

1. Semester and Year : 2012, FALL
2. Course Number : CS-227
3. Course Title : Linux & System Software
4. Work Number : LA-02
5. Work Name : Command Line
6. Work Version : Version1
7. Long Date : Tuesday, 4, September, 2012
8. Author(s) Name(s) : Jake Waffle

Task 1.5: Edit Files

In this task, we are using Vi to edit text files within a terminal. Vi has three modes that it runs in. Each mode contains a different set of commands.

The command mode is one of them and our book says that the commands are usually in the form of single letters (page24.) When I first used Vi, I would try to add text while in the command mode, but Vi doesn't work that way. The command mode is more for navigating through the text and doing simple things like removing a character/line or adding a new line.

The ex mode is for manipulating the file itself instead of the text within the file. And it allows the user to exit out of the program (with or without saving) and open new text files. To enter a command in ex mode, the user needs to type a colon (which brings the user into ex mode) and then enter the command.

The insert mode is for inserting text into a text file. And it is entered from the command mode by typing either 'i' or 'a'. After entering the insert mode, navigation is no longer possible and the user can only insert text and exit insert mode (which can be done by hitting escape and brings the user to the command mode.)

Now I'm going to edit a text file with Vi. I'll use 'sudo vi textName' to enter vi with the appropriate text file. And upon entering a random text file, I navigated around with the arrow keys (not available in non-GUI terminal 'h', 'j', 'k' and 'l' are used instead.) I removed some characters with 'x', added a new line with 'o' and then proceeded to insert mode with 'i' after putting the cursor in the new line. Then I typed in some random string of characters and used 'esc' to exit insert mode. And finally I left the text file (without saving) by entering ':q!'. I could have entered ':wq' to save and quit, but my changes weren't coherent enough to even make me want to save it.

Task 1.6: Manage Accounts

For this task I will need to add an account, delete an already existing account and modify an account. And before I can do this, the book says that I need to understand how Linux stores its passwords.

Linux stores its information on accounts within `/etc/passwd` and `/etc/shadow`. `/etc/passwd` is “world-readable” (pg30) so that programs can access information on accounts. But the account’s password field will not contain the actual password in `/etc/passwd`, but it will contain a code that means it’s stored elsewhere, `/etc/shadow`. `/etc/shadow` is only accessible by root (to disallow common users from seeing the passwords) and it contains account names, encrypted passwords as well as other advanced information.

Adding an account is done with the ‘`useradd`’ command (unless the user prefers to do it manually through a text-editor) followed by the username of the account (pg 31.) Doing this will setup an account with default arguments, but one can add arguments to change the defaults if needed. Arguments need an option (i.e. `useradd username -d /homeDir`) placed before them, indicating what the argument is actually for.

Now I’m going to add a user to my linux distro. I used command “`useradd -d /home/Jake Jake`” to add an account with a specified home directory (because the task needs the home directory I think.) I checked to see if `/home/Jake` directory exists and it does. Then I went ahead and add a password to my new account with command ‘`passwd Jake`’.

Deleting a user’s account looks very simple from the way the book describes it. All that is needed is command ‘`userdel`’ followed by the username to delete an account (pg 32.) But this does not delete the home directory or mail spool according to the book. Adding the `-r` option to ‘`userdel`’ will delete the home directory and mail spool along with the account. I will not delete my newly added account, until after I’m done using it for this task though.

Modifying a user’s account is much like adding an account and is done with the command ‘`usermod`’ followed by options and the username of the account you’d like to modify. I tested the ‘`usermod`’ by changing the username of the account I previously made.