot or stop a pending shutdown request

top - interactively display top running processes on a system

CLI Syntax: $PATH

env - list all of the environment variables set for the current shell environment

The **PATH** environment variable contains a llist of all of the directories that Bash will look in for applications or scripts to run.

echo = print what follows to the screen

echo $PATH = print the contents of the PATH environment variable to the screen

Remember that Linux is a case sensitive operating system.

So, **ls** is not the same as **LS**.

CLI Syntax: uname

uname - display the name of the system kernel

uname -r - display the kernel release number

uname -v - display the kernel build version

uname -m -display the machine type

uname -o - display the name of the operating system

uname -a - display all information that uname can show

CLI Syntax: Basic Commands

CLI Syntax: cd & pwd

pwd - print working directory, the current directory that you are in

cd - change directory to path specified. i.e. cd /path/to/folder if entered by itself with no path, you will change back to the home directory of the user you are currently logged in with

cd .. - go up one directory

cd - -return to last directory

cd ~ - change to the home directory of the currently logged in user

CLI Syntax: bash history & completion

.bash_history - hidden file within the home directory that contains a log of commands entered at the Bash prompt

HISTFILESIZE - environment variable that specifies how many lines of history to keep

HISTCONTROL - environment variable that modifies Bash's history behavior

history - command that prints out commands saved in .bash_history with each command numbered

!<history number> - re-runs command from .bash_history

TAB key is your friend when it comes to command completion and having long file and directory names autocompleted at the Bash prompt for you

Shell Config Files

/etc/bashrc - system-wide functions and aliases

/etc/profile - system-wide environment and startup programs, used during a login shell

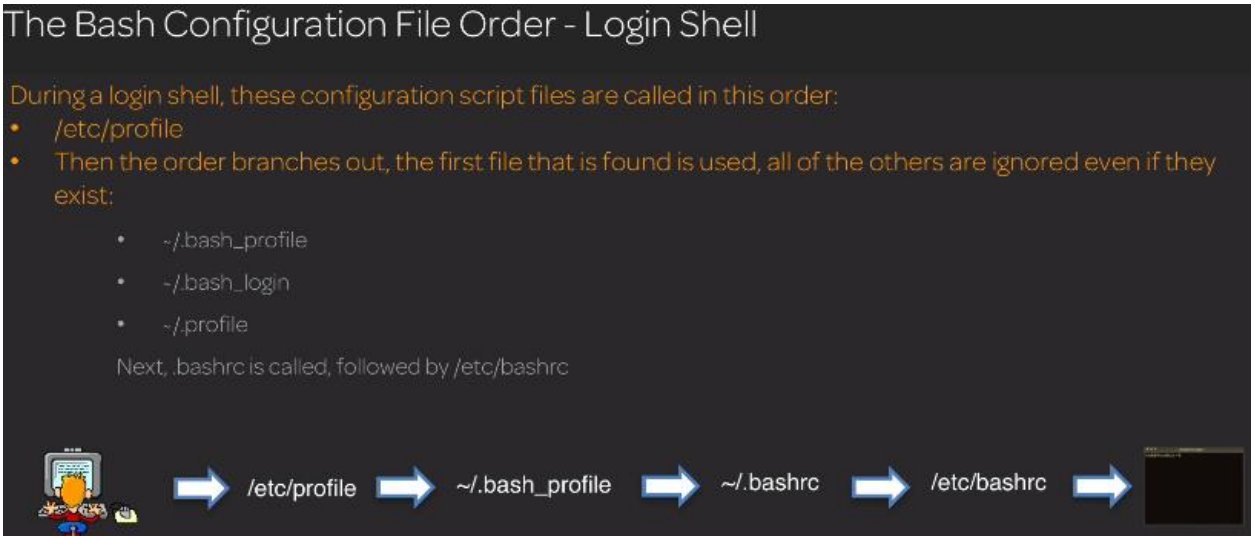
/etc/profile.d/ - location of extra environment setup scripts

The following files are in the home directory of the user (note that not all distributions will use all of these files):

.bash_profile - used to set user specific shell environment preferences

.bash_logout - ran when the user logs out of a login shell, not a terminal, does not exist on every system

.bashrc - non-login file that stores user specific functions and aliases



Shell Variables:

env - lists out environment variables of the currently logged in shell

echo $VARIABLE - prints the value of VARIABLE to the screen

set - lists out all environment variables in alphabetical order

VARIABLENAME=value - format for declaring a new variable in Bash

export VARIABLE - exports variable and its value to other shells

Globbing

***** - matches zero or more characters

? - matches any single character

[abc] - matches any one of the characters in the list, case sensitive

[^abc] - matches any one character except those in the list, case sensitive

[0-9] - matches a range of numbers

Locate, Find, Whereis

Quoting

" " - double quotes, contains strings and any variables or commands within them get evaluated or acted on

' ' - single quotes, anything within these gets treated literally, disables any special character functionality

**** - backslash, escape character, disables any special character functionality that immediately follows it

Quotes around spaces or an escape character preceding a space will be treated literally.

Formatting Commands

ls -l Documents/
Command - what to do?
Options - how to do it?
Arguments - what to do it on?

locate -searches a local database of files and folders looking for items that match the search criteria
locate cd

find - searches the file system for files that match the search criteria
find /path/to/folder -name file

When using the find command to search for part of a file name, use globbing within single quotes:
find /path/to/folder -name '*file*'

whereis - locates binary, source and/or manual pages for a command

Man:

man - manual pages command, invoked by entering:
man <command>

whatis - Command that lists summaries and related man pages based on search term, invoked by entering:
whatis <command>
Same results can be obtained by:
man -f <command>

apropos - command that searches man page for appearances of the keyword provided, invoked by entering:
apropos <keyword>
Same results can be obtained by:
man -k <keyword>

Arrow keys and vi key bindings can be used to navigate the man pages. Pressing the 'q' key will exit the man page.

Info

info - information utility command, invoked by entering:
info <command>

Arrow keys can be used to navigate the info pages. Placing the cursor over a node link and pressing 'enter' will take you to the selected node page. Pressing the 'q' key will exit the info utility.

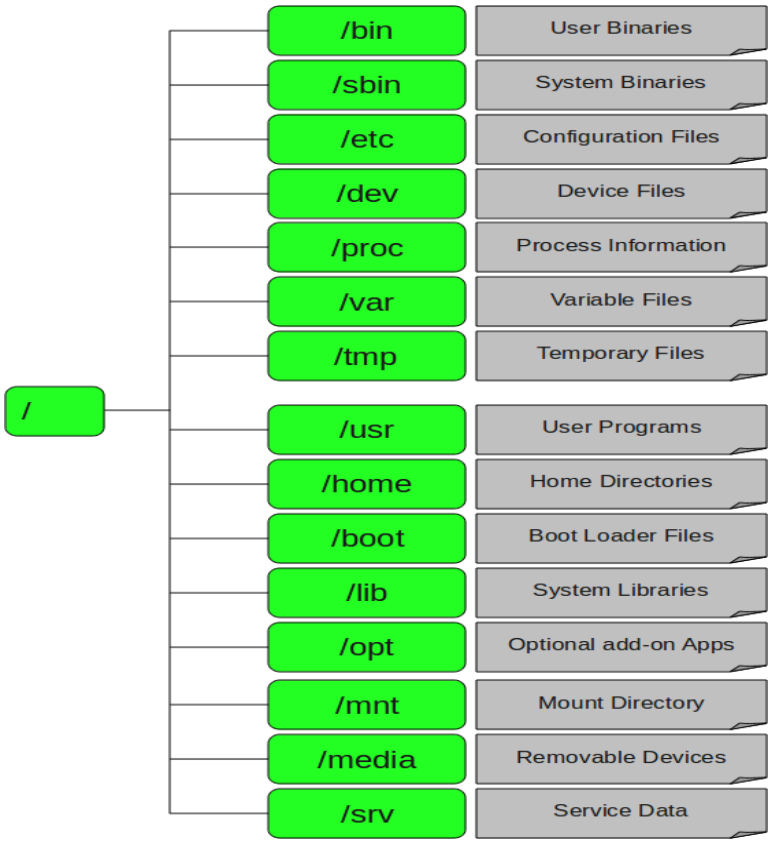
Package Local Documentation

Common local document directories (varies by distribution):

/usr/doc/packagename
/usr/share/doc/packagename
/usr/share/doc/packages/packagename

Files here are in plain text, and can be regular text files or html files that can be viewed in a web browser.

Linux File system



CLI: Files & Directories

mkdir - make a new directory
-p = make a parent directory along with a subdirectory

rmdir - remove an empty directory

touch - create an empty file or update a file's timestamp

cp - copy a file or folder
-R = copy a folder recursively
-v = verbose, display what the copy command is doing

mv - move or rename a file or folder

rm - remove a file or folder
-r = recursively remoe a folder and its contents

CLI: Archives & Compression (TAR)

tar - manipulate archive files
-c = create a new archive
-z = pass the archive through gzip compression
-j = pass the archive through bzip2 compression
-f = file name of archive to create
-x = extract an archive
-v = verbose output

zip - create a new compressed file
-r = recursively create a compressed file of directory and its contents

unzip - extract a zip archive

gzip - create a gzip archive

gunzip - extract a gzip archive

bzip2 - create a bzip2 archive

bunzip2 - extract a bzip2 archive

CLI: Viewing Text

less - view a text file with the ability to scroll through the pages of the file

head - view the first ten lines of a file
-n <number> = view the first <number> lines of a file specified

tail - view the last ten lines of a file
-n <number> = view the last <number> lines of a file specified
-f = follow the text file as new data is written to it in real time

CLI: Analyzing Text

cut - remove text from file and print specified fields to screen
-d = specified delimiter to use
-f = specifies which field to print

sort - sorts content of file alphabeticaly based on first character in file
-n = sorts content of file numerically

wc - word count, prints number of lines, words and characters in file
-l = print number of lines in file
-w = print number of words in file

'**>**' - redirects standard output to new location, if output goes to file replaces contents of file with output from stdout

'**>>**' - redirects standard output to new location, appends stdout to file

CLI: Pipes & Regular Expressions

grep - show the lines in a file that match a given pattern
-i = perform a case-insensitive search
-v = return lines that do not contain the pattern
-r = perform a recursive search

| - pipe character, used to send output of one command as input to another command
e.g.:
command1 | command2

Regular Expressions:
^ = Search the beginning of a line
$ = Search the end of a line
. = Stands in for a single character
[abc] = Search for specified characters
[^abc] = Search for other characters except for these
***** = Match zero or more of the preceding characters or expression

Nano (CLI text editor)

- nano <desired file name> === create new file
- ^ === control key
- Ctrl + o === save , Ctrl + x === save & exit
- Alt + 6 === copy a line of text
- Ctrl + U === paste

- Ctrl + T === spellcheck

vi/vim CLI Text Editor:

vi = One of the original text editors for UNIX-like operating systems. Installed by default on most Linux distributions

vim = Successor to the vi text editor. Offers the same functionality plus extra features.

i - insert text under cursor

A - append text at end of line

u - undo last change in file

dw - delete word under cursor

dd - delete whole line under cursor

:w - writes the file out to disk

:wq - write the file to disk then quit

:q! - quit but do not write the file to disk

vimtutor - built in tutorial on using vim, from beginner to advanced

Bash Scripting Part 1

#!/bin/bash = the "shebang", the first line in a bash script that tells bash what scripting language is being used

= begins a comment, a line that is ignored in the script but acts as documentation for someone viewing the script contents

Basic 'if' statement:

```
if [ something ]
then
   do this thing
fi
```

Basic 'if else' statement:

```
if [ something ]
then
   do this thing
else
   do this other thing instead
fi
```

Bash Scripting Part 2

basic for loop setup:

```
for variable in item1 item2 item3
do
   command to exectute on $variable
done
```

For various conditionals that can be used with an 'if' statement:
man test

For complete bash manual:
man bash

Hardware Commands

cat /proc/cpuinfo = view the cpuinfo file to gather details on the processor

free = view RAM statistics for the system
-m = show output in MB
-g = show output in GB

dmidecode = show details about the motherboard, BIOS, processor and RAM on a system

lsblk = view all block devices (such as hard disks) attached to the system

df = view free disk space on a hard disk
-h = show output in human readable format

top = show statistics on the processor, RAM, and running processes

Processes

ps = lists the processes currently running on a system
-u = list the processes for a specific user
-e = list all processes running on the system
-H = list all processes with indented output, showing the hierarchy
-f = full format listing, including command arguments

top = show statistics on the processor, RAM, and running processes

man proc

man signal

Networking in Linux

View your IP address configuration settings:
ifconfig
ip addr show

127.0.0.1 = local loopback address

View your default route (gateway) setting:
ip route show
route
netstat -r

Test DNS host name resolution:
host
e.g.: **host google.com**

Test network connectivity:
ping
e.g.: **ping -c 3 linuxacademy.com**

/etc/hosts = local name resolution file for the local host and for small networks

/etc/resolv.conf = file that contains the IP address(es) of DNS servers the system should contact

Linux User Accounts:

who = see who is logged into the system

w = see who is logged into the system with more details

id = view user and group ID's of a specified user, command by itself shows ID's of current user

sudo = execute a command as another user

/etc/passwd = primary configuration file for all users on a system

/etc/group = primary configuration file for groups on a system

man 5 passwd
man 5 group

Adding Accounts & Groups:

groupadd = add a new group to the system

useradd = add new users to a system

passwd = set a password for a user

/etc/default/useradd = defines some default behavior for the useradd command

/etc/skel = contents within this directory are copied into home directories of newly created users

/etc/shadow = primary configuration file for all encrypted passwords for users on a system

man 5 shadow

File & Directory Permissions

Permissions are broken down into three categories:
user group other/world

Symbolic Permissions:
r = read permission
w = write permission
x = execute permission
- = no permission

Octal Permissions:
4 = read permission
2 = write permission
1 = execute permission
0 = no permission

Examples:

```
|U | G| O|
rwxrwxr-x  =   775
rw-r--r--  =   644
```

chown = change ownership of a file or directory

chmod = change mode of a file or directory, effecting the permissions

Symbolic Links

ln - create a link to files or directories

ln -s = create a symbolic link
e.g.
ln -s myfile mylink

unlink - remove a link from a file or directory

Special Files & Folders, and the Sticky Bit

/var/tmp - contains files that **do not** get deleted on reboot

/tmp - contains temporary files that **do** get deleted on reboot

sticky bit - a permission that only allows users that create their own files and folder can delete theirs and not another users'

Apply the sticky bit to a folder:
chmod o+t /path/to/directory
chmod 1777 /path/to/directory