File System in Linux

Hierarchy

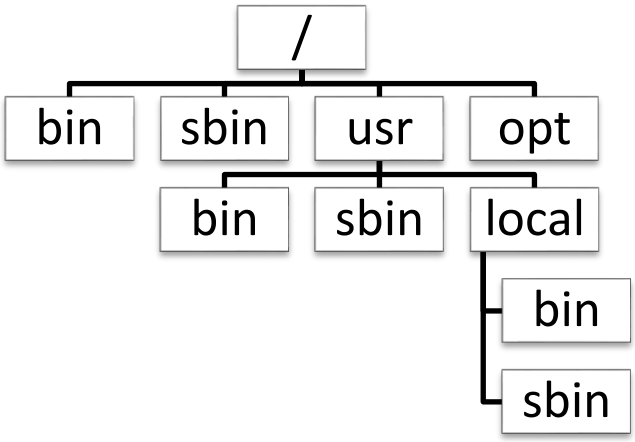
HIERARCHY OBJECTIVES:

Recognize:

- what specific folders are used for
- which folders are normal.

HIERARCHY :: DIRECTORY STRUCTURE

- All files and directories appear under the root directory, aka / regardless of how manylocal drives or network mountsare on the file system.
- Basedon FHStheLinuxFile Hierarchy Standard.
- Tools:

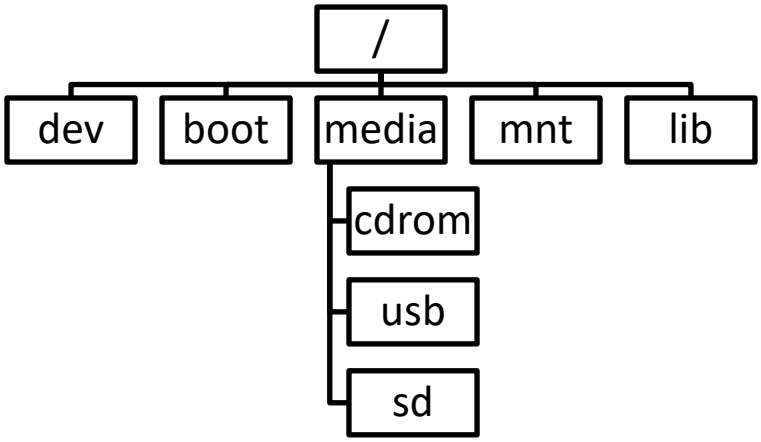


bin: binaries

sbin: System binaries

opt: Optional 3rd party tools

System:

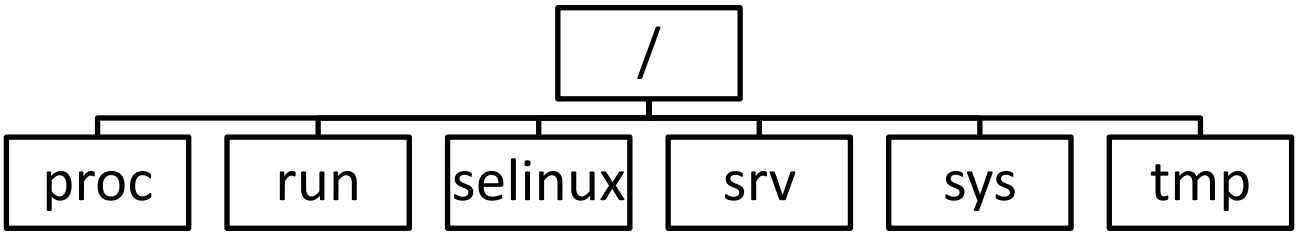


dev: devices (hardware)

boot: bootloader,kernels media: removablemedia.

mnt:temporarymounts lib: library *.so files

System:



proc: virtual file system, process & kernel info

run: run time variable data

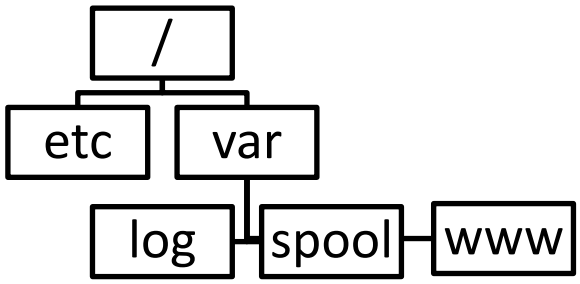
selinux: virtual fs for mandatory access control data

srv: served data (tftp)

sys: virtual file system, os & hardware info

tmp: temporary files, deleted upon reboot

System:



etc: configuration files

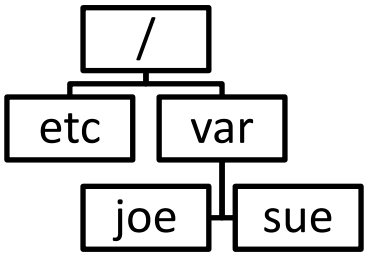
var: variable files

var/log: logs

var/spool: spooler, schedulers, mail

var/www:htmlfiles for web server

Home:



root: root user's home directory

home: regular user's home directories

Finding Files:

Finding objectives:

Learn how to	
find the full path of a tool	which, whereis
find a file anywhere	locate, find

Finding Files : which & whereis

- Find thelocationof a toolbased on your$PATHvariable.

```
# which cat
```

Only the first match in your $PATH will show which is what your shell would execute.

```
# whereis cat
```

- Another tool, less likely to be installed though:
  - Whereis shows all matches.
  - Check the options.

Finding Files: locate

- Find any file anywhere on the file system based on an index database, refreshed with updatedb.

```
# locate cat
```

Returns every file with *cat* in its name.

```
# locate -r '.*/cat$'
```

Regular expression for "something/cat."

- What does the following do?

```
# locate -i Index.HTml -L -l1 -w /var/www
```

– Help
```
# locate -h
```

– Statistics
```
# locate -S
```

Finding Files: find

- Find any file anywhere on the file system based on your specified criteria with find.

```
# find / -name cat
```

- Takes a long time to search through entire FS.

```
# find / -name passwd
# find . -iname .ssh -type d
# find / -uid 0 -gid 0
# find / -amin -30 -user www-data 2>/dev/null
# find / -type f -name *.conf
# find / -samefile /my/reference/file
# find / -regex ".*\(id_dsa\|id_ecdsa\|id_rsa\)" -exec stat {}
\; -exec cat {} \;
```

MANIPULATING

Manipulating objectives:

Learn how manipulate files	
Creating	touch, mkdir, ln
Copying	cp, ln
Moving	mv
Deleting	rm, rmdir
Viewing	file, cat, more, less
Searching	grep basics

Manipulating: Creating: touch &mkdir

- Create a new file or change the access or modify timestamps with touch.

```
# touch this
```

- Create a new folder with mkdir. Certain arguments allow permissions and parents folders to be created on the fly.

```
# mkdir ~/newfolder
# mkdir -m 640 -p /tmp/new/folder
```

Manipulating: Creating

- Create a new file or change the access or modify timestamps with touch.

```
# touch this
```

- Use a long listing [ls-l]to view the modify time.

```
# touch -a /usr/local/newtool
# touch -m /var/log/modify
# touch -t 201008091222.23 /var/log/messages
# touch -r /var/log/wtmp /var/run/utmp
# touch --date="2004-02-27 14:19:13.489392193
+0530" /tmp/foo
# touch -h
```

Manipulating: Copy: Link

Create a new file or make a "copy" by linking the file with ln. This will make a shortcut or new file reference.

```
# ln /var/log/secure ~/securer
# ln -s /etc/init.d/sshd /etc/rc.d/rc5.d/sshd
```

Manipulating: Copy

- Create a copy of a file or folder with cp.

```
# cp from to
```

There are some arguments and shortcuts, so research [cp--help] or [man cp].

Manipulating: Move

- Move a file or folder with mv.

```
# mv from to
```

- There are some arguments and shortcuts, so research [mv--help] or [man mv].

```
# mv -fu ~/dir /tmp
# mv /var/log/somelog{2,1}.log
```

- Use mv to rename files.

Manipulating: Remove

- Remove a file or folder with rm.

```
# rm file
```

There is NO undo!

There are some arguments and shortcuts, so research [rm--help] or [man rm].

```
# rm -f this
# rm -rf ~/junk/files/
```

some distros will have [alias rm=rm-i] which will ask you if you are sure you want to delete.

Manipulating: View: file

- Determine the type of file an object is.

```
# file this
```

-
- There are no file extensions per se, so a "magic" test is performed and depending on the file type, specific data is retrieved.

```
# file /tmp
/tmp: sticky directory
# file /dev/dvd
dev/dvd: symbolic link to `sr0'
```

Manipulating: View: cat

- Display the contents of file(s) with **cat.**

```
# cat this
```

  - Dumps the contents to STDOUT.
  - Can reconstitute chopped up files.
  - Can add line numbers.

```
# cat file > /dev/tcp/::1/80
# cat thing1 thing2 > things
# cat -nA /var/log/messages
# cat /bin/dir
```

Manipulating: View: more & less

- View text one page at a time with more.

```
# more oldtools.txt
```

- More& less are paging tools, which go full-screen.
- Be careful when you use these tools.
- They use arrow-keys, page-up/down, space-bar to navigate.
- Search with [/ word], jump to next match with n or previous with N.
- Quit with q.

Manipulating: View: more & less

- View files one page at a time with less.

```
# less newtools.txt
```

- Less is a better version of more.
- Depending on the age of the OS or version of less, new features will be available.
- It can read ISO images, pdf, MS word & more.
- Less can follow files like tail

```
# less +F /var/log/messages
```

Manipulating: Search

- Search within a file(s) recursively with grep.

```
# grep word file.txt
```

- Grep and other core utilities differ between versions like GNU and others. Know which OS you are on and which version of grep is there.

```
# grep --version
```

- Case insensitive search must be specified.

```
# grep -i words file.txt
```

Linux file system: Important