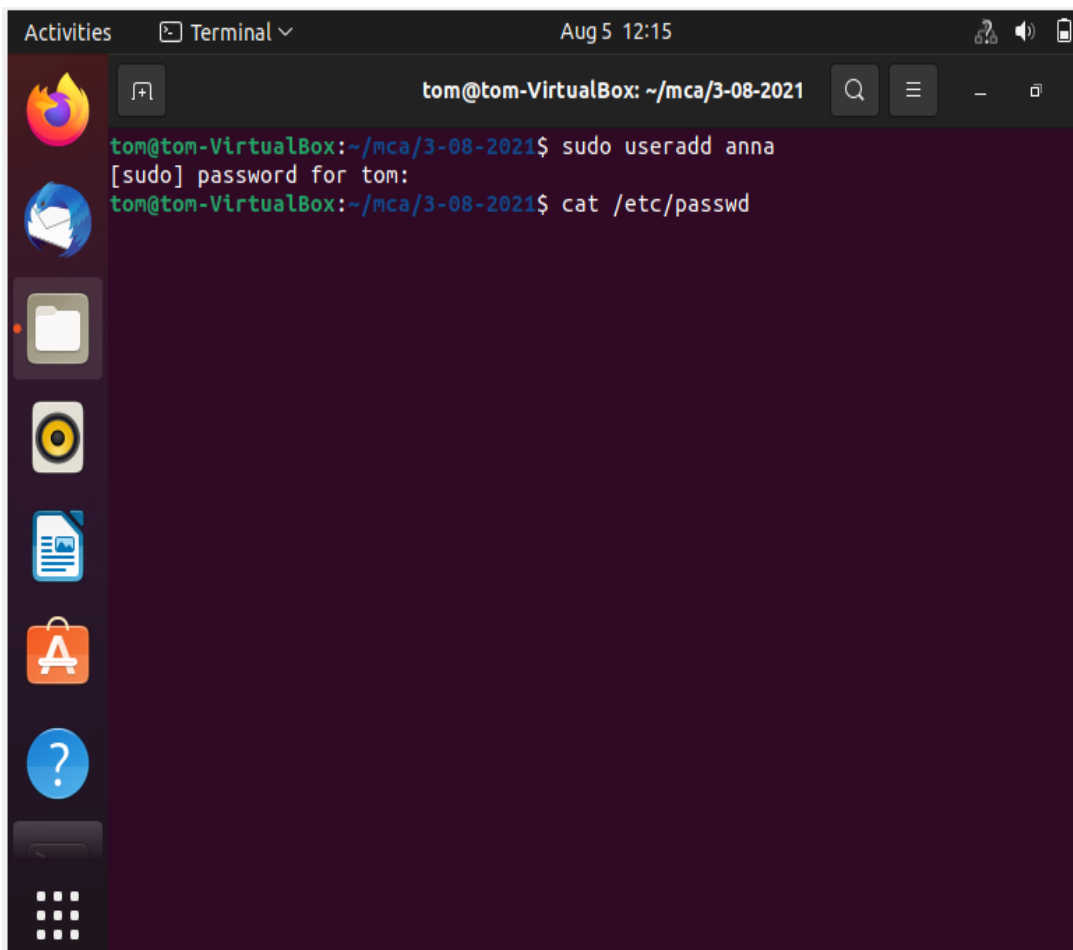


Assignment-3

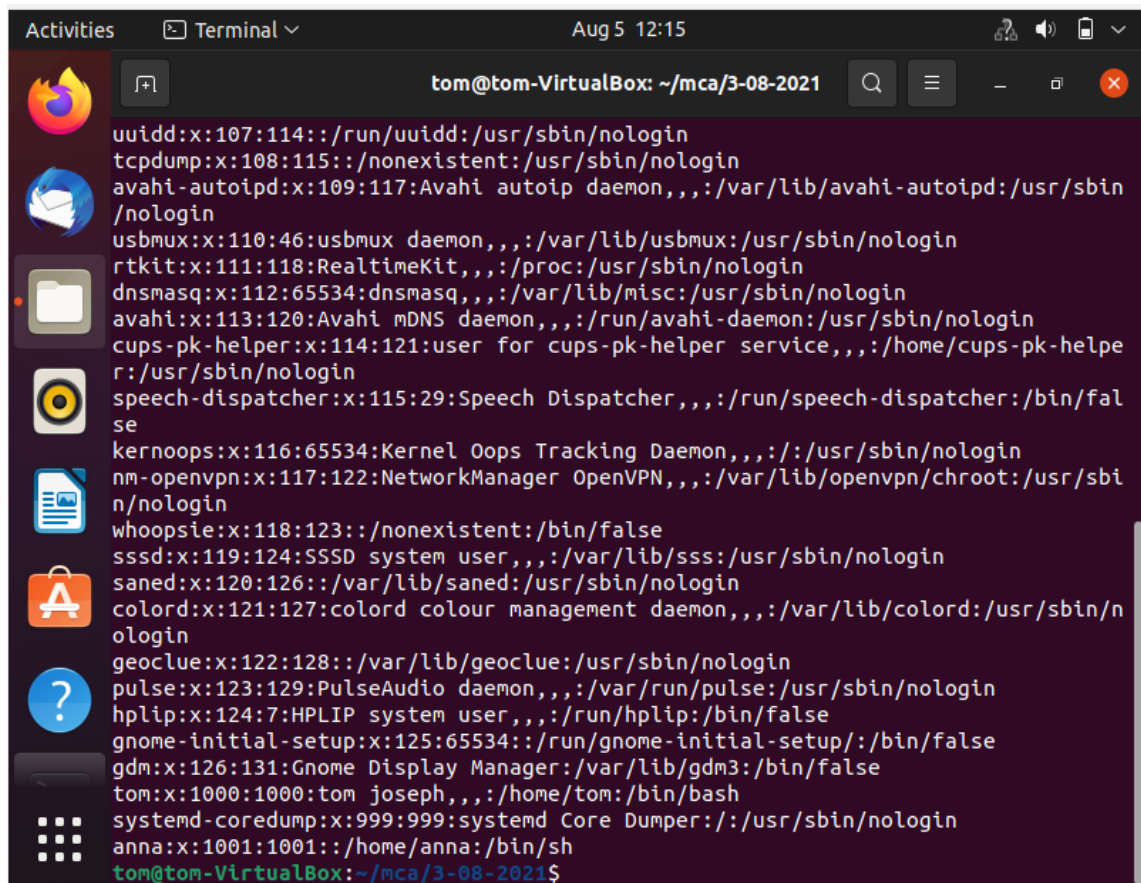
Commands

1. usermode

The usermod command or modify user is a command in Linux that is used to change the properties of a user in Linux through the command line. After creating a user we have to sometimes change their attributes like password or login directory etc.



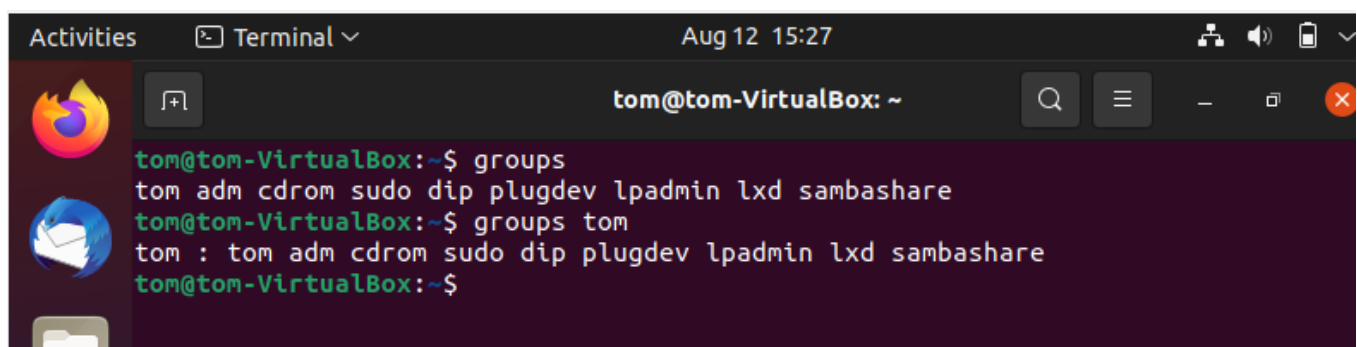
```
tom@tom-VirtualBox: ~/mca/3-08-2021
tom@tom-VirtualBox:~/mca/3-08-2021$ sudo useradd anna
[sudo] password for tom:
tom@tom-VirtualBox:~/mca/3-08-2021$ cat /etc/passwd
```

A terminal window titled 'tom@tom-VirtualBox: ~/mca/3-08-2021' showing the output of the 'cat /etc/passwd' command. The output lists system users and regular users with their IDs, names, and shell paths.

```
Activities Terminal Aug 5 12:15
tom@tom-VirtualBox: ~/mca/3-08-2021
uidd:x:107:114::/run/uidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:109:117:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
usbmux:x:110:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:111:118:RealtimeKit,,,:/proc:/usr/sbin/nologin
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
avahi:x:113:120:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
cups-pk-helper:x:114:121:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/nologin
speech-dispatcher:x:115:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin/false
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/usr/sbin/nologin
nm-openvpn:x:117:122:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
whoopsie:x:118:123::/nonexistent:/bin/false
sssd:x:119:124:SSSD system user,,,:/var/lib/sss:/usr/sbin/nologin
saned:x:120:126::/var/lib/saned:/usr/sbin/nologin
colord:x:121:127:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:122:128::/var/lib/geoclue:/usr/sbin/nologin
pulse:x:123:129:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
hplip:x:124:7:HPLIP system user,,,:/run/hplip:/bin/false
gnome-initial-setup:x:125:65534::/run/gnome-initial-setup:/bin/false
gdm:x:126:131:Gnome Display Manager:/var/lib/gdm3:/bin/false
tom:x:1000:1000:tom joseph,,,:/home/tom:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
anna:x:1001:1001::/home/anna:/bin/sh
tom@tom-VirtualBox:~/mca/3-08-2021$
```

2. group

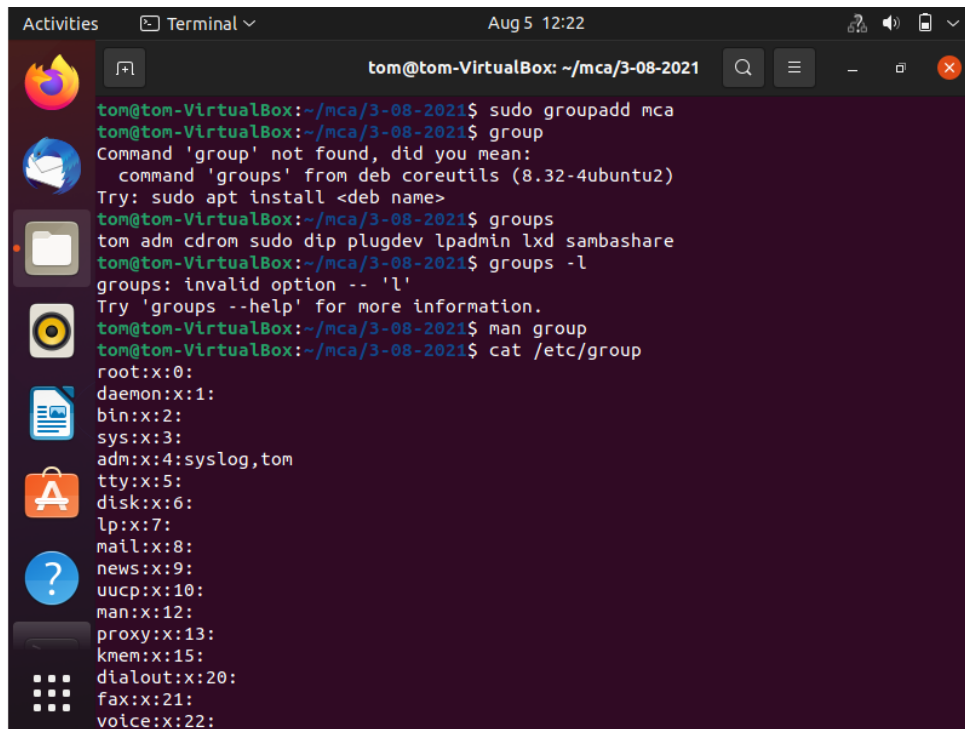
The groups command prints the names of the primary and any supplementary groups for each given username, or the current process if no names are given. If more than one name is given, the name of each user is printed before the list of that user's groups and the username is separated from the group list by a colon.

A terminal window titled 'tom@tom-VirtualBox: ~' showing the output of the 'groups' command. It displays the groups for the current user 'tom' and for the user 'tom' explicitly.

```
Activities Terminal Aug 12 15:27
tom@tom-VirtualBox: ~
tom@tom-VirtualBox:~$ groups
tom adm cdrom sudo dip plugdev lpadmin lxd sambashare
tom@tom-VirtualBox:~$ groups tom
tom : tom adm cdrom sudo dip plugdev lpadmin lxd sambashare
tom@tom-VirtualBox:~$
```

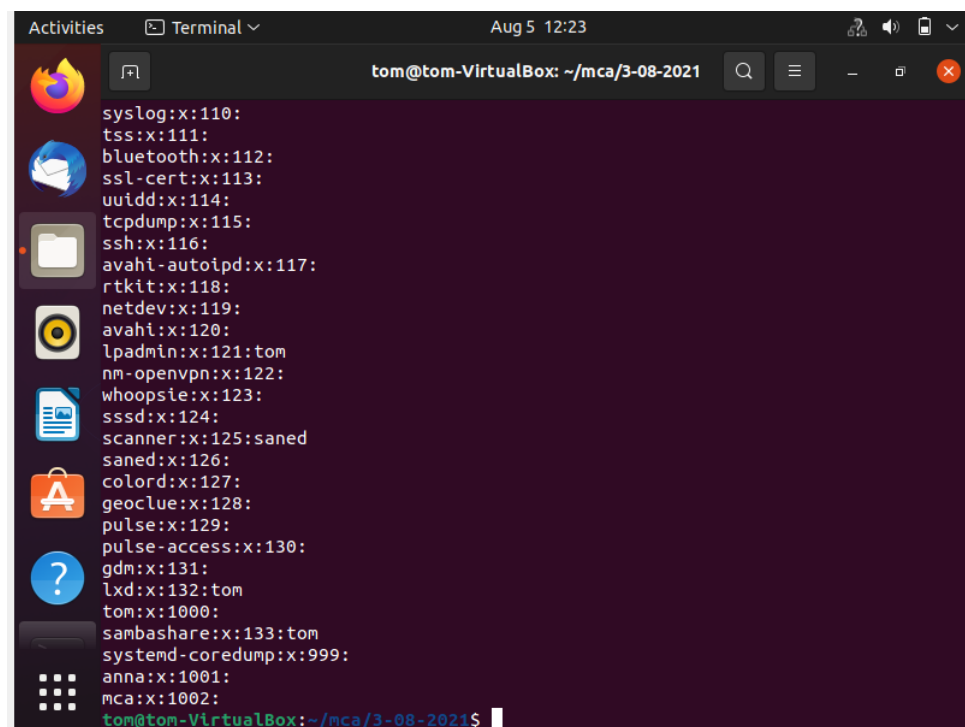
3. groupadd

The groupadd command is used for create a new group to create a new group in Linux



A terminal window titled 'tom@tom-VirtualBox: ~/mca/3-08-2021' showing the following commands and output:

```
tom@tom-VirtualBox:~/mca/3-08-2021$ sudo groupadd mca
tom@tom-VirtualBox:~/mca/3-08-2021$ group
Command 'group' not found, did you mean:
  command 'groups' from deb coreutils (8.32-4ubuntu2)
Try: sudo apt install <deb name>
tom@tom-VirtualBox:~/mca/3-08-2021$ groups
tom adm cdrom sudo dip plugdev lpadmin lxd sambashare
tom@tom-VirtualBox:~/mca/3-08-2021$ groups -l
groups: invalid option -- 'l'
Try 'groups --help' for more information.
tom@tom-VirtualBox:~/mca/3-08-2021$ man group
tom@tom-VirtualBox:~/mca/3-08-2021$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,tom
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
```

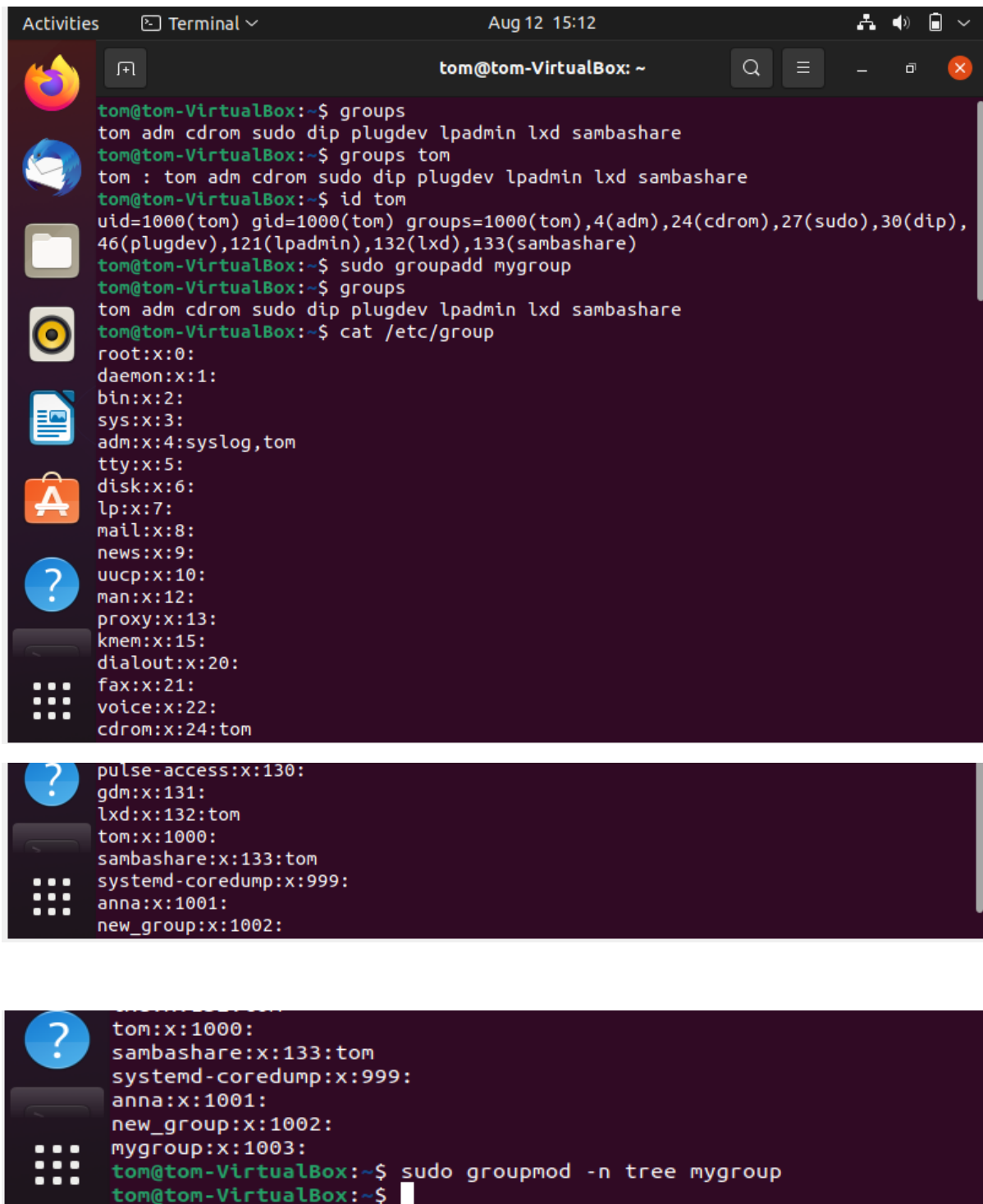


A terminal window titled 'tom@tom-VirtualBox: ~/mca/3-08-2021' showing the continuation of the group list from the previous screenshot:

```
syslog:x:110:
tss:x:111:
bluetooth:x:112:
ssl-cert:x:113:
uudd:x:114:
tcpdump:x:115:
ssh:x:116:
avahi-autoipd:x:117:
rtkit:x:118:
netdev:x:119:
avahi:x:120:
lpadmin:x:121:tom
nm-openvpn:x:122:
whoopsie:x:123:
sssd:x:124:
scanner:x:125:saned
saned:x:126:
colord:x:127:
geoclue:x:128:
pulse:x:129:
pulse-access:x:130:
gdm:x:131:
lxd:x:132:tom
tom:x:1000:
sambashare:x:133:tom
systemd-coredump:x:999:
anna:x:1001:
mca:x:1002:
tom@tom-VirtualBox:~/mca/3-08-2021$
```

4. groupmod

The `groupmod` command in Linux is used to modify or change the existing group on Linux system. It can be handled by superuser or root user. Basically, it modifies a group definition on the system by modifying the right entry in the database of the group. Syntax: `groupmod [option] GROUP`.

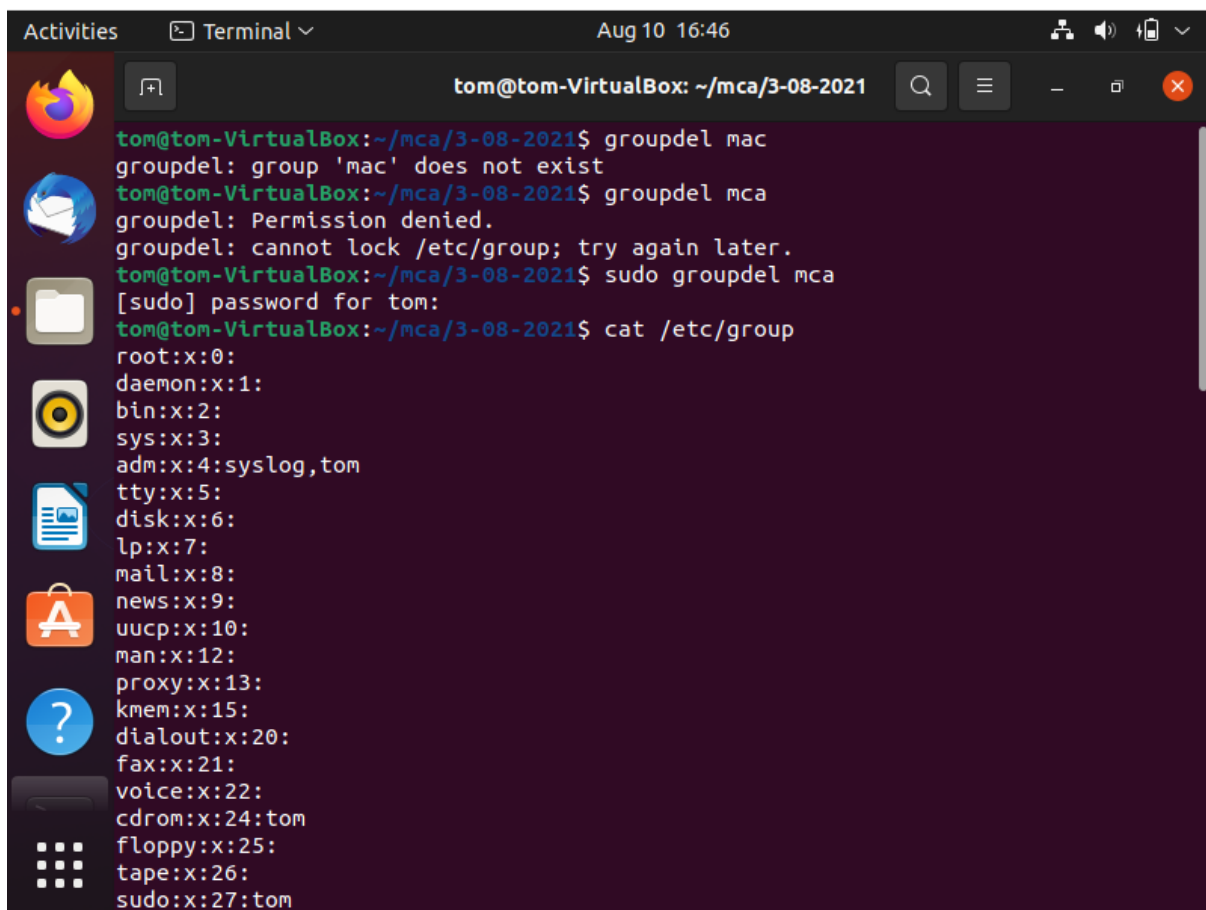


```
tom@tom-VirtualBox:~$ groups
tom adm cdrom sudo dip plugdev lpadmin lxd sambashare
tom@tom-VirtualBox:~$ groups tom
tom : tom adm cdrom sudo dip plugdev lpadmin lxd sambashare
tom@tom-VirtualBox:~$ id tom
uid=1000(tom) gid=1000(tom) groups=1000(tom),4(adm),24(cdrom),27(sudo),30(dip),
46(plugdev),121(lpadmin),132(lxd),133(sambashare)
tom@tom-VirtualBox:~$ sudo groupadd mygroup
tom@tom-VirtualBox:~$ groups
tom adm cdrom sudo dip plugdev lpadmin lxd sambashare
tom@tom-VirtualBox:~$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,tom
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:tom
pulse-access:x:130:
gdm:x:131:
lxd:x:132:tom
tom:x:1000:
sambashare:x:133:tom
systemd-coredump:x:999:
anna:x:1001:
new_group:x:1002:
tom:x:1000:
sambashare:x:133:tom
systemd-coredump:x:999:
anna:x:1001:
new_group:x:1002:
mygroup:x:1003:
tom@tom-VirtualBox:~$ sudo groupmod -n tree mygroup
tom@tom-VirtualBox:~$
```

```
ptsc:x:130:
gdm:x:131:
lxd:x:132:tom
tom:x:1000:
sambashare:x:133:tom
systemd-coredump:x:999:
anna:x:1001:
new_group:x:1002:
tree:x:1003:
tom@tom-VirtualBox:~$
```

5. groupdel

The groupdel command modifies the system account files, deleting all entries that refer to group. The named group must exist.

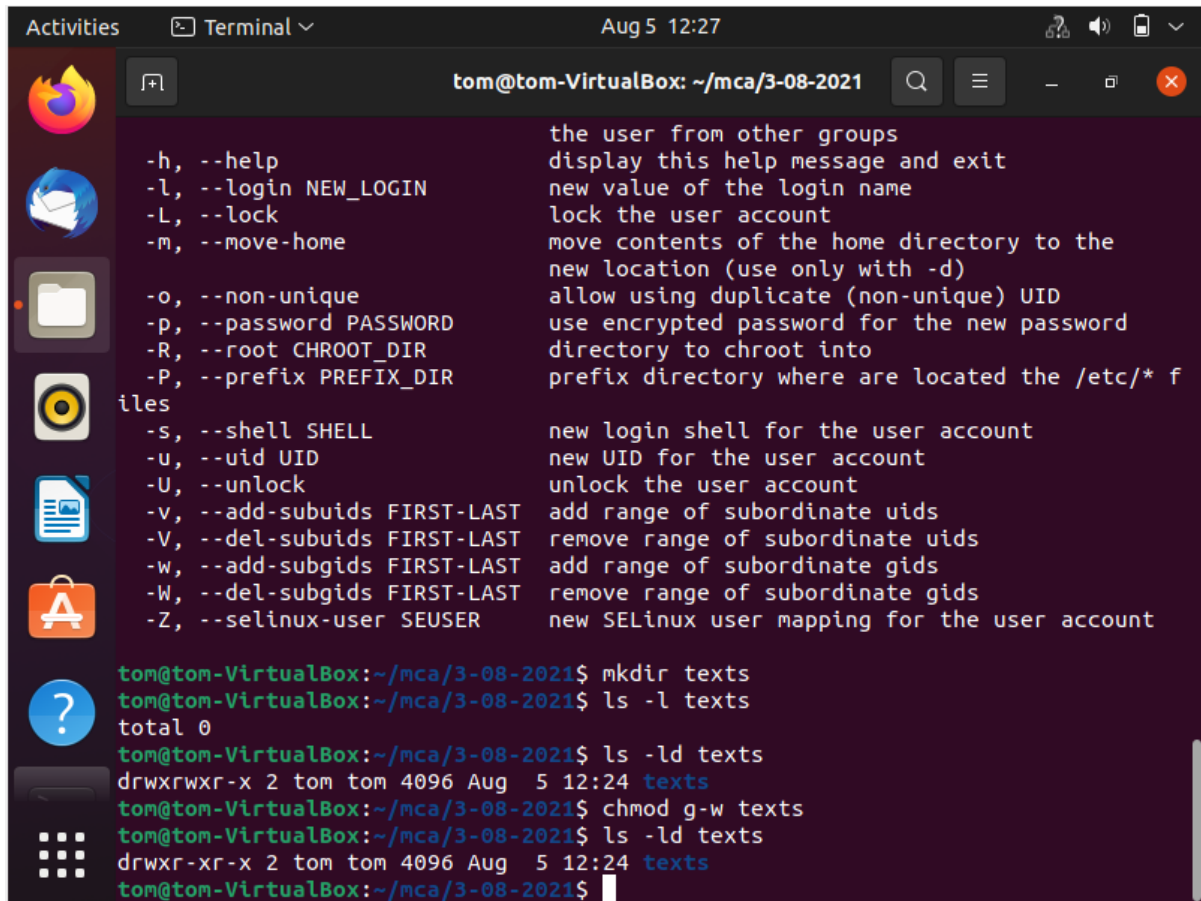


The image shows a terminal window titled "tom@tom-VirtualBox: ~/mca/3-08-2021" with a search bar and window controls. The terminal output shows the following sequence of commands and responses:

```
tom@tom-VirtualBox:~/mca/3-08-2021$ groupdel mac
groupdel: group 'mac' does not exist
tom@tom-VirtualBox:~/mca/3-08-2021$ groupdel mca
groupdel: Permission denied.
groupdel: cannot lock /etc/group; try again later.
tom@tom-VirtualBox:~/mca/3-08-2021$ sudo groupdel mca
[sudo] password for tom:
tom@tom-VirtualBox:~/mca/3-08-2021$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,tom
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:tom
floppy:x:25:
tape:x:26:
sudo:x:27:tom
```

6. chmod

The chmod command is used to change the access mode of a file.

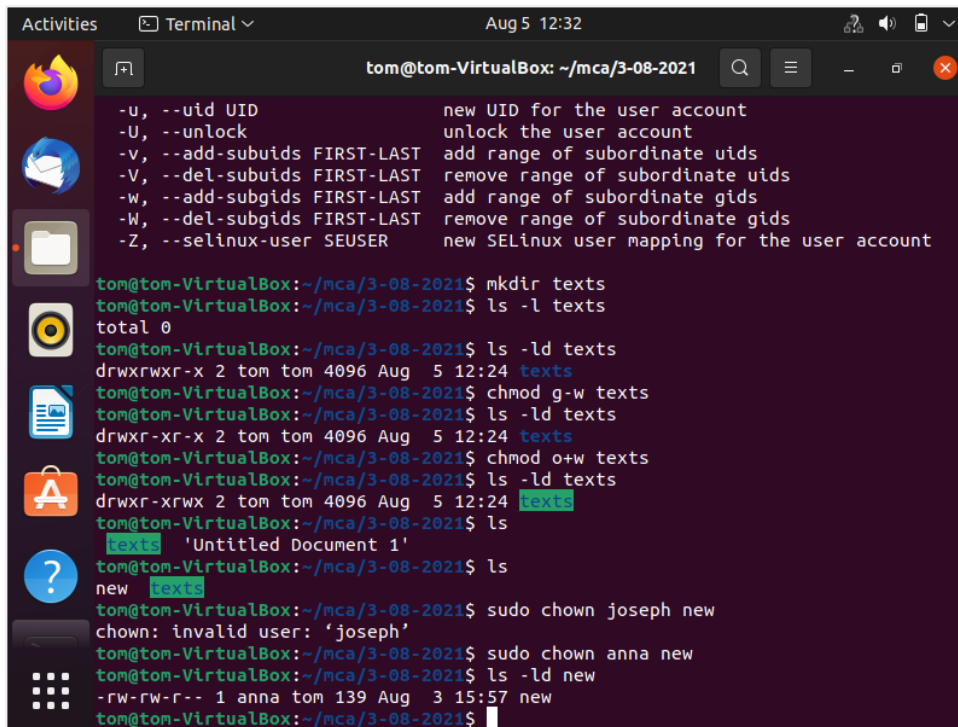


```
tom@tom-VirtualBox: ~/mca/3-08-2021
-h, --help                the user from other groups
-l, --login NEW_LOGIN     display this help message and exit
-L, --lock                new value of the login name
-m, --move-home           lock the user account
                           move contents of the home directory to the
                           new location (use only with -d)
-o, --non-unique          allow using duplicate (non-unique) UID
-p, --password PASSWORD   use encrypted password for the new password
-R, --root CHROOT_DIR     directory to chroot into
-P, --prefix PREFIX_DIR   prefix directory where are located the /etc/* f
iles
-s, --shell SHELL         new login shell for the user account
-u, --uid UID             new UID for the user account
-U, --unlock              unlock the user account
-v, --add-subuids FIRST-LAST add range of subordinate uids
-V, --del-subuids FIRST-LAST remove range of subordinate uids
-w, --add-subgids FIRST-LAST add range of subordinate gids
-W, --del-subgids FIRST-LAST remove range of subordinate gids
-Z, --selinux-user SEUSER new SELinux user mapping for the user account

tom@tom-VirtualBox:~/mca/3-08-2021$ mkdir texts
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -l texts
total 0
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -ld texts
drwxrwxr-x 2 tom tom 4096 Aug  5 12:24 texts
tom@tom-VirtualBox:~/mca/3-08-2021$ chmod g-w texts
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -ld texts
drwxr-xr-x 2 tom tom 4096 Aug  5 12:24 texts
tom@tom-VirtualBox:~/mca/3-08-2021$
```

7. chown

The chown command allows you to change the user and/or group ownership of a given file, directory, or symbolic link. In Linux, all files are associated with an owner and a group and assigned with permission access rights for the file owner, the group members, and others.

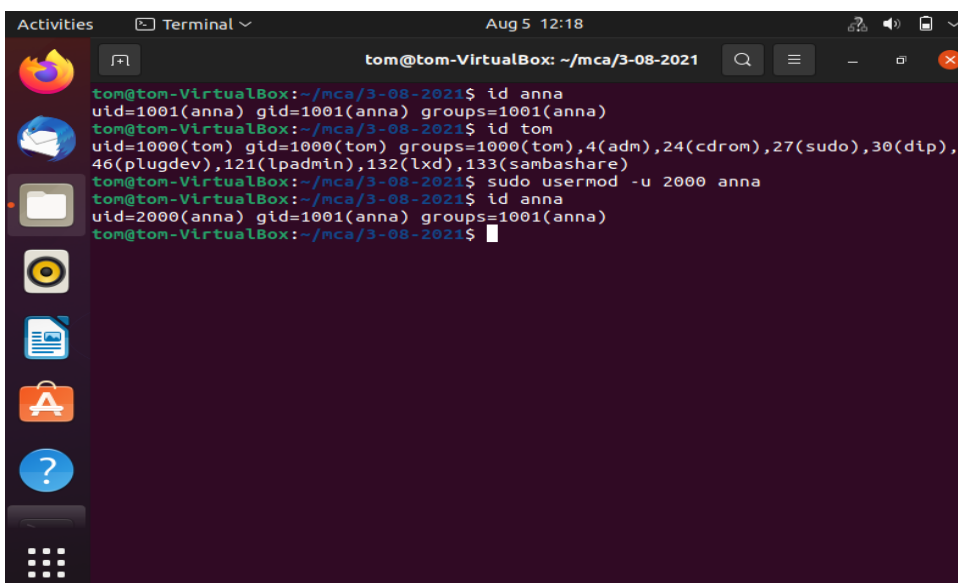
A terminal window titled 'tom@tom-VirtualBox: ~/mca/3-08-2021' showing the execution of the chown command. It lists the options for chown: -u, --uid UID; -U, --unlock; -v, --add-subuids FIRST-LAST; -V, --del-subuids FIRST-LAST; -w, --add-subgids FIRST-LAST; -W, --del-subgids FIRST-LAST; and -Z, --selinux-user SEUSER. Then, it shows the creation of a directory 'texts', its permissions, and the use of chown to change ownership to 'anna' and 'tom'. Finally, it shows an attempt to change ownership to 'joseph' which fails with 'chown: invalid user: 'joseph'', and then successfully to 'anna' using 'sudo chown anna new'.

```
Activities Terminal Aug 5 12:32
tom@tom-VirtualBox: ~/mca/3-08-2021
-u, --uid UID                new UID for the user account
-U, --unlock                 unlock the user account
-v, --add-subuids FIRST-LAST add range of subordinate uids
-V, --del-subuids FIRST-LAST remove range of subordinate uids
-w, --add-subgids FIRST-LAST add range of subordinate gids
-W, --del-subgids FIRST-LAST remove range of subordinate gids
-Z, --selinux-user SEUSER    new SELinux user mapping for the user account

tom@tom-VirtualBox:~/mca/3-08-2021$ mkdir texts
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -l texts
total 0
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -ld texts
drwxrwxr-x 2 tom tom 4096 Aug  5 12:24 texts
tom@tom-VirtualBox:~/mca/3-08-2021$ chmod g-w texts
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -ld texts
drwxr-xr-x 2 tom tom 4096 Aug  5 12:24 texts
tom@tom-VirtualBox:~/mca/3-08-2021$ chmod o+w texts
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -ld texts
drwxr-xrwx 2 tom tom 4096 Aug  5 12:24 texts
tom@tom-VirtualBox:~/mca/3-08-2021$ ls
texts 'Untitled Document 1'
tom@tom-VirtualBox:~/mca/3-08-2021$ ls
new texts
tom@tom-VirtualBox:~/mca/3-08-2021$ sudo chown joseph new
chown: invalid user: 'joseph'
tom@tom-VirtualBox:~/mca/3-08-2021$ sudo chown anna new
tom@tom-VirtualBox:~/mca/3-08-2021$ ls -ld new
-rw-rw-r-- 1 anna tom 139 Aug  3 15:57 new
tom@tom-VirtualBox:~/mca/3-08-2021$
```

8. id

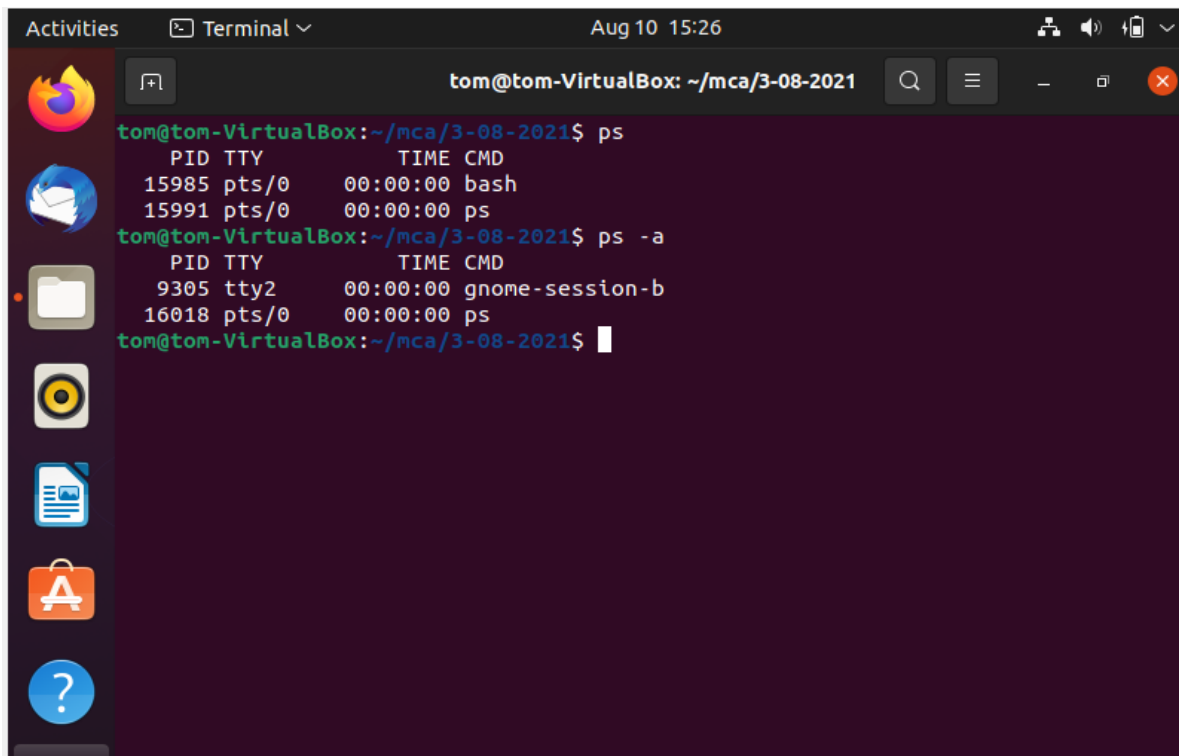
The id command in Linux is used to find out user and group names and numeric ID's (UID or group ID) of the current user or any other user in the server. This command is useful to find out the following information as listed below: User name and real user id.

A terminal window titled 'tom@tom-VirtualBox: ~/mca/3-08-2021' showing the execution of the id command. It shows the output of 'id anna' and 'id tom'. Then, it shows the use of 'sudo usermod -u 2000 anna' to change the UID of 'anna' to 2000. Finally, it shows the output of 'id anna' after the change, confirming the new UID.

```
Activities Terminal Aug 5 12:18
tom@tom-VirtualBox: ~/mca/3-08-2021
tom@tom-VirtualBox:~/mca/3-08-2021$ id anna
uid=1001(anna) gid=1001(anna) groups=1001(anna)
tom@tom-VirtualBox:~/mca/3-08-2021$ id tom
uid=1000(tom) gid=1000(tom) groups=1000(tom),4(adm),24(cdrom),27(sudo),30(dip),
46(plugdev),121(lpadmin),132(lxd),133(sambashare)
tom@tom-VirtualBox:~/mca/3-08-2021$ sudo usermod -u 2000 anna
tom@tom-VirtualBox:~/mca/3-08-2021$ id anna
uid=2000(anna) gid=1001(anna) groups=1001(anna)
tom@tom-VirtualBox:~/mca/3-08-2021$
```

9. ps

The `ps` command in Linux is used to display about running processes on the system. You can get information like process ID (PID) for the processes you or any other user is running on the same Linux system.

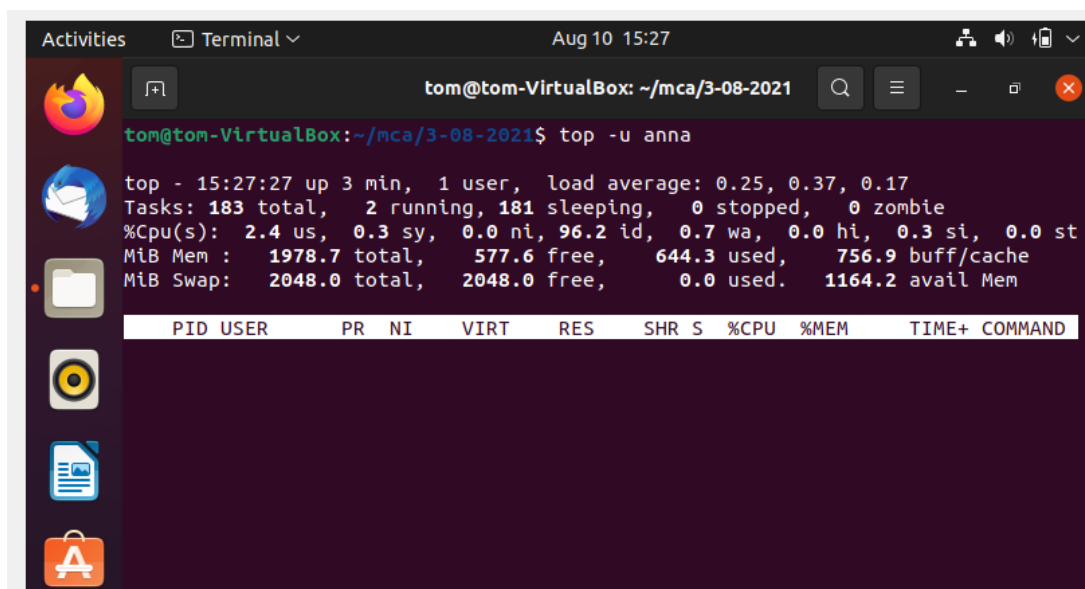


A terminal window titled 'tom@tom-VirtualBox: ~/mca/3-08-2021' showing the output of the `ps` and `ps -a` commands. The `ps` command shows the current user's processes, and `ps -a` shows all processes.

```
tom@tom-VirtualBox:~/mca/3-08-2021$ ps
  PID TTY          TIME CMD
 15985 pts/0    00:00:00 bash
 15991 pts/0    00:00:00 ps
tom@tom-VirtualBox:~/mca/3-08-2021$ ps -a
  PID TTY          TIME CMD
  9305 tty2      00:00:00 gnome-session-b
 16018 pts/0    00:00:00 ps
tom@tom-VirtualBox:~/mca/3-08-2021$
```

10. top

The Linux `top` command shows the running processes within your Linux environment that consume the most system resources.



A terminal window titled 'tom@tom-VirtualBox: ~/mca/3-08-2021' showing the output of the `top -u anna` command. The output displays system statistics and a table of running processes.

```
tom@tom-VirtualBox:~/mca/3-08-2021$ top -u anna
top - 15:27:27 up 3 min,  1 user,  load average: 0.25, 0.37, 0.17
Tasks: 183 total,  2 running, 181 sleeping,  0 stopped,  0 zombie
%Cpu(s):  2.4 us,  0.3 sy,  0.0 ni, 96.2 id,  0.7 wa,  0.0 hi,  0.3 si,  0.0 st
MiB Mem : 1978.7 total,  577.6 free,  644.3 used,  756.9 buff/cache
MiB Swap: 2048.0 total,  2048.0 free,  0.0 used.  1164.2 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
    1  anna      20   0    4096   1080  1024  S   0.0   0.0   0:00.0 top
```