Activity: Add and manage users with Linux commands

Activity overview

Previously, you focused on authorization, the concept of granting access to specific resources in a system. Another important concept in security is authentication. Authentication is the process of a user proving that they are who they say they are in the system.

When managing this, security analysts need to ensure

- not all users get access to the system,
- new users (those who are new to the organization or a group) are added to the system, and
- current users who change groups or leave the organization are deleted from the system.

In this lab activity, you'll use the useradd, usermod, userdel, and chown commands to manage user access in the Linux Bash shell.

Important: You must use sudo at the beginning of all the commands you use in this lab. Adding or removing users and groups are tasks that require root (super user) privileges, and you'll need to use sudo with the commands that are used to perform these tasks.

Scenario

In this scenario, a new employee with the username researcher9 joins an organization. You have to add them to the system and continue to manage their access during their time with the organization.

Here's how you'll do this task: **First**, you'll add a new employee to the system and then to their primary group. **Second**, you'll make this employee the owner of a file related to a particular project. **Third**, you'll add the new employee to a supplementary group. **Finally**, you'll delete the employee from the system.

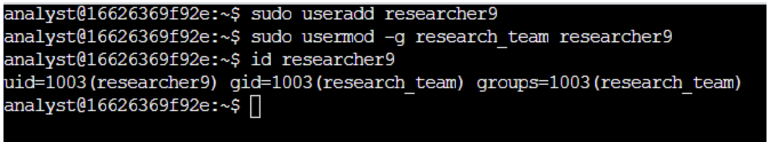
Task 1. Add a new user

A new employee has joined the Research department. In this task, you must add them to the system. The username assigned to them is researcher9.

1. Write a command to add a user called researcher9 to the system.

Next, you need to add the new user to the research_team group.

2. Use the usermod command and -g option to add researcher9 to the research_team group as their primary group.



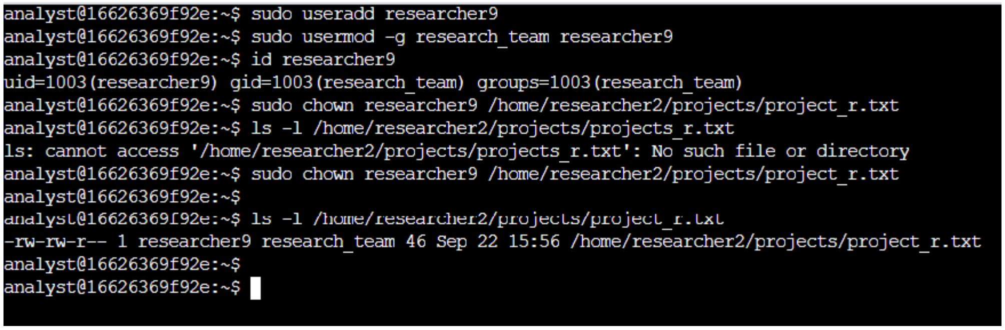
```
analyst@16626369f92e:~$ sudo useradd researcher9
analyst@16626369f92e:~$ sudo usermod -g research_team researcher9
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team)
analyst@16626369f92e:~$ 
```

Task 2. Assign file ownership

The new employee, researcher9, will take responsibility for project_r. In this task, you must make them the owner of the project_r.txt file.

The project_r.txt file is located in the /home/researcher2/projects directory, and owned by the researcher2 user.

- Use the chown command to make researcher9 the owner of /home/researcher2/projects/project_r.txt.



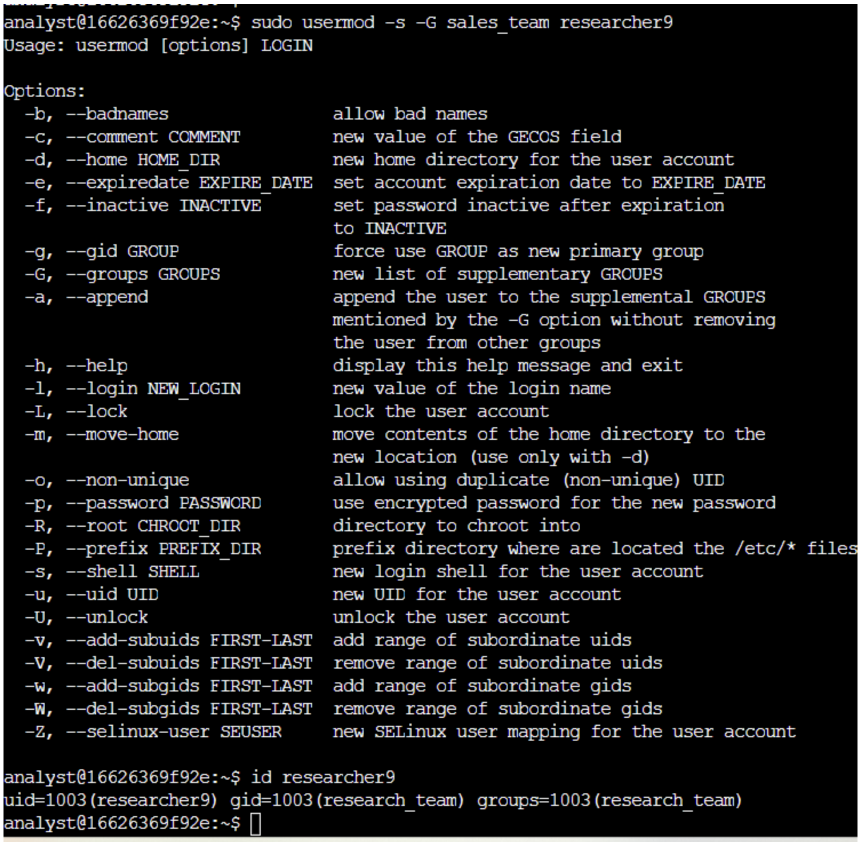
```
analyst@16626369f92e:~$ sudo useradd researcher9
analyst@16626369f92e:~$ sudo usermod -g research_team researcher9
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team)
analyst@16626369f92e:~$ sudo chown researcher9 /home/researcher2/projects/project_r.txt
analyst@16626369f92e:~$ ls -l /home/researcher2/projects/projects_r.txt
ls: cannot access '/home/researcher2/projects/projects_r.txt': No such file or directory
analyst@16626369f92e:~$ sudo chown researcher9 /home/researcher2/projects/project_r.txt
analyst@16626369f92e:~$
analyst@16626369f92e:~$ ls -l /home/researcher2/projects/project_r.txt
-rw-rw-r-- 1 researcher9 research_team 46 Sep 22 15:56 /home/researcher2/projects/project_r.txt
analyst@16626369f92e:~$
analyst@16626369f92e:~$ 
```

Task 3. Add the user to a secondary group

A couple of months later, this employee's role at the organization has changed, and they are working in both the Research and the Sales departments.

In this task, you must add researcher9 to a secondary group (sales_team). Their primary group is still research_team.

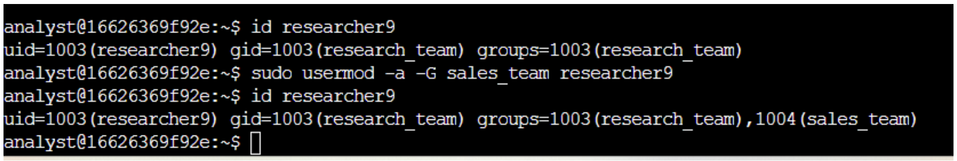
- Use the usermod command with the -a and -G options to add researcher9 to the sales_team group as a secondary group.



```
analyst@16626369f92e:~$ sudo usermod -s -G sales_team researcher9
Usage: usermod [options] LOGIN

Options:
  -b, --badnames                allow bad names
  -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
                                the user 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
  -R, --root CHROOT_DIR         directory to chroot into
  -P, --prefix PREFIX_DIR       prefix directory where are located the /etc/* files
  -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

analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team)
analyst@16626369f92e:~$
```

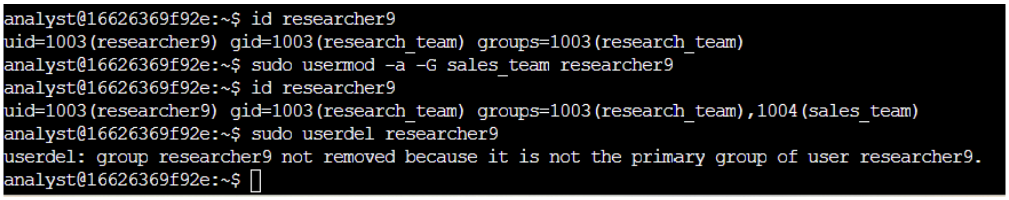


```
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team)
analyst@16626369f92e:~$ sudo usermod -a -G sales_team researcher9
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team),1004(sales_team)
analyst@16626369f92e:~$
```

Task 4. Delete a user

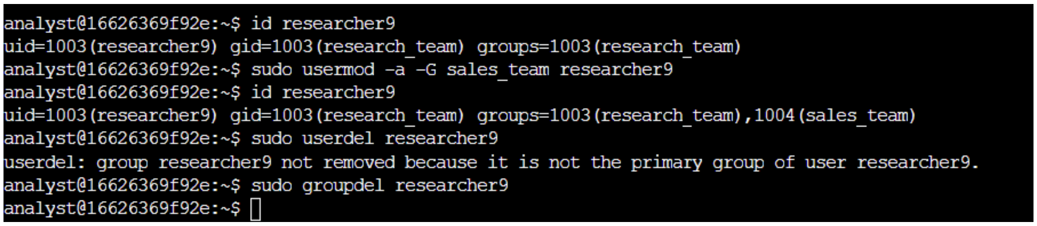
A year later, researcher9, decided to leave the company. In this task, you must remove them from the system.

1. Run a command to delete researcher9 from the system:



```
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team)
analyst@16626369f92e:~$ sudo usermod -a -G sales_team researcher9
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team),1004(sales_team)
analyst@16626369f92e:~$ sudo userdel researcher9
userdel: group researcher9 not removed because it is not the primary group of user researcher9.
analyst@16626369f92e:~$
```

2. Run the following command to delete the researcher9 group that is no longer required:



```
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team)
analyst@16626369f92e:~$ sudo usermod -a -G sales_team researcher9
analyst@16626369f92e:~$ id researcher9
uid=1003(researcher9) gid=1003(research_team) groups=1003(research_team),1004(sales_team)
analyst@16626369f92e:~$ sudo userdel researcher9
userdel: group researcher9 not removed because it is not the primary group of user researcher9.
analyst@16626369f92e:~$ sudo groupdel researcher9
analyst@16626369f92e:~$ 
```

Lab Summary: Add and Manage Users with Linux Commands

In this lab I practiced managing users and groups with Linux commands. I added a new user researcher9 and set their primary group to research_team. I changed ownership of the project_r.txt file so that researcher9 was responsible for it. Later, I updated their access by adding them to the sales_team group as a secondary group. Finally, when they left the organization, I removed researcher9 from the system and deleted their group.