ET2504 Computer Networking Software Tools

Introduction to lab environment

Objectives:
The purpose of this laboratory is to provide an introduction to GNU/Linux environment and to get in practice with the basic commands that we use in GNU/Linux system.

Before you start remember **'man'** command is our friend, which provides in depth information for the topic or command typed in. 'man' pages are very helpful while learning GNU/Linux.
usage: $man <command>
For example: $man man

Start the Terminal (Applications->Accessories->Terminal)
1. Issue the following basic commands and observe the output

$ whoami	It shows who logged on this system
$ hostname	It tells on which machine you are
$ pwd	shows the path of current working directory
$ ls	displays the list of files in the current working directory

2. Working with text editors

Text editors are essential items to handle many important tasks, for example to edit the configuration files, programming language source code and to create README files etc. GNU/Linux supports many number of text editors like Kate, Geany, emacs, vi, pico , Gedit, Kwrite etc. some of them are pre installed, for example “vi”.

Exercise: Use pico editor and create a text file with name 'README.txt' in your 'home' directory, enter some text and exit by saving. Now display the content of that file on screen.

3. Working with files and directories

These are the basic commands that are useful when working with files and directories.

$ mkdir <dir_name>	creates a directory with specified 'dir_name'
$ cd <dir_name>	switches to 'dir_name' directory
$ cd ..	moves one directory up
$ cd ../ ../	moves two directories up (and so on)
$ cd	brings you to the highest level of your home directory
$ rmdir <dir_name>	removes entire directory
$ rm <file_name>	removes file name
$ rm -r < dir_name>	removes directory including it's contents
$ mv <name1> <name2>	renames the directories or files
$ mv <name> <path>	moves files/directories to the specified path
$ cp <name> <path>	copies file/directory as specified in path

Exercise: Use emacs editor (if not installed in your Linux box, install it) and create four text files with the following names: File1.txt, File11.txt, File12.txt, and a folder File12. Now add some text in each file and verify the directory contents and location with 'ls' and the 'pwd' commands.

- Create two directories with names 'backup1' and 'backup2' and 'copy' all the files with '.txt' extension
- to 'backup1' directory. Now 'move' the files with the '.txt' extension to the 'backup2' directory. List the contents of appropriate directories.
- Now position yourself in the 'backup2' directory. Do make changes in 'File11.txt' and copy this file to 'backup1' directory. Now delete all the files with '.txt' extension after copying them to '/home' directory.

Explore with the following commands while you are doing this exercise

'find' command used to find location of the filename
$ find / -name <name of the file>

'locate' command also do the same but more quickly
$locate <name of the file>

where as 'grep' is used to search for the word in a file
grep <word to be searched> <name of the file>

4. File permissions

Permissions are settings which specifies what type of access does the user can have to the respective file or folder. i.e they indicate what kind of access does the user have in terms of Read/ Write/ Execute .

More information file permissions can be found on
http://www.comptechdoc.org/os/linux/usersguide/linux_ugfilesp.html

Exercise:
Now position your self in backup1 'directory', first look at read/ write access of all the files in the respective directory. Now restrict the rights to read 'File1.txt' except you.
– Give all permissions to all users(read/write/execute) to the file 'File11.txt' .
– Give read/write permissions to you ,but only read permission to others to the file 'File12.txt'.

5. Backing up your files
GNU/Linux supports two key features to back up your files one that keeps all your important data together and the other will compress data that you can store in any external drive.

Go to 'backup1' directory, give the following commands which will move all files together.
$ tar -cvf folder_backup.tar *
(it will copy all the files and store them folder_backup.rar directory) 'tar' just arrange all files together in one file.

All we have to do is to compress the file size to manage easily. To compress the we have following command:
$gzip <tar file >
(the result would be like folder_backup.rar.gz)

Exercise: unzip the “folder_backup.rar.gz” folder, and store all the files with folder name unzip_backup

6. Package management
Advanced package management APT is a powerful command-line tool used to perform such functions as installation of new software packages, upgrade of existing software packages.

$sudo apt-get install <package name>
$sudo apt-get remove <package name>

Exercise: install emacs22-el package and remove it

7. Working with ssh
ssh is a program for logging into a remote machine and for executing commands on a remote machine. It is intended provide secure encrypted communications between two untrusted hosts over an insecure network.

Exercise: log on to the computer 'ssh.student.bth.se' via ssh with following command
$ ssh <username>@ssh.student.bth.se
here 'username' is your student login name (acronym)

Now create a directory with name 'Student' in your home directory and also create a text file 'ssh.txt' with in 'Student' directory.

8. Secure transfer files between two computers
scp (secure file copy) copies files between hosts on a network. It uses 'ssh' for data transfer, and uses the same authentication and provides the same security as 'ssh'.
Exercise:
– Copy the 'backup1' directory from your local system to the Student directory of remote system.
– Copy the 'backup1' directory from your local system to the Student directory of remote system.
– Now copy 'ssh.txt' from the Student directory of your remote system to 'backup2' of your local system.

$ scp -r file1 rako07@ssh.student.bth.se:/home/saxon/students/20071/rako07/
$ sudo scp -r rako07@ssh.student.bth.se:/home/saxon/students/20071/rako07/file2 .

9. Shutting down/ reboot the Linux system
Issue the command “ $shutdown -h now ” to shutdown or “$ shutdown -r now” to reboot .

Exercise: Now, instruct the GNU/Linux system to shut it down at a specified time. For help $man shutdown .