

For above Linux instance create a new user call Arun and he should be part of DBA group and secondary group should be eranki and make sure that group has a group ID of 1001

Steps to create user, Group and add user in group:

1. To create a user with user_name (Arun and eranki):

```
$ sudo useradd Arun  
$ sudo useradd eranki
```

2. To create a new group with group_name(DBA and second-grp):

```
$ sudo groupadd DBA  
$ sudo groupadd second-grp
```

3. We can use the **usermod** command to add a user to a group:

```
$ sudo usermod -a -G DBA  
$ sudo usermod -a -G second-grp
```

The *-a* option tells *usermod* for appending and the *-G* option tells *usermod* we are appending to the group name that follows the option.

4. To change group ID:

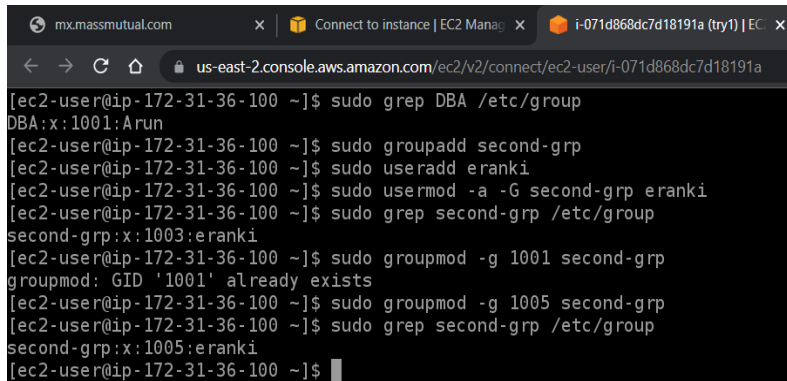
```
$ sudo groupmod -g 1005 second-grp
```

5. The command is used to list pertinent information about the group:

```
$ sudo grep DBA /etc/group
```

```
$ sudo grep second-grp /etc/group
```

Output:



The screenshot shows a terminal window within the AWS Management Console. The terminal output is as follows:

```
[ec2-user@ip-172-31-36-100 ~]$ sudo grep DBA /etc/group
DBA:x:1001:Arun
[ec2-user@ip-172-31-36-100 ~]$ sudo groupadd second-grp
[ec2-user@ip-172-31-36-100 ~]$ sudo useradd eranki
[ec2-user@ip-172-31-36-100 ~]$ sudo usermod -a -G second-grp eranki
[ec2-user@ip-172-31-36-100 ~]$ sudo grep second-grp /etc/group
second-grp:x:1003:eranki
[ec2-user@ip-172-31-36-100 ~]$ sudo groupmod -g 1001 second-grp
groupmod: GID '1001' already exists
[ec2-user@ip-172-31-36-100 ~]$ sudo groupmod -g 1005 second-grp
[ec2-user@ip-172-31-36-100 ~]$ sudo grep second-grp /etc/group
second-grp:x:1005:eranki
[ec2-user@ip-172-31-36-100 ~]$
```

i-071d868dc7d18191a (try1)

Public IPs: 18.216.40.141 Private IPs: 172.31.36.100

