Session #3 Part 2

Users, Package Managers, & Intro to Processes

Summary

Users and Groups

Users

A user is anyone who has access to the system, it could be a user account for a real user (i.e a human) or a system user assosciated with a service or a program.

The root user is considered the admin of the system and has access to everything.

You can know any user's ID and the groups that they're in by using the command

```
id username
```

```
satharus@Argon:~$ id bin
uid=1(bin) gid=1(bin) groups=1(bin),3(sys),2(daemon)
```

A userID is a positive integer assigned to the user to identify it. Root will have the userID 0. After that will be system users that are assosciated with services or programs, numbered from 1 up to 999, real user accounts start from the UID 1000.

UID 0	-	Root user
UID 1 - 999	-	System and Program users
UID >= 1000	-	Real users

The users of the system are stored in the /etc/passwd file.

```
hassan@hassan-VirtualBox:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
_apt:x:105:65534::/nonexistent:/bin/false
messagebus:x:106:110::/var/run/dbus:/bin/false
uuidd:x:107:111::/run/uuidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:117::/nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/bin/false
colord:x:113:123:colord colour management daemon,,,:/var/lib/colord:/bin/false
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/:/bin/false
pulse:x:117:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:118:126:RealtimeKit,,,:/proc:/bin/false
saned:x:119:127::/var/lib/saned:/bin/false
usbmux:x:120:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
hassan:x:1000:1000:hassan,,,:/home/hassan:/bin/bash
lfs:x:1001:1001::/home/lfs:/bin/bash
```

Looking at the content of the file here, we can see that the root user is the first with UID 0.

We have system users ranging from 1 to 120.

Example: lightdm has a UID of 108, it is a system application. It is a display manager.

The final 2 lines have the 2 real users on the system.

Example: Hassan, UID 1000 (First real user created).

Adding and Deleting Users

To create a user, simply enter the command

```
sudo useradd -m username
```

Breaking it down:

`sudo`: Needed because you need administrator privilage to create a new a user.

`useradd`: The command used to add users.

`-m`: An option used to make a home directory for the new user by default.

If we create a user named temp and check the content of the etc/passwd file again:

```
temp:x:1002:1002::/home/temp:
hassan@hassan-VirtualBox:~$
```

To delete a user, simply enter the command

```
sudo userdel username
```

Setting passwords for users

To set or change a password for a user, we use the command

```
sudo passwd username
```

It'll then prompt you to enter the password.

Switching Users

To switch user, we use the command

`su username`

and you'll be prompted to enter the password

```
hassan@hassan-VirtualBox:~$ su temp
Password:
temp@hassan-VirtualBox:/home/hassan$
```

To return to the original user use the command

`exit`

```
temp@hassan-VirtualBox:/home/hassan$ exit
exit
hassan@hassan-VirtualBox:~$
```

Groups

Groups are basically a collection of users, it helps organize user access on the system

For example: If you're working for a company, you don't want the HR to edit the code and at the same time you don't want the developers to read the HR files. You'll put all the HR personnel in a group and give that group access to HR files and deny access to anyone who isn't in the HR group.

Adding and Deleting Groups

To create a new group, use the command

`sudo groupadd username` `sudo groupadd tempg`

To delete a group, use the command

`sudo groupdel groupname` `sudo groupdel tempg`

User Modification

Every user has 1 primary group and supplementary groups, to modify any user we use the command `usermod`

To change the primary group we'll add the option
`-g new_primary_group`

To change the supplementary groups we'll add the option
`-G new_supplementary_groups`

However this will overwrite the current supplementary groups a user has.
If we want to append the stated groups, we'll add the a option to append
`-aG new_supplementary_groups`

Example: `usermod -g prim_group -aG sup_groups user`

Package Managers

Packages and Repositories

A package in linux is considered to be a collection of files, it can be an application, a program or even documentation. Packages in Linux are stored in repositories where the package manager can easily find, download, and install them.

Repositories can be considered something like an app store, that has many packages on it, and you choose to install and upgrade packages from it.

Package Manager

The package manager is responsible for downloading, installing, searching, removing, and upgrading packages.

It consists of high and low level parts.

The **high level** package manager, called **“apt”** or **“apt-get”** in Debian-based distributions, is responsible for searching the repositories and finding the packages, it is also responsible for resolving **dependancies**.

A dependancy is a package required for another package to work.

For example: The program GIMP requires a toolkit called GTK+ to work, so the package manager automatically installs GTK+ when installing GIMP.

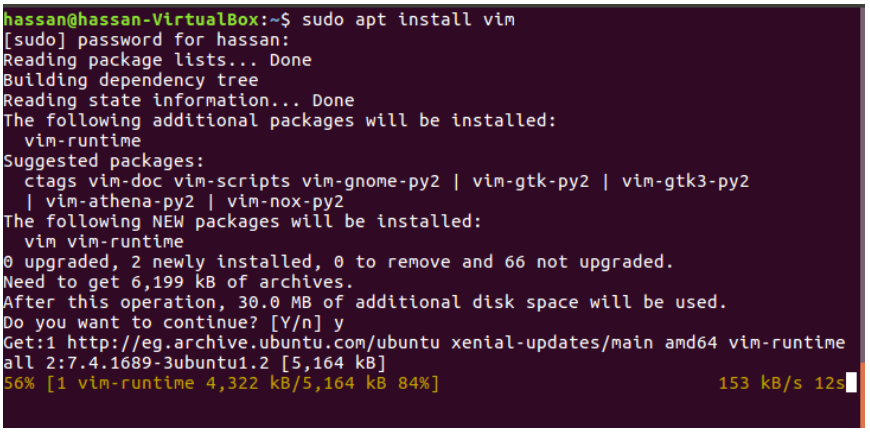
The **low level** manager, called **“dpkg”** in Debian-based distributions, is the one responsible for the actual **installation** and **compilation** of the packages.

Installing and Removing Packages

To install a package, we use the command

```
sudo apt install packages_names
```

In this example, vim-runtime is considered a dependancy, as vim needs it to work, the package manager notified us that it'll be installed alongside vim.



> Note: VIM (Vi IMproved) is a very powerful CLI Text editor for Unix-Like system.

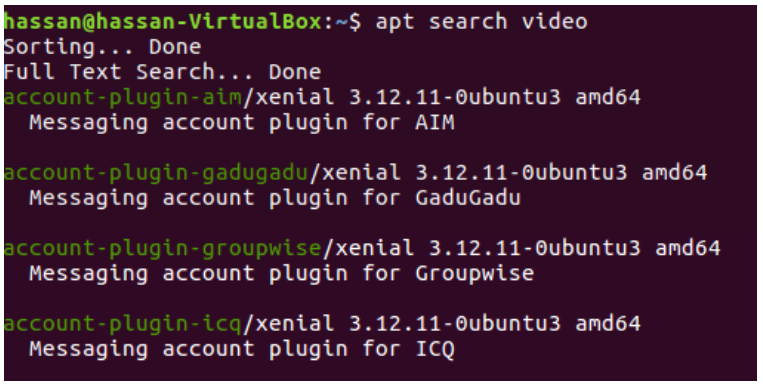
To remove a package, we use the command

```
sudo apt remove package_name
```

Searching for Packages

To search for packages, use the command

```
sudo apt search "keyword"
```



Updating and Upgrading

As mentioned before, packages are downloaded from repositories, which can be considered a storage for packages. However after a while the packages get updated and maybe new packages are added, the local repository data on your system may get outdated so you need to update the local data.

The command `apt update` will update the links inside the repository data file so that when you download or update something from the repository you'll get the lastest version.

As for the command `apt upgrade` it upgrades all the packages on your system to their latest versions available in the repositories.

Processes

What are Processes

A process is any program that is currently running on the system, you'll have foreground processes and background processes.

Background processses aren't seen by the user, this will include things such as update managers, network managers, etc ..

Foreground processes are programs that are currently being used by the user, such as Google Chromium, Firefox, GIMP, Codeblocks, etc..

PS and Top Command

The `ps` command is responsible for telling you all the processing currently running on the terminal

Note: PID is short for Process ID.

```
hassan@hassan-VirtualBox:~$ ps
  PID TTY          TIME CMD
 3284 pts/1    00:00:00 bash
 5117 pts/1    00:00:00 ps
```

Note: ps aux will show you ALL of the processes running of the system.

The `top` command will tell you all the current processes running in the system, and update them if any processes are killed or changed.

```
top - 07:39:12 up  1:35,  1 user,  load average: 0.64, 0.73, 0.63
Tasks: 177 total,   1 running, 176 sleeping,   0 stopped,   0 zombie
%Cpu(s): 32.6 us,  3.9 sy,  0.0 ni, 63.6 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  4044232 total,  1424680 free,  1246096 used,  1373456 buff/cache
KiB Swap:  2147324 total,  2147324 free,        0 used.  2447676 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 1427 hassan    20   0 1439844 218396  78528 S  53.2  5.4  20:16.72 compiz
 4261 hassan    20   0 1735056 249184 149376 S  13.6  6.2   2:23.90 Web Content
  863 root      20   0  538148 146796  46640 S   9.6  3.6   2:56.36 Xorg
 5173 hassan    20   0  427960  21872  18480 S   1.3  0.5   0:00.13 gnome-scre+
    1 root      20   0  185508   6064   3940 S   0.0  0.1   0:01.54 systemd
    2 root      20   0       0      0      0 S   0.0  0.0   0:00.01 kthreadd
    4 root       0 -20       0      0      0 S   0.0  0.0   0:00.00 kworker/0:+
    6 root      20   0       0      0      0 S   0.0  0.0   0:00.05 ksoftirqd/0
    7 root      20   0       0      0      0 S   0.0  0.0   0:00.60 rcu_sched
    8 root      20   0       0      0      0 S   0.0  0.0   0:00.00 rcu_bh
```

If you look at the figure above, you'll see a command with the PID 1 called systemd, this is the initialisation service responsible for the whole system after booting.

Signals

A signal is basically a command sent by the system to a process, the signal we'll discuss today is called **sigkill** which is responsible for **ending a process.**
It is similar to "End Task" in Windows.

To send a sigkill, we can use the command

```
kill PID
```

or we can use the command

```
killall process_name
```

Note: '#'means the start of a comment in the Linux shell.

```
satharus@Argon:~$ ps aux | grep -i -m 1 "firefox" #Finding the
PID of firefox
satharus 13393 61.1  4.5 2331712 373184 ?      Rl   11:48   0:0
4 /usr/lib/firefox/firefox
satharus@Argon:~$ kill 13393 #Killing Firefox by pid
satharus@Argon:~$ #Restarting firefox
satharus@Argon:~$ killall firefox #Killing Firefox by name
satharus@Argon:~$
```

Test yourself:

Let's have some fun, if you don't get something from the first time don't worry!

1. **Users and groups**
    a) Create 2 users named Ahmed and Ali.
    b) Create a group named moderators.
    c) Verify the groups that ahmed is in.
    d) Add ali to the moderators group, without changing his primary group.
    e) Verify the groups ali is in.
    f) Set a password for ali.
    g) Switch to ali.
    h) Exit back to your main user.
    i) Delete the user ali.
    j) Delete the group moderators.
    k) Delete the user ahmed.

2. **Package Managers**
    a) Update your system.
    b) Upgrade your system.
    c) Update and upgrade your system using one command.
    d) Install the package "vim".
    e) Remove the package "vim".

3. **Processes**
    a) Start firefox on your system.
    b) Verify that it is running, without using the GUI.
    c) Get the PID of firefox and write it down.
    d) Kill firefox using the process ID.

Solution:

1. Users and groups
   a) `sudo useradd ahmed`
   `sudo useradd ali`

   b) `sudo groupadd moderators`

   c) `id ahmed`

   d) `sudo usermod -aG moderators ali`

   e) `id ali`

   f) `sudo passwd ali`

   g) `su ali`

   h) `exit`

   i) `sudo userdel ali`

   j) `su groupdel moderators`

   k) `sudo userdel ahmed`

2. **Package Managers**
   a) `sudo apt update`
   b) `sudo apt upgrade`
   c) `sudo apt update && sudo apt upgrade`
   d) `sudo apt install vim`
   e) `sudo apt remove vim`

3. **Processes**
   a) `firefox` Or you can just open it from the GUI.
   b) `top` Or `ps aux`
   c) Look for the PID.
   d) `killall`