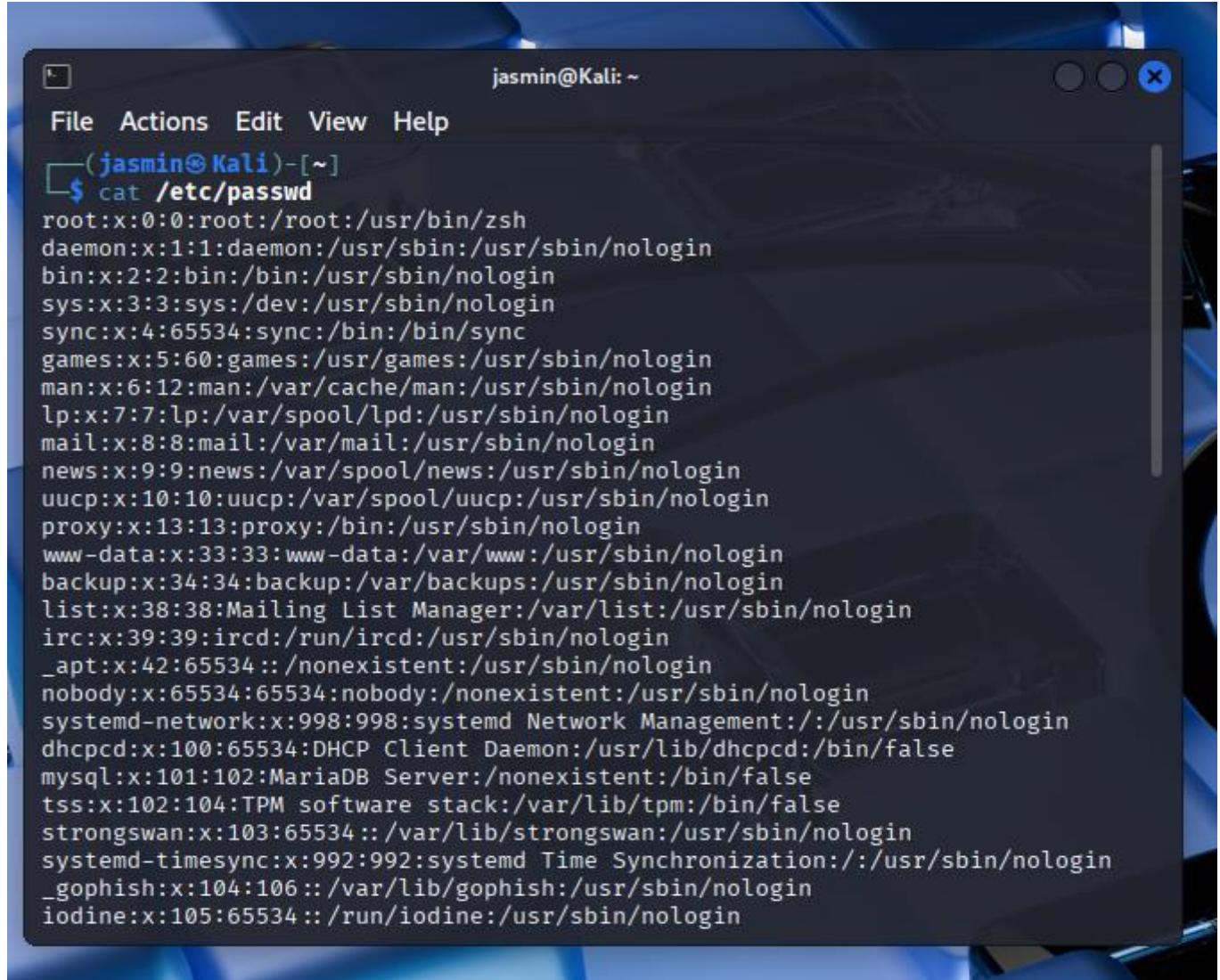


Task A:

Step 1: The command that I used to display user account information is “`cat /etc/passwd`”.

This command is used to display the users information such as UID’s, usernames, and home directories in the terminal.

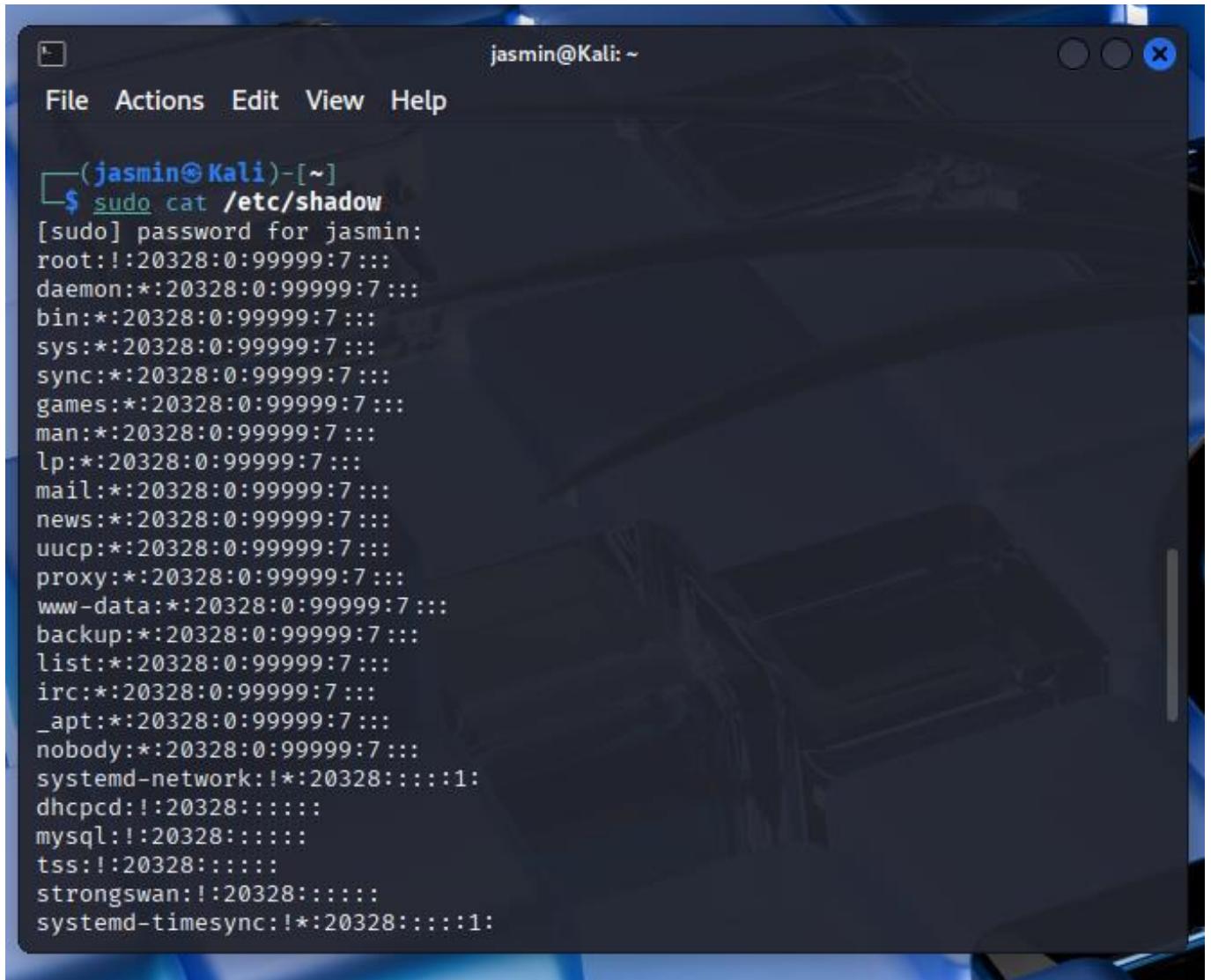


The screenshot shows a terminal window with a dark blue background. The title bar says "jasmin@Kali: ~". The menu bar includes "File", "Actions", "Edit", "View", and "Help". The command line shows the user is in their home directory (~) and has run the command "cat /etc/passwd". The output lists numerous user accounts, each consisting of a colon-separated string of values: username:password:UID:GID:realname:home_directory:shell. Some accounts like "root" and "daemon" have empty password fields (x). Many accounts have UID and GID values of 65534, and their shells are set to "/bin/false". The output is as follows:

```
root:x:0:0:root:/root:/usr/bin/zsh
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:/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:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
dhpcd:x:100:65534:DHCP Client Daemon:/usr/lib/dhpcd:/bin/false
mysql:x:101:102:MariaDB Server:/nonexistent:/bin/false
tss:x:102:104:TPM software stack:/var/lib/tpm:/bin/false
strongswan:x:103:65534::/var/lib/strongswan:/usr/sbin/nologin
systemd-timesync:x:992:992:systemd Time Synchronization:/:/usr/sbin/nologin
_gophish:x:104:106::/var/lib/gophish:/usr/sbin/nologin
iodine:x:105:65534::/run/iodine:/usr/sbin/nologin
```

Step 2. The command that I used to display password information for the user is “`sudo cat /etc/shadow`”. Etc/shadow is used to display group information including admin and group

passwords. Passwords are always stored in the etc/shadow file which is only accessible by the root. The sudo command runs commands at the root.

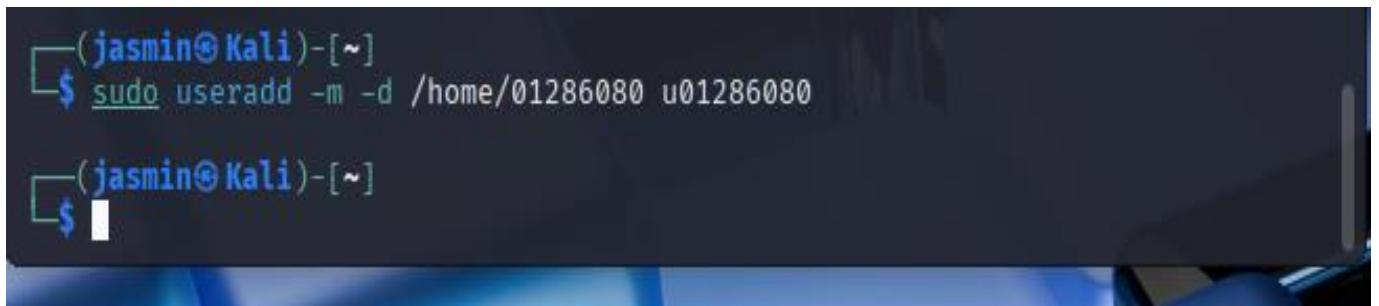


The screenshot shows a terminal window with a dark blue background. The title bar reads "jasmin@Kali: ~". The menu bar includes "File", "Actions", "Edit", "View", and "Help". The terminal window displays the following command and its output:

```
(jasmin@Kali)-[~]
$ sudo cat /etc/shadow
[sudo] password for jasmin:
root:!20328:0:99999:7:::
daemon:*:20328:0:99999:7:::
bin:*:20328:0:99999:7:::
sys:*:20328:0:99999:7:::
sync:*:20328:0:99999:7:::
games:*:20328:0:99999:7:::
man:*:20328:0:99999:7:::
lp:*:20328:0:99999:7:::
mail:*:20328:0:99999:7:::
news:*:20328:0:99999:7:::
uucp:*:20328:0:99999:7:::
proxy:*:20328:0:99999:7:::
www-data:!:20328:0:99999:7:::
backup:*:20328:0:99999:7:::
list:*:20328:0:99999:7:::
irc:*:20328:0:99999:7:::
_apt:*:20328:0:99999:7:::
nobody:*:20328:0:99999:7:::
systemd-network:!*:20328:::::1:
dhcpcd:!:20328:::::
mysql:!:20328:::::
tss:!:20328:::::
strongswan:!:20328:::::
systemd-timesync:!*:20328:::::1:
```

Step 3. The command that I would use to create a new user under my MIDAS ID

01286080 would be “`sudo useradd -m -d /home/02180680 02180680`” The `sudo useradd` command is used to add users to a group.



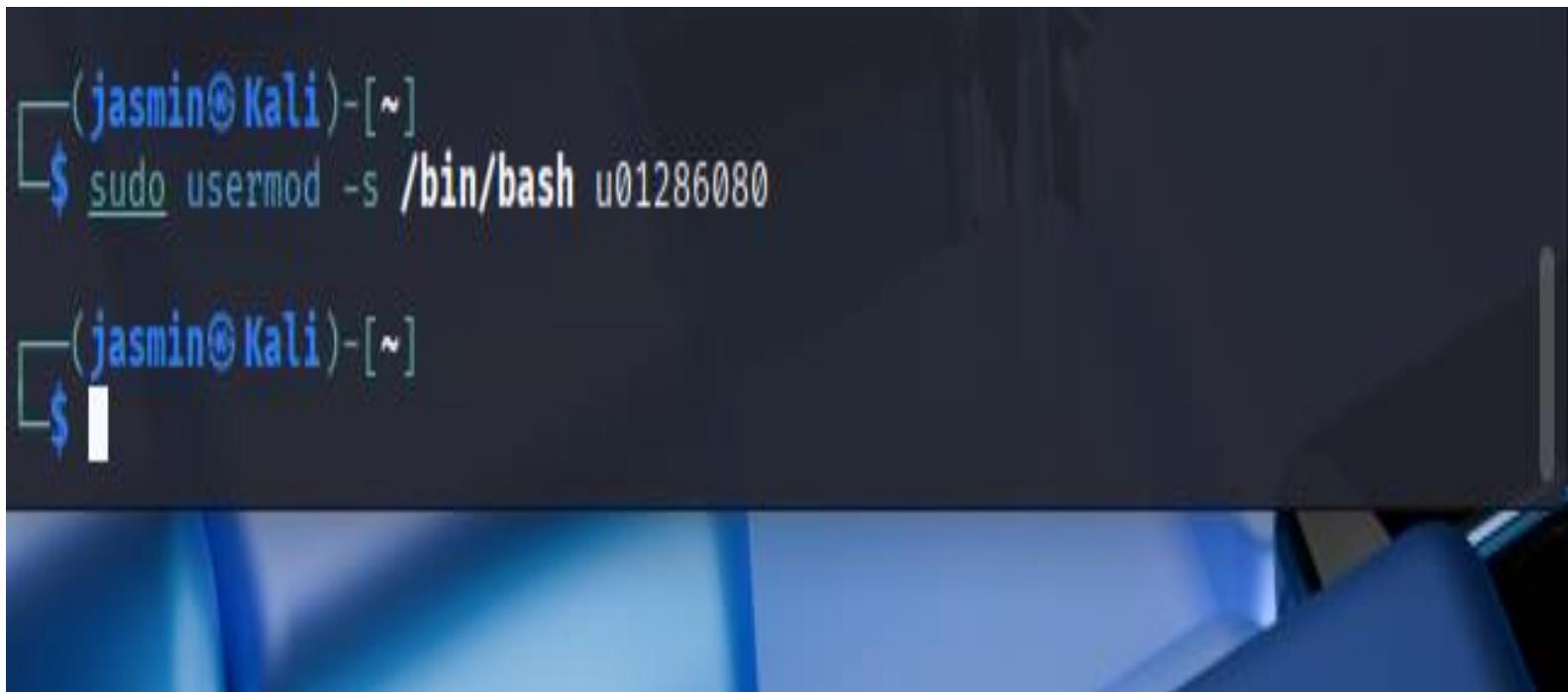
```
(jasmin㉿Kali)-[~]
$ sudo useradd -m -d /home/01286080 u01286080

(jasmin㉿Kali)-[~]
$
```

Step 4. To set a password for the user I used the command “`sudo passwd 01286080`”. The “`sudo passwd`” command is used to change the users password and to specify which user it is followed by the users username.

```
(jasmin㉿Kali)-[~]
└$ sudo passwd u01286080
New password:
Retype new password:
passwd: password updated successfully
```

Step 5. The command that I used to set bash shell as the default log in is “`sudo usermod -s /bin/bash u01286080`”. The sudo usermod command is used to add or change users in a group.



```
(jasmin㉿Kali)-[~]$ sudo usermod -s /bin/bash u01286080
```

Step 6. To display the user information using grep, the command that I executed is “grep u01286080 /etc/passwd”. All information pertaining to the user is stored in /etc/passwd and the “grep” command is used to verify information.

```
(jasmin㉿Kali)-[~]
$ grep u01286080 /etc/passwd
u01286080:x:1001:1002::/home/01286080:/bin/bash

(jasmin㉿Kali)-[~]
$ []
```

Step 7. The command that I used to add the new user 01286080 to the sudo group without overriding the existing group membership is “`sudo usermod -aG sudo 01286080`”. The command `usermod -aG` is used to append a user to groups without removing existing ones.

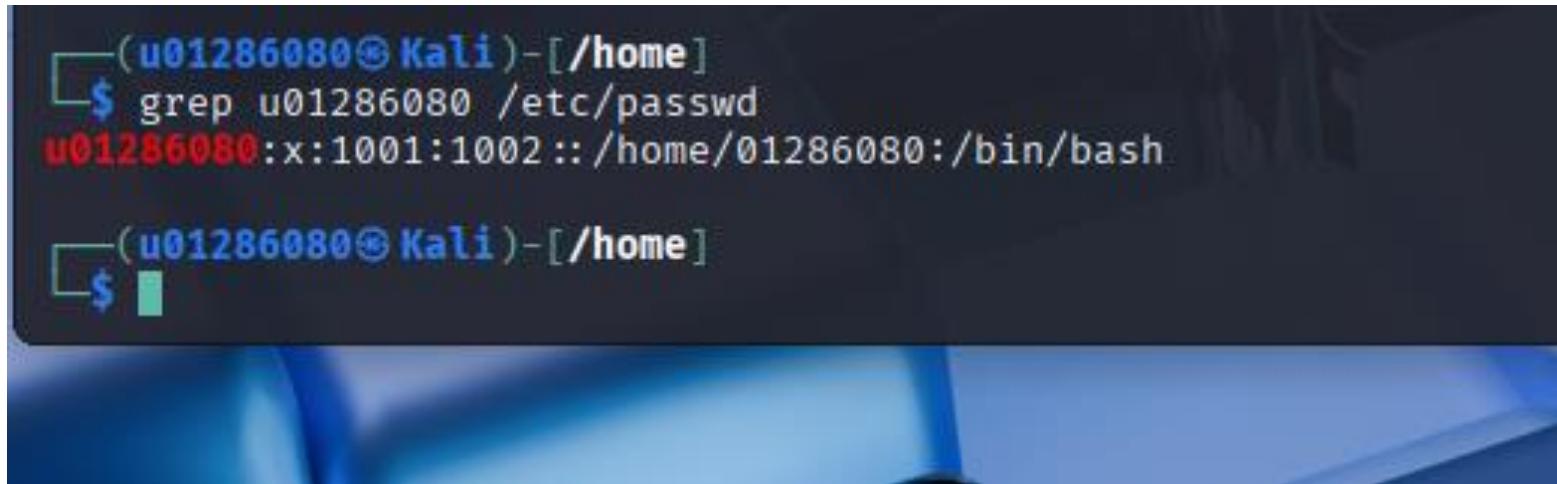
```
min㉿Kali)-[ ~]
└ usermod -aG sudo u01286080
min㉿Kali)-[ ~]
```

Step 8. The command that i used to switch to the new users account was the **-su** command. The command “su” means switch user which allowed me to switch from “jasmin” to “u01286080”.

```
[jasmin@Kali)-[~]
$ su - u01286080
Password:
[u01286080@Kali)-[~]
$
```

Task B

1. I used two different commands to take me back to the home directory and determine which shell I was in . The first command I used was “`cd /home`”. The `cd /home` command is used to switch back over to the home directory. To determine the shell I am using, I used the command “`grep u01286080 /etc/passwd`”. This command is used to verify information to include shell location.



```
(u01286080@Kali)-[~/home]
$ grep u01286080 /etc/passwd
u01286080:x:1001:1002 ::/home/01286080:/bin/bash

(u01286080@Kali)-[~/home]
$
```

Step 2. To display the current user's ID and group membership, I used the "id" command. As see below the "id" command is executed to display identifying user and group information.

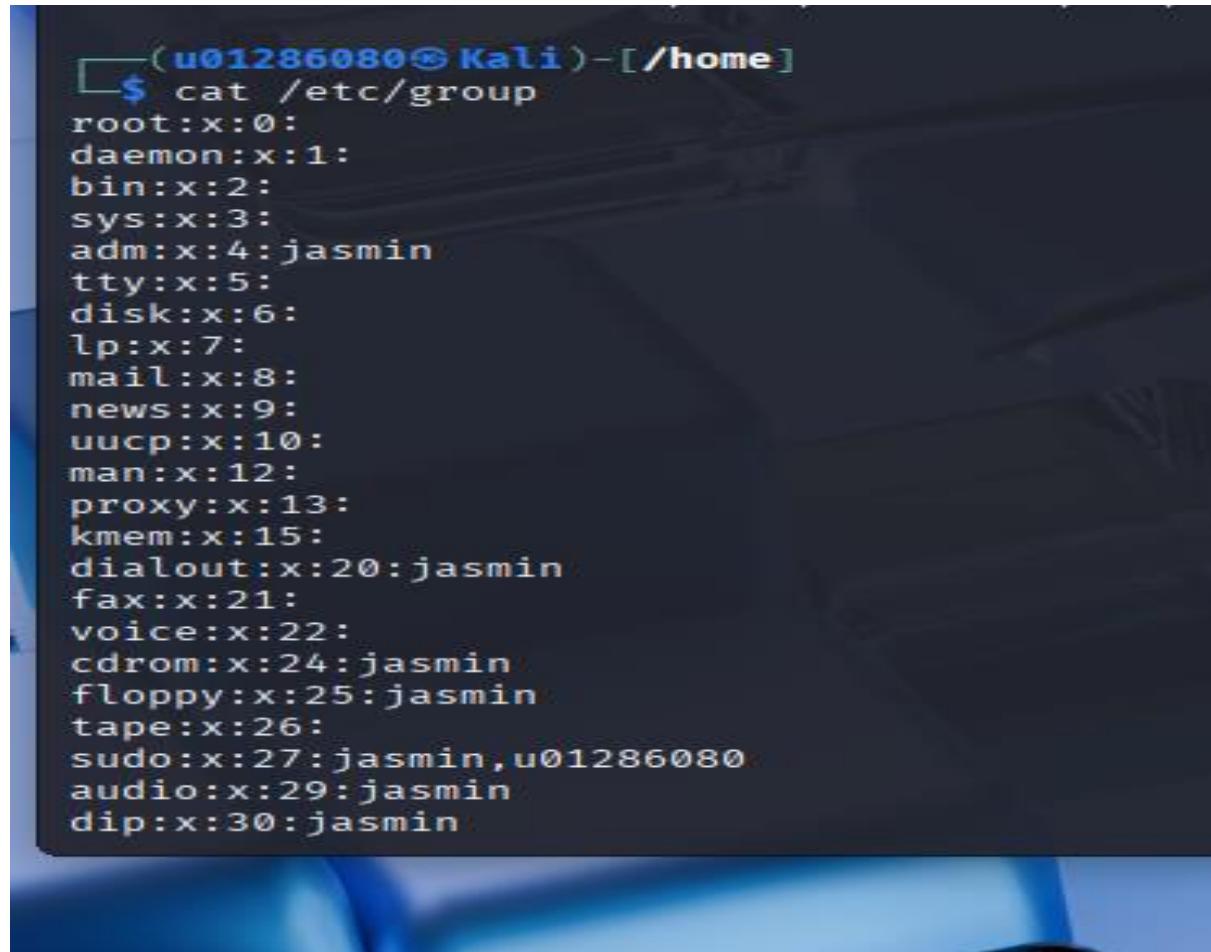
```
└─(u01286080㉿Kali)-[~/home]
└─$ id
uid=1001(u01286080) gid=1002(u01286080) groups=1002(u01286080),27(sudo)

└─(u01286080㉿Kali)-[~/home]
└─$ █
```

Step 3. To Display the group membership of the root account I used the “`cat /etc/passwd`” command. This command is used to define the users primary group membership.

```
(u01286080㉿Kali)-[~/home]
$ cat /etc/passwd
root:x:0:0:root:/root:/usr/bin/zsh
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:/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:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534 ::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
dhcpcd:x:100:65534:DHCP Client Daemon:/usr/lib/dhcpcd:/bin/false
mysql:x:101:102:MariaDB Server:/nonexistent:/bin/false
tss:x:102:104:TPM software stack:/var/lib/tpm:/bin/false
strongswan:x:103:65534 ::/var/lib/strongswan:/usr/sbin/nologin
systemd-timesync:x:992:992:systemd Time Synchronization:/:/usr/sbin/nologin
_gophish:x:104:106 ::/var/lib/gophish:/usr/sbin/nologin
```

4. to determine the user owner and group owner of the /etc/group I used the “`cat /etc/group`” command. This command is used to display information pertaining to the group to include group name, GID, and membership.



```
(u01286080㉿Kali)-[~/home]
$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:jasmin
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:jasmin
fax:x:21:
voice:x:22:
cdrom:x:24:jasmin
floppy:x:25:jasmin
tape:x:26:
sudo:x:27:jasmin,u01286080
audio:x:29:jasmin
dip:x:30:jasmin
```

5. The command that I used to create a new group named test and use my UIN as the GID is “`sudo groupadd -g 01286080 test`” The command “`sudo groupadd`” is used to create groups and `-g` was used to adding the GID.



```
(u01286080㉿Kali)-[~/home]
$ sudo groupadd -g 01286080 test

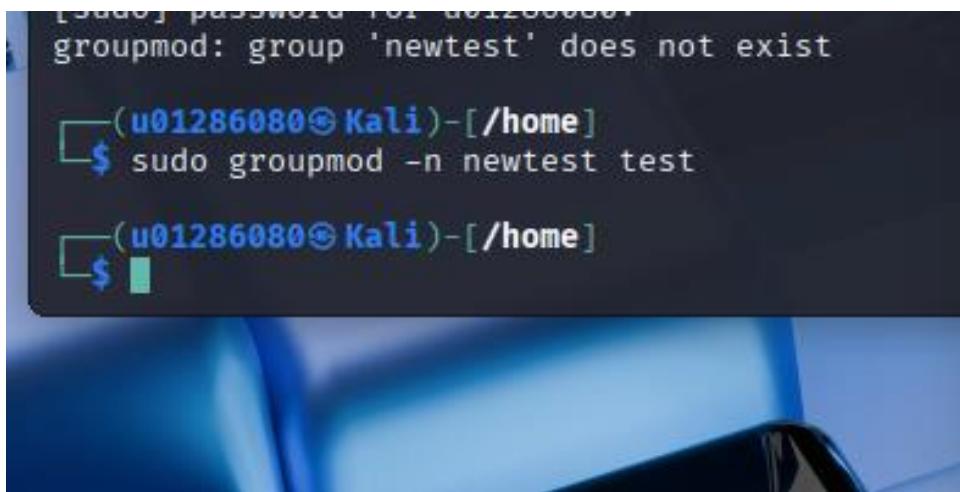
(u01286080㉿Kali)-[~/home]
$
```

Step 6. To display the group account information for the test group using grep I used the command “`grep test /etc/group`”. Grep is used to verify information specifically for the test group.

```
(u01286080㉿Kali)-[~/home]
$ grep test /etc/group
test:x:1286080:

(u01286080㉿Kali)-[~/home]
```

7. To change the group name "test" to "newtest" I used the command `groupmod -n newtest test`. This command changes the name of the group from "test" to "newtest"



```
[sudo] password for av1286080:
groupmod: group 'newtest' does not exist

[u01286080@Kali ~]$ sudo groupmod -n newtest test

[u01286080@Kali ~]$
```

8. To add the current account as a secondary member of the newtest group without overriding this user's current group membership I used the command “sudo usermod -aG newtest u01286080. This -aG command is used to add yourself to the group.



```
(u01286080㉿Kali)-[~/home]
$ sudo usermod -aG newtest u01286080

(u01286080㉿Kali)-[~/home]
$
```

9. To create a new file named testfile in the account's home directory, then change the group owner to newtest I used two different commands. The first command I used to create the file is `touch ~/testfile`. The touch command is used to create files and `sudo chgrp newtest ~/testfile` was used to change the group owner to newtest. The sudo chgrp command is used to change ownership of the file.

```
└─(u01286080㉿Kali)-[~/home]
$ touch ~/testfile

└─(u01286080㉿Kali)-[~/home]
$ sudo chgrp newtest ~/testfile
```

Step 10. The command I used to display the user owner and group owner information of the file testfile `ls -l ~/testfile`. The `ls -l` command is used to identify the current group information of the file.

```
└─(u01286080㉿Kali)-[~/home]
$ ls -l ~/testfile
-rw-rw-r-- 1 u01286080 newtest 0 Sep 21 17:44 /home/u01286080/testfile

└─(u01286080㉿Kali)-[~/home]
$
```

Step 11. The First command that I used to delete the newtest group is `sudo groupdel newtest`. The groupdel command is used to delete groups. I then used the previous command `ls -l ~/testfile` and this was my result.

```
(u01286080㉿Kali)-[~/home]
$ sudo groupdel newtest

(u01286080㉿Kali)-[~/home]
$ ls -l ~/testfile
-rw-rw-r-- 1 u01286080 1286080 0 Sep 21 17:44 /home/01286080/testfile

(u01286080㉿Kali)-[~/home]
$
```

12. I tried using the command below which did not work. The `rm` command is typically used to remove user information.

```
└─(u01286080㉿Kali)-[/home]
└─$ rm /home/01286080
rm: cannot remove '/home/01286080': Is a directory

└─(u01286080㉿Kali)-[/home]
└─$ sudo gpasswd -d 01286080
Usage: gpasswd [option] GROUP
```