

Lab 03 - Filesystem Hierarchy & User Management

1. Name the different directories under the `/` directory

```
[root@UWS /]# ls -lah | grep '^d'
```

drwxrwxrwx	22	root	root	501	Sep 28 18:11	.
drwxrwxrwx	22	root	root	501	Sep 28 18:11	..
drwxr-xr-x	2	root	root	2.3K	Nov 29 2020	bin
drwxrwxr-x	3	root	root	82	Dec 10 2020	boot
drwxr-xr-x	4	root	root	2.4K	Sep 28 18:11	dev
drwxr-xr-x	17	root	root	1.3K	Dec 10 2020	etc
drwxr-xr-x	3	root	root	61	Sep 28 18:11	home
drwxr-xr-x	6	root	root	1.6K	Nov 29 2020	lib
drwxr-xr-x	2	root	root	57	Nov 29 2020	libexec
drwxr-xr-x	4	root	root	82	Sep 9 2020	media
drwxr-xr-x	2	root	root	37	Sep 9 2020	mnt
drwxr-xr-x	2	root	root	37	Sep 9 2020	opt
dr-xr-xr-x	60	root	root	0	Sep 28 18:11	proc
drwx-----	3	root	root	85	Sep 9 2020	root
drwxr-xr-x	5	root	root	380	Sep 28 18:11	run
drwxr-xr-x	2	root	root	2.6K	Dec 10 2020	sbin
dr-xr-xr-x	12	root	root	0	Sep 28 18:11	sys
drwxrwxrwt	5	root	root	160	Sep 28 18:11	tmp
drwxr-xr-x	10	root	root	253	Dec 10 2020	usr
drwxrwxr-x	3	root	root	289	Dec 10 2020	uwslabs
drwxr-xr-x	5	root	root	221	Nov 29 2020	var

Dir. Name	Contents
/bin	Contains critical programmes required by the system in order to function
/boot	Location for the bootable kernel and bootloader configuration
/dev	Access points for devices present in the system
/etc	Configuration files
/home	With the exception of root, the location of each user's home directory
/lib	Shared libraries for applications
/media	A default location for mounting devices
/mnt	An additional mount point for devices
/opt	A location where applications can be installed
/proc	Information about resources available to the system
/root	The root user's home directory
/sbin	Applications generally only available to the root user, daemon processes
/sys	Contains a <code>sysfs</code> filesystem, information about system hardware
/tmp	A location for temporary files generated, used, by applications
/usr	Contains executables, libraries, other system resources
/var	Contains files which are subject to change often, system logs, spools, etc.

2. (After attempting to change to a parent directory from `/`) Explain why there is no difference.

`/` has no parent folder, it is the top level of the file system.

3. Are there any hidden files in the root directory? If yes: What are their names?

Yes. `.fscmd`.

4. Explain the meaning of the `.` ('dot') and the `..` ('double dot') in the command lines `cd .` and `cd ..`

`..`

- `.` represents the current working directory
- `..` represents the parent directory
- `cd .` change directory to the present working directory
- `cd ..` change directory to the parent of the present working directory

5. Are there any subdirectories in `/bin`?

```
[root@UWS bin]# ls -ld */
ls: */: No such file or directory
```

If `.` and `..` are considered directories then there are two, otherwise no.

6. How many commands are in `/bin`? Write down and explain two commands that you already know.

```
[root@UWS bin]# ls -ALd * | wc -w
103
```

List all files in the present directory, except `.` and `..`, list the referenced file for any symbolic link, and list the directory's entries instead of its contents, omitting the total. Pipe the result through to `wc` and print the number of words to standard out.

7. Which of the four directories `/bin`, `/usr/bin`, `/usr/sbin`, `/opt` contains the most commands?

```
[root@UWS /]# find /bin/ -type f -executable | wc -l
97
[root@UWS /]# find /usr/bin/ -type f -executable | wc -l
254
[root@UWS /]# find /usr/sbin/ -type f -executable | wc -l
46
[root@UWS /]# find /opt -type f -executable | wc -l
0
```

8. Which of the four directories contains a large set of gnome-desktop related applications?

```
[student@UWS usr]$ sudo find / -name '*gnome*' 2> /dev/null
Password:
/usr/share/bash-completion/completions/gnome-mplayer
```

It appears that the only gnome-related *file* appears in `/usr/share/bash-completion/completions`, and there appear to be zero executable *applications* on the filesystem.

```
[student@UWS usr]$ find / -name -executable '*gnome*' 2> /dev/null
[student@UWS usr]$
```

9. Can you locate the `chroot` binary within `/usr/sbin`? Where does it point to?

The question seems to imply that `chroot` should symbolically linked to an alternate location, however this appears to not be the case. The `chroot` binary is located in `/`

```
[student@UWS usr]$ sudo find / -executable -name 'chroot' 2> /dev/null | xargs
file
/usr/sbin/chroot: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV),
dynamically linked, interpreter /lib/ld-
linux.so.2, for GNU/Linux 4.12.0, stripped
[student@UWS usr]$ sudo find / -executable -name 'chroot' 2> /dev/null | xargs
file
"/usr/sbin/chroot: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV),
dynamically linked, interpreter /lib/ld-
linux.so.2, for GNU/Linux 4.12.0, stripped"
```

10. What is the smallest size allocated to a directory file in `/usr`?

A directory called `i586-buildroot-linux-gnu`.

```
[student@UWS usr]$ ls -lahSr /usr/ | grep '^d'
drwxr-xr-x   3 root    root      57 Nov 11 2020 i586-buildroot-linux-gnu
drwxrwxr-x   6 root    root     123 Dec 10 2020 local
-- snip --
drwxr-xr-x  11 root    root     4.2K Dec 10 2020 lib
drwxr-xr-x   2 root    root     6.8K Dec 10 2020 bin
```

11. Why might `/usr/lib` be so large?

```
[student@UWS usr]$ du -sh /usr/lib
68.0M    /usr/lib
```

Given that it contains shared libraries to be used by (potentially) all applications, it stands to reason that a significant amount of common utility will be located in `/usr/lib`, hence the overly large file size relative to other directories.

By comparison, on my own computer the same folder comes in at `6.0GB`.

12. Which is the biggest standard directory in our system?

```
[student@UWS /]$ sudo du -hc -d 1 2> /dev/null | sort -gr | grep M
136.0M    total
136.0M    .
111.4M    ./usr
5.8M      ./lib
5.0M      ./boot
4.6M      ./var
3.3M      ./sbin
2.5M      ./uwslabs
2.4M      ./bin
```

`/usr` appears to be the largest standard directory.

13. What is the total size of our current system?

The total size of the current system is 136 Megabytes.

14. Try the command `cat /etc/passwd > /dev/stdout`. (The `cat` command displays (concatenates) the contents of a file to an output device such as the screen...) Explain why you see the contents of the file displayed

```
[student@UWS /]$ cat /etc/passwd > /dev/stdout
root:x:0:0:root:/root:/bin/sh
daemon:x:1:1:daemon:/usr/sbin:/bin/false
-- snip --
nobody:x:65534:65534:nobody:/home:/bin/false
student:x:1000:1000:Linux User,,,:/home/student:/bin/bash
```

The content of the file is redirected from standard output, the default behaviour for `cat` and redirected to `stdout`, which then outputs an input stream to standard output. See below for a further example.

```
[student@UWS /]$ echo "hello world" > /dev/stdout
hello world
```

15. If `stderr` is the standard channel for displaying error messages. Where is `stderr` directed to?

Since the terminal functions in terms of text streams, and `stderr` outputs to standard output, `stderr` directs output to standard output as default behaviour.

16. What is the meaning of the `-l` qualifier in the `grep` command?

`grep -l` will only output matching filenames.

17. Which files reference the IP-address?

Assuming the solution uses the loopback address:

```
[student@UWS ~]$ sudo find /etc -type f -exec grep -l '127.0.0.1' {} \;  
/etc/security/access.conf  
/etc/hosts
```

Assuming otherwise:

```
[student@UWS ~]$ netstat -ie  
Kernel Interface table  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 10.5.226.169 netmask 255.255.0.0 broadcast 10.5.255.255  
    ether 02:8e:83:9a:5a:ff txqueuelen 1000 (Ethernet)  
    RX packets 3091 bytes 222020 (216.8 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 12 bytes 1535 (1.4 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 2 bytes 140 (140.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 2 bytes 140 (140.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
[student@UWS ~]$ sudo find /etc -type f -exec grep -l '10.5.226.169' {} \; #  
zero results  
[student@UWS ~]$
```

18. How would you make absolutely sure, you are examining the whole filesystem for e.g. the occurrence of a pattern like `127.0.0.1`?

I'd alter the command to search from `/` rather than `/etc`:

```
[student@UWS ~]$ sudo find / -mindepth 0 > find_from_root  
[student@UWS ~]$ sudo find /etc -mindepth 0 > find_from_etc  
[student@UWS ~]$ wc -l find_from_root  
20935 find_from_root  
[student@UWS ~]$ wc -l find_from_etc  
108 find_from_etc
```

19. Locate the Linux commands `fscck` and `find` in the filesystem

```
[student@UWS ~]$ find / -name 'find' -or -name 'fsck' 2> /dev/null
/sbin/fsck
/usr/bin/find
/usr/share/bash-completion/completions/find
/usr/share/bash-completion/completions/fsck
[student@UWS ~]$ whereis find
find: /usr/bin/find /usr/share/man/man1/find.1.gz
[student@UWS ~]$ whereis fsck
fsck: /sbin/fsck.ext4 /sbin/fsck /sbin/fsck.ext2 /sbin/fsck.ext3
/usr/share/man/man8/fsck.8.gz
```

- /sbin/fsck
- /usr/bin/find

20. Locate the standar C library represented by the file `libc.so.6`

```
[student@UWS ~]$ find / -name 'libc.so.6' 2> /dev/null
/lib/libc.so.6
```

21. Locate and run the application `dmesg` using only the `find` command. Write down the full (find) command line that you used in your log-book.

```
[student@UWS ~]$ find / -name 'dmesg' -executable 2> /dev/null -exec {} \;
[ 0.000000] Linux version 4.12.0 (hecmargi@maximo) (gcc version 9.3.0 (Ubuntu
9.3.0-17ubuntu1~20.04) ) #2 Mon Nov 30 12:20:22 CET 2020
[ 0.000000] CPU: vendor_id 'AuthenticX86' unknown, using generic init.
        CPU: Your system may be unstable.
[ 0.000000] x86/fpu: x87 FPU will use FXSAVE
[ 0.000000] e820: BIOS-provided physical RAM map:
-- snip --
[ 5.947823] clocksource: Switched to clocksource pit
[ 8.657060] random: crng init done
```

22. What is the full size of the `/home` directory?

```
[student@UWS ~]$ du -hc /home
4.0K   /home/student/Desktop
680.0K /home/student
684.0K /home
684.0K total
```

684K

23. Determine the number of shared libraries in `/lib`. Shared libraries end in `.so.*` (Where `*` represents a wildcard).

```
[student@UWS ~]$ find /lib -name "*.so*" | wc -l
88
```

24. Find out whether `libreoffice` has an entry in `/var`.

It does not.

```
[student@UWS ~]$ sudo find /var -name 'libreoffice'
[student@UWS ~]$
```

25. Deduce the role of `/var/log/messages` by reviewing its contents.

```
[student@UWS ~]$ less /var/log/messages
-- snip --
Oct  2 11:58:09 UWS user.info kernel: console [hvc0] enabled
Oct  2 11:58:09 UWS user.info kernel: loop: module loaded
Oct  2 11:58:09 UWS user.info kernel: i8042: No controller found
Oct  2 11:58:09 UWS user.info kernel: NET: Registered protocol family 17
Oct  2 11:58:09 UWS user.info kernel: 9pnet: Installing 9P2000 support
Oct  2 11:58:09 UWS user.info kernel: registered taskstats version 1
Oct  2 11:58:09 UWS user.info kernel: VFS: Mounted root (9p filesystem) readonly
on device 0:14.
Oct  2 11:58:09 UWS user.info kernel: devtmpfs: mounted
Oct  2 11:58:09 UWS user.info kernel: Freeing unused kernel memory: 320K
Oct  2 11:58:09 UWS user.info kernel: Write protecting the kernel text: 3780k
Oct  2 11:58:09 UWS user.info kernel: Write protecting the kernel read-only data:
900k
-- snip --
```

It appears that the file is a place for kernel messages, information about the system, information about boot processes, critical information, etc.

26. What information is given about the unsuccessful login attempt? Could you identify the hacking culprit at once?

From `man ps`:

ruser ... real user ID. This will be the textual user ID, if it can be obtained and the field width permits, or a decimal representation otherwise.

```
[student@UWS ~]$ tail /var/log/messages
-- snip --
Oct  2 13:35:36 UWS authpriv.notice su: pam_unix(su-l:auth): authentication
failure; logname=student uid=1000 euid=0 tty=console ruser=student rhost=
user=root
Oct  2 13:35:38 UWS auth.notice su: FAILED SU (to root) student on console
[student@UWS ~]$ cat /etc/passwd | grep 1000
student:x:1000:1000:Linux User,,,:/home/student:/bin/bash
```

User `student` failed to switch user to root at 12:35:38 on the 2nd October.

27. How does the `tail` command compare to the `more` command?

`more` reads the entire input file before paging through its content. `less` performs much the same functionality without the need to read the entire file. It also allows for a command mode which gives users opportunities to interact with the content, i.e. sending a file to `less` and searching for a specific pattern.

28. What is the name and the size of the biggest filesystem entry of `/proc`? Do you have any idea what it may represent?

```
[student@UWS proc]$ ls -lahS
total 4
-r----- 1 root root 1023.9M Oct 2 14:08 kcore
drwxrwxrwx 22 root root 501 Oct 2 10:57 ..
lrwxrwxrwx 1 root root 11 Oct 2 14:08 mounts -> self/mounts
lrwxrwxrwx 1 root root 8 Oct 2 14:08 net -> self/net
dr-xr-xr-x 60 root root 0 Oct 2 10:58 .
dr-xr-xr-x 8 root root 0 Oct 2 11:16 1
dr-xr-xr-x 8 root root 0 Oct 2 11:16 10
-- snip --
```

`kcore` appears to be the largest file. We can find more information about it by using `file`:

```
[student@UWS proc]$ sudo file kcore
kcore: ELF 32-bit LSB core file Intel 80386, version 1 (SYSV), SVR4-style, from
loglevel=3 console=hvc0 root=root rootfstype=9p rootflags=trans=virtio ro TZ=UT
```

It's the kernel core!

29. What is the link between the `PID` of the running processes and the directory names within `/proc`?

There is a direct relationship in that there is a directory in `/proc` for each process ID running at the time. Consider the following example:

```
[student@UWS ~]$ ps -ef | grep sleep
student 1398 1117 0 14:25 hvc0 00:00:00 grep sleep
[student@UWS ~]$ sleep 600 &
[2] 1399
[student@UWS ~]$ ps -ef | grep sleep
student 1399 1117 1 14:26 hvc0 00:00:00 sleep 600
student 1401 1117 0 14:26 hvc0 00:00:00 grep sleep
```

- `sleep` is not currently running
- `sleep` is executed and sent to the background using `&` for 600 seconds
- querying `ps -ef` again shows `sleep` with a process ID of `1399`

```
[student@UWS ~]$ ls -lah /proc | grep 1399
dr-xr-xr-x 8 student student 0 Oct 2 14:26 1399
[student@UWS ~]$ ls /proc/1399
auxv          cpuset        gid_map       mounts        oom_score_adj
-- snip --
[student@UWS ~]$ kill 1399
[2]- Terminated sleep 600
[student@UWS ~]$ ls /proc/1399
ls: /proc/1399: No such file or directory
```

- `ls` is run on `/proc` and fed to `grep` with a pattern matching the process ID
- a directory is found containing a number of files, directories, links, etc.
- the process is terminated using `kill` against the corresponding process ID
- running `ls` a second time on the directory reveals that the directory no longer exists. A direct relationship exists.

30. What do you think will happen to each directory in `/proc` after the associated process has been killed?

When a process is killed its corresponding directory entry in `/proc` will no longer exist.

31. What is written in the `cmdline` file? Does this agree with the information as given by the command, `ps -ef`?

each instance of `cmdline` contains the content of the `CMD` column for each entry returned by `ps -ef`.

32. What is the status (check the contents of `/proc/1/status`) and the memory size (`VmSize`) that is used by the `init` process?

```
[student@UWS ~]$ cat /proc/1/status | grep VmSize
VmSize:      2108 kB
```

33. Name the different directories that are present in the `/media` directory.

- `disk1`

34. Now set in `/media/floppy` (having a formatted floppy with some data on it inserted).

```
root@UWS ~)# cd /media/floppy/
[root@UWS floppy]# echo 'a dummy file' > testfile.dat
[root@UWS floppy]# ls -l
total 14
-rw-r--r--    1 root    root          61 Sep 13  2020 file.txt
drwx-----    2 root    root     12288 Sep 13  2020 lost+found
-rw-r--r--    1 root    root         13 Oct  2 20:58 testfile.dat
```

35. Why is the actual password depicted as `x`, although the password is not `x`?

This indicates that the password is encrypted as part of `/etc/shadow`.

36. What is the user identification (UID), home directory and login shell of the `root`-user?

```
[student@UWS /]$ more /etc/passwd
root:x:0:0:root:/root:/bin/sh
```

- `0`
- `/root`
- `/bin/sh`

37. What is the `UID` of the daemon called `uucp`?

```
[student@UWS ~]$ grep uucp /etc/passwd
[student@UWS ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/sh
daemon:x:1:1:daemon:/usr/sbin:/bin/false
bin:x:2:2:bin:/bin:/bin/false
sys:x:3:3:sys:/dev:/bin/false
sync:x:4:100:sync:/bin:/bin/sync
mail:x:8:8:mail:/var/spool/mail:/bin/false
www-data:x:33:33:www-data:/var/www:/bin/false
operator:x:37:37:operator:/var:/bin/false
nobody:x:65534:65534:nobody:/home:/bin/false
student:x:1000:1000:Linux User,,,:/home/student:/bin/bash
```

There appears to be no entry for `uucp` as part of the list of users.

38. Determine the `UID` of student in the Debian system.

1000

39. How many users use the `/bin/sh` shell as their login shell?

```
[student@UWS ~]$ more /etc/passwd | grep '/bin/sh' | wc -l
1
```

- 1
- `wc -l` receives standard input and returns the number of lines present in the input.