Manage Users in Linux

Project description

In this scenario, a new employee (`researcher9`) just joined the organization. As a result, the research team tasked me to manage the user access within the organization. The operating system is Linux, indicating that the tasks require a command-line interface (Linux Bash shell) approach via Linux Terminal.

Describe the commands

The following are the commands when managing the user access:

1. `useradd` allows us to add users to the system.
   `sudo useradd researcher9`

2. `usermod` allows us to assign the users to a specific group as well as a secondary group.
   `sudo usermod -g research_team researcher9`

3. `userdel` allows us to delete the users from the system.
   `sudo userdel researcher9`

4. `groupdel` allows us to delete the user's group.
   `sudo groupdel researcher9` (When a new user is created, a group with the same name as the user is also created and that user is the only member of that group. After removing users, it is highly recommended to clean up any such empty groups that are left behind).

5. `chown` allows us to assign file ownership.
   `sudo chown researcher9 /home/researcher2/projects/project_r.txt`

Notice that all the commands require `sudo`. `Sudo` means "super user do!". On the other hand, `-g` here adds users "to" a specific group (primary).

Adding users and group

The organization would like to add researcher9 to the system and add him to the `research_team` group as his primary group. Here's the command:

`1. Sudo useradd researcher9`

```
2. Sudo usermod -g research_team researcher9
```
Alternatively, we can perform both steps at once:
```
1. Sudo useradd researcher9 -g research_team
```

```
analyst@be8fd768d3c9:~$ sudo useradd researcher9
analyst@be8fd768d3c9:~$ sudo usermod -g research_team researcher9
analyst@be8fd768d3c9:~$ []
```

Assign file ownership

`Researcher9` will take over `project_r` and the owner has to be him (`project_r.txt`). The `project_r.txt` is located in the `/home/researcher2/projects` and is currently owned by `researcher2`.
The command to make it happen is:

```
sudo chown researcher9 /home/researcher2/projects/project_r.txt
```

```
analyst@be8fd768d3c9:~$ sudo chown researcher9 /home/researcher2/projects/project_r.txt
analyst@be8fd768d3c9:~$ []
```

Add the user to a secondary group

A few months later, `researcher9` is now working in both the research and sales departments. My task is to add `resesarcher9` to the sales department as a secondary group without removing him from research department. Sales team is `sales_team`.

```
Sudo usermod -a -G sales_team researcher9
```
```
analyst@be8fd768d3c9:~$ sudo usermod -a -G sales_team researcher9
analyst@be8fd768d3c9:~$ []
```

Delete a user

A year later, researcher9 decided to leave the company. In this task, I have to remove him from the system. The command to make it happen is:
1. `sudo userdel researcher9` (Will not remove it. Please refer to "describe the command" section)
2. `sudo groupdel researcher9`

```
analyst@be8fd768d3c9:~$ sudo userdel researcher9
userdel: group researcher9 not removed because it is not the primary group of user researcher
9.
analyst@be8fd768d3c9:~$ sudo groupdel researcher9
analyst@be8fd768d3c9:~$
```

Summary

In this project, I demonstrated how to add the users, assign the users to the specific group and secondary group, assign the file ownership, and delete their account.