Linux System Administration

Linux File Editor

- A text editor is a program which enables you to create and manipulate data (text) in a Linux file

- There are several standard text editors available on most Linux systems
  - **vi - Visual editor**
  - **ed - Standard line editor**
  - **ex - Extended line editor**
  - **emacs - A full screen editor**
  - **pico - Beginner's editor**
  - **vim - Advance version of vi**

- Our editor = vi (available in almost every Linux distribution)

Introduction to vi Editor

- vi supplies commands for:
  - Inserting and deleting text
  - Replacing text
  - Moving around the file
  - Finding and substituting strings
  - Cutting and pasting text

- Most common keys:
  - **`i – insert`**
  - **`Esc – Escape out of any mode`**
  - **`r – replace`**
  - **`d – delete`**
  - **`:q! – quit without saving`**
  - **`:wq! – quit and save`**

User Account Management

Commands

- useradd
- groupadd
- userdel
- groupdel
- usermod

Files

- /etc/passwd
- /etc/group
- /etc/shadow

Example:

```
useradd -g superheros -s /bin/bash -c "user description" -m -d
/home/spiderman spiderman
```

Switch Users and sudo Access

Commands

- su - username
- sudo command
- visudo

File

- /etc/sudoers

Monitor Users

- `who`
- `last`
- `w`
- `finger`
- `id`

System Utility Commands

- `date`
- `uptime`
- `hostname`
- `uname`
- `which`
- `cal`
- `bc`

Process / Services Commands

- systemctl or service
- ps
- top
- kill
- crontab
- at.

“ps” command

- **ps** command stands for process status and it displays all the currently running processes in the Linux system

Usage examples:

- **ps** = Shows the processes of the current shell

PID = the unique process ID
TTY = terminal type that the user logged-in to
TIME = amount of CPU in minutes and seconds that the process has been running
CMD = name of the command

- **ps -e** = Shows all running processes
- **ps aux** = Shows all running processes in BSD format
- **ps -ef** = Shows all running processes in full format listing *(Most commonly used)*
- **ps -u username** = Shows all processes by username.

“top” command

- top command is used to show the Linux processes and it provides a real-time view of the running system
- This command shows the summary information of the system and the list of processes or threads which are currently managed by the Linux Kernel
- When the top command is executed then it goes into interactive mode and you can exit out by hitting **`q`**

- **`Usage: top`**

PID: Shows task’s unique process id
USER: Username of owner of task
PR: The “PR” field shows the scheduling priority of the process from the perspective of the kernel
NI: Represents a Nice Value of task. A Negative nice value implies higher priority, and positive Nice value means lower priority.
VIRT: Total virtual memory used by the task
RES: Memory consumed by the process in RAM
SHR: Represents the amount of shared memory used by a task
S: This field shows the process state in the single-letter form
%CPU: Represents the CPU usage
%MEM: Shows the Memory usage of task
TIME+: CPU Time, the same as ‘TIME’, but reflecting more granularity through hundredths of a second.

“kill” command

- **kill** command is used to terminate processes manually
- It sends a signal which ultimately terminates or kills a particular process or group of processes

Usage:
kill [OPTION] [PID]
OPTION = Signal name or signal number/ID
PID = Process ID

kill -l = to get a list of all signal names or signal number

Most used signals are:
kill PID = Kill a process with default signal
kill -1 = Restart
kill -2 = Interrupt from the keyboard just like Ctrl C
kill -9 = Forcefully kill the process
kill -15 = Kill a process gracefully

System Monitoring

- `top`
- `df`
- `dmesg`
- `iostat 1`
- `netstat`
- `free`
- `cat /proc/cpuinfo`
- `cat /proc/meminfo`

Log Monitoring

Another and most important way of system administration is log monitor

Log Directory = **`/var/log`**

- **`boot`**
- **`chronyd = NTP`**
- **`cron`**
- **`maillog`**
- **`secure`**
- **`messages`**
- **`httpd`**

System Maintenance Commands

- `shutdown`
- `init 0-7`
- `reboot`
- `halt`

Changing System Hostname

- `hostnamectl – set-hostname newhostname`

- Version 7 = Edit /etc/hostname
- Version 6 = Edit /etc/sysconfig/network

Finding System Information

- `cat /etc/redhat-release`
- `uname -a`
- `dmidecode`

Terminal Control Keys

Several key combinations on your keyboard usually have a special effect on the terminal.

These "control" (CTRL) keys are accomplished by holding the CTRL key while typing the second key.
For example, CTRL-c means to hold the CTRL key while you type the letter "c".

The most common control keys are listed below:

- **`CTRL-u - erase everything you've typed on the command line`**
- **`CTRL-c - stop/kill a command`**
- **`CTRL-z - suspend a command`**
- **`CTRL-d - exit from an interactive program (signals end of data).`**

Terminal Commands

- **`clear`**

Clears your screen

- **`exit`**

Exit out of the shell, terminal or a user session

- **`script`**

The script command stores terminal activities in a log file that can be named by a user, when a name is not provided by a user, the default file name, typescript is used

Environment Variables

- To view all environment variables
  - **printevn OR env**
- To view ONE environment variable
  - **echo $SHELL**
- To set the environment variables
  - **export TEST=1**
  - **echo $TEST**
- To set environment variable permanently
  - **vi .bashrc**
  - **TEST='123'**
  - **export TEST**
- To set global environment variable permanently
  - **vi /etc/profile or /etc/bashrc**
  - **Test='123'**
  - **export TEST**