UNIX/IP Preparation Course

July 15th, 2010

Exercises: Using Commands

1. **Log in as the *sanog* user:**

   At the login log in using

   username: *sanog*
   password: <*given in class*>

2. **If you are running on Gnome/X, open a terminal window**

   Under Applications menu open the Accessories sub-menu and then choose Terminal. Once you have a terminal window open you can proceed with the exercises.

   Note: if you are on your laptop, this is not necessary, just use the text console on the PC

3. **Become the *root* user**

   At the command prompt type the following command:

   ```
   $ su -
   ```

   You will need to enter the password for the *root* user given in class.

   Now that you are *root* the command prompt will change. We indicate this using the "#" symbol.

4. **Add a default gateway**

   In order to complete your network configuration for the remaining exercises we need to add one additional line to your central configuration file, /etc/rc.conf. To do this type:

   ```
   # cp /etc/rc.conf /etc/rc.conf.orig
   # bash
   # echo 'defaultrouter="119.2.100.225"' >> /etc/rc.conf
   ```

   Note the use of single and double quotes (' and ") in the line above. We typed "bash" as the default shell ("sh") does not understand the format of the text in the echo command above.

   Note the use of redirection ">>" to append to an existing file.

5. **Install the sudo package**

   The sudo package allows you to run system commands as a regular user. To install sudo do the following:

   ```
   # pkg_add –r sudo-\*
   ```

   Now we must do a bit of configuration. Feel free to ask your instructor what this really does. Note the spaces in the following command

   [cp <space> /usr/local/etc/sudoers <space> /usr/local/etc/sudoers.orig]

```
# cp /usr/local/etc/sudoers /usr/local/etc/sudoers.orig
# echo "%wheel        ALL=(ALL) ALL"  >> /usr/local/etc/sudoers
```

Now go back to being a "regular" user (i.e. sanog):

```
# exit
```

6. **View files:**

   Use `ls` to list files:

   ```
   $ cd                                    [go to your home directory]
   $ ls
   ```

   Do you see anything? Try this instead:

   ```
   $ ls -lah
   ```

   What's inside one of these files? (If there is no .mailrc file us another "." file)

   ```
   $ cat .mailrc
   ```

   Try:

   ```
   $ less .mailrc
   ```

   Press "q" to get out of the `less` display.

   Now try:

   ```
   $ clear
   $ cat .mailrc
   ```

   If you don't understand what `cat, clear` or `less` do, then type:

   ```
   $ man cat
   $ man clear
   $ man less
   ```

   And, now try:

   ```
   $ more .mailrc
   ```

7. **Working with the command prompt:**

   You can recover previous commands by using the up-arrow and down-arrow keys. Give this a try now.

   Alternately, try typing this command:

   ```
   $ history
   ```

If you wish to execute one of the commands in the list you saw type:

```
$ !nn
```

Where "nn" is the number of the command in the history list. This is useful if you want to run a past command that was long and/or complicated.

Command completion:

With the bash shell you can auto-complete commands using the tab key. This means, if you type part of a command, once you have a unique string if you press the TAB key the command will complete. If you press the TAB key twice you'll see all your available options. Your instructor will demonstrate this, but give it a try by doing:

```
$ hist <TAB>
$ del <TAB><TAB>
$ rm <TAB><TAB>
```

[Include the space after the "rm"]

8. **Working with pipes:**

We saw an example of using pipes when we sorted the contents of our /sbin directory during the presentation. What if you wanted to have this information available in a file and sorted?

```
$ cd
$ ls /sbin | sort > sbin.txt
```

Now view the contents of what is in sbin.txt to verify that this worked. Do you remember how to do this? If not, go back to exercise 2.

9. **Finding text strings:**

Use the command `grep` to print lines matching a pattern in a data stream (such as a file). For example, view the entry for the *sanog* account in the system passwd file:

```
$ sudo grep sanog /etc/passwd
```

Note the use of the `sudo` command. The file `/etc/passwd` is restricted with r/o access for for the *root* user. To view the file you need to elevate your privilege level while executing the `grep` command. This is what `sudo` does. You will be prompted for a password – enter in the password for the *sanog* user.

You should see something like:

```
sanog:*:1001:0:sanog 16 User:/home/sanog:/usr/local/bin/bash
```

The previous items above are:

```
userid:passwd:uid:gid:Name:HomeDir:LoginShell
```

`grep` is often used with a pipe to reduce the number of results. For instance:

```
$ history | grep ls
```

Will display your previous use of the `ls` command from exercise 2.

10. **Editing the command line revisited:**

    It is particularly useful to realize that you can edit a command just as you would a line of text in a file. For instance, you can:

    - Use your back-arrow (left) and forward-arrow (right) keys to change text in a command.
    - Use the Home and End keys to go to the start and the end of a command.
    - Note: you *do not* need to go to the end of a command before pressing <ENTER> to execute the command.
    - You can use the `history` command with `grep` to find a previous command. You can copy and paste this command, then edit it to make adjustments. For long commands this can save considerable time.
    - Alternatively you can use the reverse-search feature of bash:
     1.) Press <CTRL>-R
     2.) type the term you are searching for.
     3.) Press <CTRL>-R again to cycle through all occurrences of the term in your history.
     4.) Press the right or left-arrow, HOME or END key to start editing the command.

    Let's give some of these editing rules a try:

    ```
    $ ls -lah /usr/local/lib | grep perl*
    ```

    Then, let's look for perl in /usr/local/lib/perl5. Try this:

    `<CTRL>-R,` type “perl”, then press left arrow. Edit the previous command (which you should now have) and add “/perl5” past “lib”. Use the left-arrow key to move. You should now have:

    ```
    $ ls -lah /usr/share/perl5 | grep perl*
    ```

    With your cursor just past the “5” in “perl5”. Press <ENTER> to execute the command. (by the way, you won't find anything.)

11. **Copy and pasting commands:**

    A nice feature in Unix is the built-in automatic copy buffer (think “automatic clipboard”). As soon as you highlight text it is available in your copy buffer. You ***do not*** need need to use the <CTRL>-C, <CTRL>-V keyboard combination to copy and paste text.

    After you have highlighted text, then you can place your cursor where you want to paste the highlighted text and press the *middle* mouse button to do the text paste.

    Give this a try:

    ```
    $ history | grep perl
    ```

    Locate the previous command from exercise 10 and highlight just that command. Highlight the command text (not the number next to it).

Now press the middle mouse key. The command should be pasted to your prompt.

Did this work? Now you could edit the command or just press enter to execute it.

This is very useful if you have two terminals open and want to use text from one terminal in a different terminal. Or, if you wish to copy resulting text from a command in one terminal in to a file that's open in another terminal – etc.

12. **Software Installation**

If you are logged in as the *root* user please do the following:

```
# exit
```

To become a normal user, like *sanog*. Your prompt should change to include a "$" sign.

```
$
```

Install some tools to help us find packages.

```
$ cd /usr/ports/ports-mgmt/portupgrade
$ make install
```

When you are prompted for optional settings (probably twice) use the tab key to select "OK" and then press ENTER to continue.

figure out what package installs the "*lynx"* text-based web browser

```
$ sudo ports_glob lynx
```

After some initial database creation of available ports you should see a response like this:

```
www/lynx
www/lynx-current
```

Let's install the lynx package. To do this do

```
$ sudo portinstall www/lynx
```

You will be asked some questions about what you would like to install with lynx. Leave the options as they are chosen and select "OK" using the tab key, then press ENTER.

You will see a considerable amount of installation information go across your screen. Once the lynx package is installed we can try using it.

```
$ lynx nsrc.org
```

```
Press "q" to exit lynx.
```

Q.) What do you think is happening?
A.) You are viewing the home page for the Network Startup Resource Center (NSRC) a text-based web browser called *lynx*.