Sean Poston

4/10/2020

CS351 Assignment 6

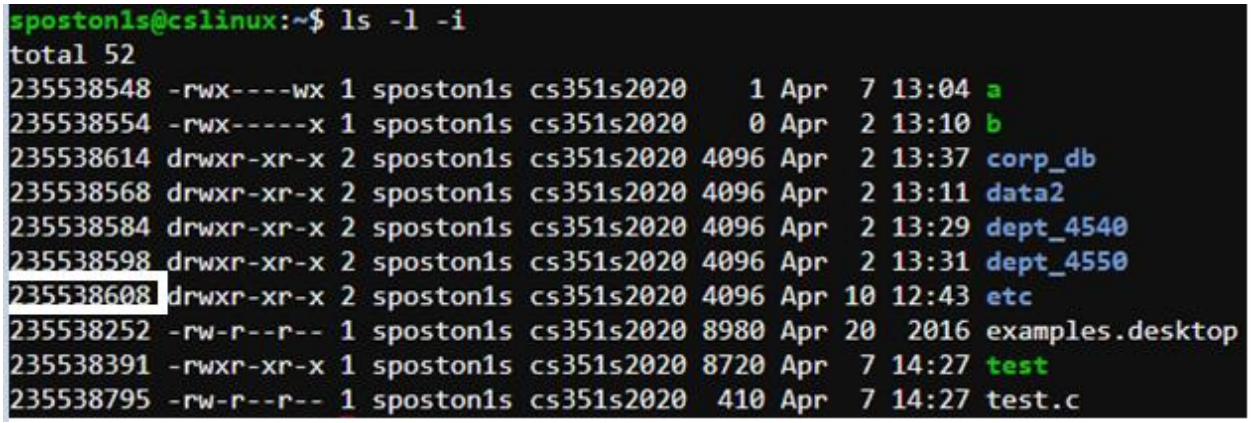
109, 2:

```
sposton1s@cslinux:~$ ls -l
total 48
-rwx----wx 1 sposton1s cs351s2020    1 Apr  7 13:04 a
-rwx-----x 1 sposton1s cs351s2020    0 Apr  2 13:10 b
drwxr-xr-x 2 sposton1s cs351s2020 4096 Apr  2 13:37 corp_db
drwxr-xr-x 2 sposton1s cs351s2020 4096 Apr  2 13:11 data2
drwxr-xr-x 2 sposton1s cs351s2020 4096 Apr  2 13:29 dept_4540
drwxr-xr-x 2 sposton1s cs351s2020 4096 Apr  2 13:31 dept_4550
-rw-r--r-- 1 sposton1s cs351s2020 8980 Apr 20  2016 examples.desktop
-rwxr-xr-x 1 sposton1s cs351s2020 8720 Apr  7 14:27 test
-rw-r--r-- 1 sposton1s cs351s2020  410 Apr  7 14:27 test.c
```

The *ls* command is used to list the files and directories in the current directory. As exemplified above, the *ls* command is used with the *-l* option. This puts the list in a “long listing” format, which displays it in a table format. If the *ls* command is used without any options, it will be a list with a few spaces between each file, without any other information. This command with the *-l* option is one of the most useful commands because it allows you to see privileges, owner, group, size in Bytes, and the last date the file or directory was edited. This is extremely useful for being an admin on a server because it allows for a nicely formatted overview of all files created and their permissions.

The first column is permissions in RWX. RWX stands for read, write, execute. These are all locked place (except for the first character) and are in a state of on or off. The first three are permissions for the owner, the second set is the group, and the third set is “others.”

109, 3:



The “etc” directory has the inode of **235538608**. Adding the *-i* option to the *ls* command will print the inode information.

210, 3:

```
sposton1s@cslinux:~$ cp a a.bak
sposton1s@cslinux:~$ ls -l
total 56
-rwx----wx 1 sposton1s cs351s2020    1 Apr  7 13:04 a
-rwx-----x 1 sposton1s cs351s2020    1 Apr 10 12:50 a.bak
```

211, 10:

After creating a script file, let's say it's called test.c, you must compile it and then run it. To compile it using the GCC compiler, you have to use the command "gcc -o test test.c". This will compile the script file "test.c" into a new file "test" that the compiler is able to run. Then, once this is completed, you type "./test" to run the compiled file.