

Lab 4.7.3 Creating and Renaming Groups on a Linux System

From TestOut CompTIA Security+ Course

In this lab I will be creating new groups and renaming others, on a Linux Server System.

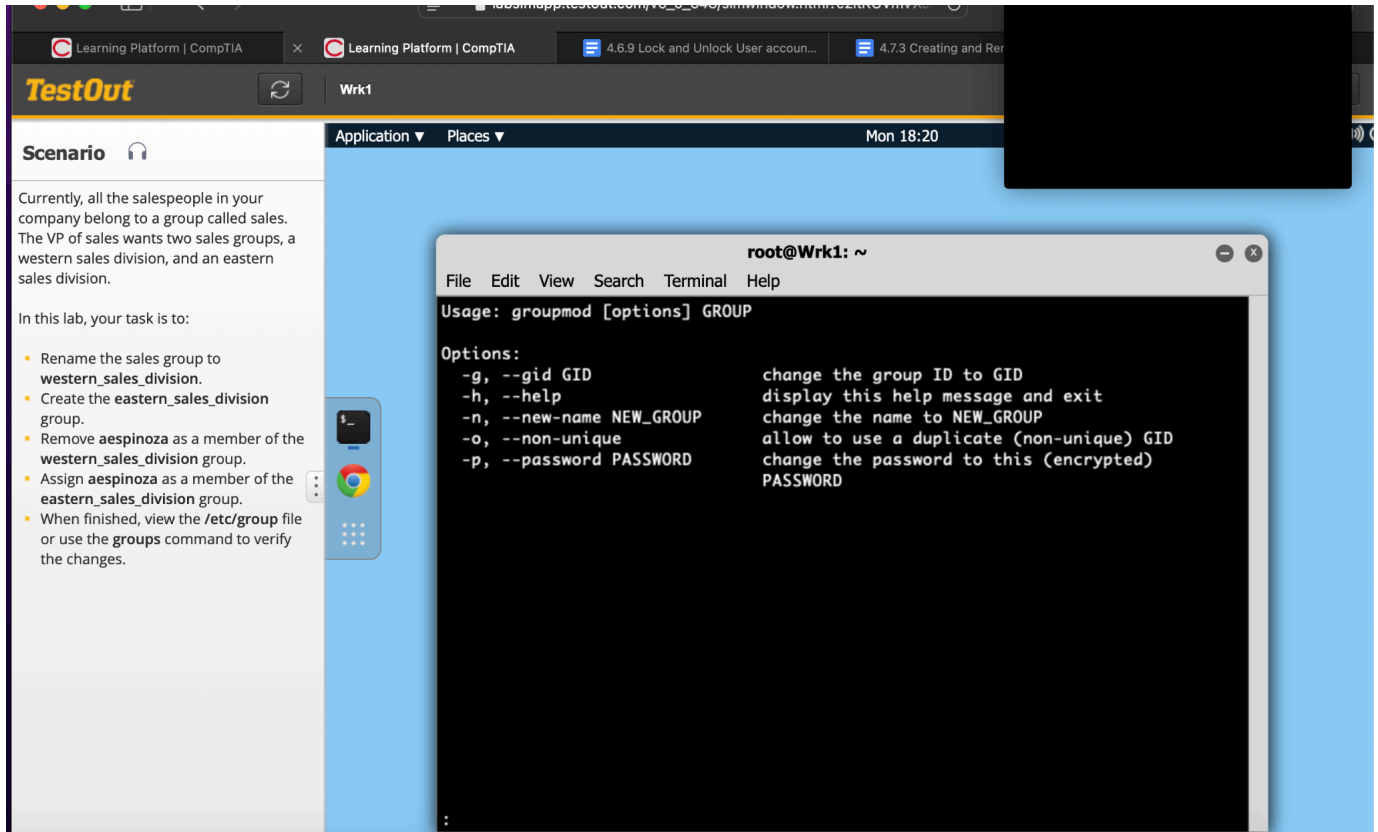
The scenario for this lab is as follows:

“Currently, all the salespeople in your company belong to a group called sales. The VP of sales wants two sales groups, a western sales division, and an eastern sales division.

In this lab, your task is to:

1. **Rename the sales group to western_sales_division.**
2. **Create the eastern_sales_divisiongroup.**
3. **Remove aespinoza as a member of the western_sales_division group.**
4. **Assign aespinoza as a member of the eastern_sales_division group.**
5. **When finished, view the /etc/group file or use the groups command to verify the changes.”**

For our first task we need to rename the group on this system called “sales” to “western_sales_division.” Since all users apart of sales will be transferred to the new group we can use the “groupmod” command that’s built into Linux. To see what we can do with the groupmod command type “man groupmod” at the shell prompt:



We can see that the only flag of use is “-n” which is “change the name to NEW_GROUP. So in order to rename the sales group we will issue the command:

```
groupmod -n western_sales_division sales
```

Note that you put the new name first because you have to put your flags before the GROUP argument.

```
root@Wrk1:~# groupmod -n western_sales_division sales
root@Wrk1:~#
```


Command success! Now, moving on to step #2, we need to create a new group called “eastern_sales_division.” Looking back at the groupmod man page I see that there are no options to create a new group. For this, we need to use a different command called “groupadd.” It functions much like usermod and useradd binaries. Query the man page for groupadd to see options:



```
root@Wrk1: ~  
File Edit View Search Terminal Help  
Usage: groupadd [options] GROUP  
  
Options:  
-f, --force          exit successfully if the group already exists,  
                     and cancel -g if the GID is already used  
-g, --gid GID        use GID for the new group  
-h, --help           display this help message and exit  
-K, --key KEY=VALUE  override /etc/login.defs defaults  
-o, --non-unique      allow to create groups with duplicate  
                     (non-unique) GID  
-p, --password PASSWORD use this encrypted password for the new group  
-r, --system          create a system account
```

In this case we don't need to supply any flags. Simply type:

```
#groupadd eastern_sales_division
```



```
root@Wrk1:~# groupadd eastern_sales_division  
root@Wrk1:~# _
```

Now I see that the shell is prompting me for another command. This means we succeeded in creating this group!

Moving on to step #3. We are asked to remove a given user from the western_sales_division group. Note that we are asked to remove the user from the GROUP not the USER themselves! Since we are dealing with a single user let's see what the "usermod" command can do for us which is the same command except for users specifically. Type man usermod at the shell:

```
root@Wrk1: ~
File Edit View Search Terminal Help

Usage: usermod [options] LOGIN

Options:
  -c, --comment COMMENT          new value of the GECOS field
  -d, --home HOME_DIR            new home directory for the user account
  -e, --expiredate EXPIRE_DATE  set account expiration date to EXPIRE_DATE
  -f, --inactive INACTIVE        set password inactive after expiration
                                 to INACTIVE
  -g, --gid GROUP                force use GROUP as new primary group
  -G, --groups GROUPS            new list of supplementary GROUPS
  -a, --append                   append the user to the supplemental GROUPS
                                 mentioned by the -G option without removing
                                 him/her from other groups
  -h, --help                     display this help message and exit
  -l, --login NEW_LOGIN          new value of the login name
  -L, --lock                     lock the user account
  -m, --move-home                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
  -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
  -Z, --selinux-user             new SELinux user mapping for the user account
  :_
```

Now this is a bit of a “if you know ,you know” situation. In this command using the -a flag will indeed accomplish what we want in the next step (which is to add them to the eastern_sales_division group) but it doesn’t remove them from their current group. In this case the -G flag will do what we need even though it says “new list of supplementary group.” Now, that doesn’t really sound like it will do what we want but I know that it does. Another way to think about it is to imagine the -a flag synonymous to the “Copy Paste” function, and the -G flag as a “Cut Paste” function.

Issuing the command “usermod -G eastern_sales_division aespinoza” will accomplish both Step #3: Remove aespinoza from west_sales_divison and Step#4: Add them to the eastern_sales_division group. Let’s do that now:

```
root@Wrk1: ~  
File Edit View Search Terminal Help  
root@Wrk1:~# groupmod -n western_sales_division sales  
root@Wrk1:~# groupadd eastern_sales_division  
root@Wrk1:~# usermod -G eastern_sales_division aespinoza  
root@Wrk1:~#
```

Awesome! Now for the last step we need to verify our changes took place. To do this we need to query the `/etc/group` file. To print it out at the terminal issue the command : `cat /etc/group`

```
root@Wrk1: ~  
File Edit View Search Terminal Help  
mailnull:x:492:  
smmsp:x:491:  
mysql:x:490:  
haldaemon:x:489:  
sshd:x:488:  
wadams:x:500:  
rcronn:x:501:  
vedwards:x:502:  
cflynn:x:503:  
mbrown:x:504:  
placy:x:505:  
bcassini:x:506:  
aespinoza:x:507:  
bkahn:x:508:  
schawla:x:509:  
mgmt1:x:510:wadams,rcronn,cflynn,mbrown,placy,schawla  
mgmt2:x:511:wadams,rcronn,vedwards,bkahn  
hr:x:512:wadams,vedwards,cflynn,mbrown,placy  
western_sales_division:x:null:bkahn,schawla,bcassini  
devel:x:514:mbrown  
it:x:515:rcronn,cflynn  
proj:x:516:placy  
eastern_sales_division:x:513:alespinoza  
root@Wrk1:~#
```

Robert Carpenter

github.com/robertmcarpenter

Sun November 17th 2024

We can see that all of our users from sales have transferred to western_sales_division with only aespinoza being the sole member of the eastern_sales_division successfully! (Feels awful like a gang war with East and West)

This now concludes this lab!

