**User & Permissions in Ubuntu**

* **Note**: This project is provided by **Codecademy** (<https://www.codecademy.com>)
* **Note**: I did this project in the **Ubuntu** terminal.

**Project description**

I am the system administrator of a small start-up. I need to manage access levels on a Linux computer. The following three users need to work with their specific files on the Linux computer.

* **James** is a member of the Marketing team who needs full access to all the Marketing files.
* **Destiny** is a member of the Sales team who needs full access to all the Sales files.
* **Carolyn** is an administrator who needs read access to all files of all departments to monitor security and compliance.

**Start Project Timestamp**

A screenshot of a computer screen

Description automatically generated

**Environment Setup**

1. I first escalated my access level to the root using the **sudo -i** command and entering my password.

A screenshot of a computer

Description automatically generated

1. I created the groups for the marketing team, sales team, and the information technology team with the **groupadd**  command. Then I verified the creation of the group with the **cat /etc/group** command (results highlighted in yellow).

**A screen shot of a computer program

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

1. I then created and added the users to the group names with the **useradd** command. I also added **Carolyn** to the **sudo group** to have full environmental access with the **adduser** command.

* **Note**: The users are promptly added to their proper groups using the **-g** option. A new **home** directory is created for each new user since the **-m** flag is enabled.

A screenshot of a computer

Description automatically generated

1. The newly created **home** directories for the users can be observed using the **ls /home/** command.

A black background with white text

Description automatically generated

* 1. Verify that the users are added to their proper group by listing the groups for each user using the **groups [username]** command. Example shown below:

A black background with white text

Description automatically generated

1. To create separate directories for each department, I download Codecademy’s available archive of pre-created directories using the **wget** command (highlighted in yellow). Alternatively, the archive can be downloaded using the **curl** command. The archive website to download is:

<https://static-assets.codecademy.com/Courses/learn-linux/linux-shell-utilities/project-data.tar.gz>

A screen shot of a computer

Description automatically generated

1. I extracted the archive to create the data directory and the sub-directories for the department with the following command:

A screen shot of a computer

Description automatically generated

**Note**: This command extracts the tar ball and creates the directory structure in the **/temp** directory, which is a temporary directory in Linux where contents are automatically wiped out every 10 days.

1. The creation of files and directories can then be verified by visualizing the directory structure using the following command that relies on **sed**, a stream editor for filtering and transforming text:
2. find /tmp/data | sed -e "s/[^-][^\/]\*\// |/g" -e "s/|\([^ ]\)/|-\1/"

or using the **tree utility**. Which can be installed if not done so:

tree -d /tmp/data

The file structure looks like the following:

A screen shot of a computer

Description automatically generated

A computer screen with text

Description automatically generated

1. I modified the file permissions using the **chown** command to reflect the requirements for each user and group. For example, I reflected **marketing-team** as the owner of the **Marketing** group (**/tmp/data/Marketing**). The same was done for the **sales-team** and **it-team**. Which is shown below

A black background with white text

Description automatically generated

1. I then adjusted the permission levels for each directory with the **chmod** command. Read, write, and execute for the **user** and **group.** Read only for **others**. Which is shown below.

A group of numbers on a black background

Description automatically generated

1. The owner group for each directory can be verified with the **ls** command if you list all directories in **/tmp/data**. Which is highlighted in yellow.

A screenshot of a computer

Description automatically generated

**Access Verification**

1. I then impersonated a user to verify their level of access. I started with **James**.

A black background with white text

Description automatically generated

1. I had access to the Marketing directory as **James’**.

A black background with white text

Description automatically generated

1. The data of one of the file can be read with the **cat** command. Example shown below:

A screenshot of a computer program

Description automatically generated

1. When trying to read into the Sales directory, which **James** shouldn’t have access to, this is the result:



1. I logged into **Destiny**, and this is the result:

A black background with white text

Description automatically generated

**End Project Timestamp**

A screen shot of a computer

Description automatically generated