## Files in Linux

## Project description

In this scenario, the analyst team tasked me to execute some organizational tasks. The following are as follows:

- 1. Find and search files
- 2. Create and remove a directory
- 3. Move and remove a file
- 4. Create and edit a file

The operating system is Linux, indicating that the tasks require a command-line interface (Linux Bash shell) approach via Linux Terminal.

```
analyst@ec0337e39025:~$ 1s

logs project reports temp

analyst@ec0337e39025:~$ 1s -la

total 44

drwxr-xr-x 6 analyst analyst 4096 Dec 20 06:40 .

drwxr-xr-x 1 root root 4096 Dec 20 05:24 .

-rw------ 1 analyst analyst 9 Dec 20 06:40 .bash_history

-rw-r--r-- 1 analyst analyst 220 Apr 18 2019 .bash_logout

-rw-r--r-- 1 analyst analyst 3597 Dec 20 05:24 .bashrc

-rw-r--r-- 1 analyst analyst 3597 Dec 20 05:24 .profile

drwxr-xr-x 2 analyst root 4096 Dec 20 05:24 logs

drwxr-xr-x 2 analyst root 4096 Dec 20 05:24 project

drwxr-xr-x 3 analyst root 4096 Dec 20 05:24 reports

drwxr-xr-x 2 analyst root 4096 Dec 20 05:24 temp

analyst@ec0337e39025:~$ []
```

#### Find and search files

The analyst team asked me to locate a log file whose lines contain the text string error. The file is server\_logs.txt.within the directory logs. We can examine the actual lines within the file by using cat server\_logs.txt after cd logs / cat

/home/analyst/logs/server\_logs.txt command. The image below shows the entire lines of the text.

```
analyst@ec0337e39025:~/logs$ cat server_logs.txt
2022-09-28 13:55:55 info User logged on successfully
2022-09-28 13:56:22 error The password is incorrect
2022-09-28 13:56:48 warning The file storage is 75% full
2022-09-28 15:55:55 info User logged on successfully
2022-09-28 15:56:22 error The username is incorrect
2022-09-28 15:56:48 warning The file storage is 90% full
2022-09-28 16:55:55 info User navigated to settings page
2022-09-28 16:56:22 error The password is incorrect
2022-09-28 16:56:48 warning The current user's password expires in 15 days
2022-09-29 13:55:55 info User logged on successfully
2022-09-29 13:56:22 error An unexpected error occurred
2022-09-29 13:56:48 warning The file storage is 90% full
2022-09-29 15:55:55 info User navigated to settings page 2022-09-29 15:56:22 error Unauthorized access
022-09-29 15:56:48 warning The file storage is 75% full
2022-09-29 16:55:55 info User requested security reports
2022-09-29 16:56:22 error Unauthorized access
2022-09-29 16:56:48 warning The current user's password expires in 15 daysanalyst@ec0337e3902
```

Now, let's filter this file so it will return a list of the lines that match the text string "error" in that file. The command grep error server\_logs.txt will make it happen. As the result shows below, there are six lines that match the text string "error".

The analyst team also would like me to locate files whose names contain Q1 and access within the users directory. First, let's write the command cd /home/analyst/reports/users to enter users directory and ls /ls -la commands to view all the files within the directory.

This command (grep) will allow us to find the files whose names contain Q1:

ls | grep Q1 (make sure you're already in the directory) or ls

/home/analyst/reports/users | grep Q1. There are three files associated with Q1.

```
analyst@ec0337e39025:~/reports/users$ ls | grep Q1
Q1_access.txt
Q1_added_users.txt
Q1 deleted users.txt
```

Using the same logic, we can locate a file whose name contains access:

1s | grep access (make sure you're already in the directory) or 1s

/home/analyst/reports/users | grep access

As the result shows, there are four files associated with access.

```
analyst@ec0337e39025:~/reports/users$ ls | grep access
Q1_access.txt
Q2_access.txt
Q3_access.txt
Q4_access.txt
analyst@ec0337e39025:~/reports/users$
```

Last but not least, the analyst team would like to search information contained in user files and report on users that were added and deleted from the system. They would like to search the  $Q2\_deleted\_users.txt$  file within the users directory for the username jhill. As the result shows, we found jhill in this file.

```
grep jhill Q2 deleted users.txt
```

```
analyst@ec0337e39025:~/reports/users$ grep jhill Q2_deleted_users.txt
1025 jhill Sales
```

To see whose people have been added to the Human Resources department, we can use grep command. For more than one word, we should use "" to execute the command.

```
grep "Human Resources" Q4 added users.txt
```

```
analyst@ec0337e39025:~/reports/users$ grep "Human Resources" Q4_added_users.txt
1151 sshah Human Resources
1145 msosa Human Resources
analyst@ec0337e39025:~/reports/users$ [
```

# Create and remove a directory

The analyst team would like me to create a new directory named logs and remove temp directory from the system. Here's the command line to create and remove directory:

mkdir logs (create)

rmdir temp (remove)

```
analyst@95cc38ed66c7:~$ mkdir logs
analyst@95cc38ed66c7:~$ ls
logs notes reports temp
analyst@95cc38ed66c7:~$
```

```
analyst@95cc38ed66c7:~$ rmdir temp
analyst@95cc38ed66c7:~$ ls
logs notes reports
analyst@95cc38ed66c7:~$
```

#### Move and remove a file

The team would like to move Q3patches.txt file to another directory. In the Notes directory (please navigate here), the command mv Q3patches.txt /home/analyst/report will move the file to the notes directory.

```
analyst@95cc38ed66c7:~$ cd /home/analyst/notes
analyst@95cc38ed66c7:~/notes$ cd /home/analyst/notes
analyst@95cc38ed66c7:~/notes$ cd notes
-bash: cd: notes: No such file or directory
analyst@95cc38ed66c7:~/notes$ mv Q3patches.txt /home/analyst/reports/
analyst@95cc38ed66c7:~/notes$ ls /home/analyst/reports
Q1patches.txt Q2patches.txt Q3patches.txt
analyst@95cc38ed66c7:~/notes$ [
```

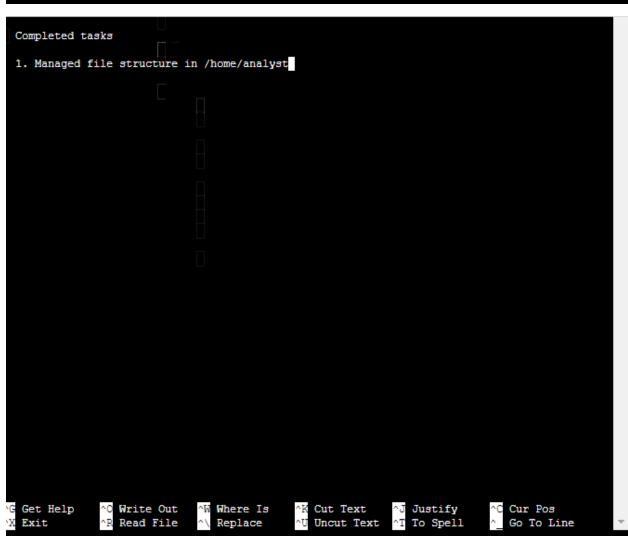
The team would like me to remove tempnotes as it is no longer required in the notes directory. The command rm tempnotes.txt will allow us to delete it (please make sure to use cd command to navigate to notes directory).

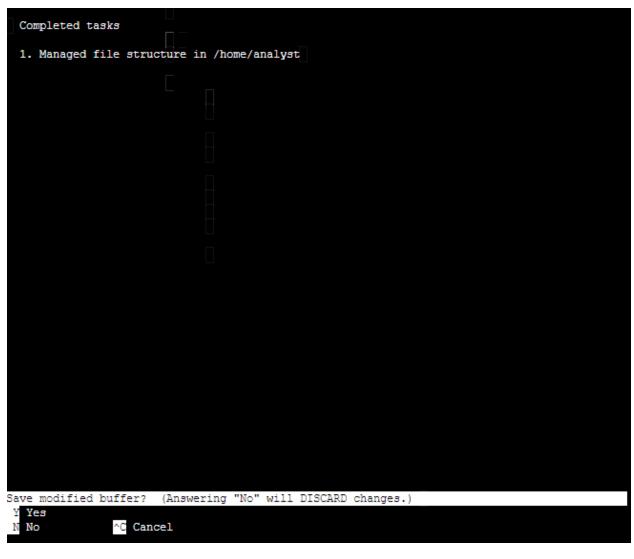
```
analyst@95cc38ed66c7:~/notes$ rm tempnotes.txt
analyst@95cc38ed66c7:~/notes$ ls
analyst@95cc38ed66c7:~/notes$ [
```

#### Create and edit a file

The analyst team would like me to create a new file and edit it. The command touch tasks.txt allows us to create a file. On the other hand, the command nano tasks.txt allows us to edit a file. Make sure to press CTRL + X to exit from the nano editor. When being asked "Save modified bufferer", press Y to save the new data to the file. Then, please press enter to confirm that the file name to write is tasks.txt. Use cat tasks.txt command to display the contents of the tasks.txt.

```
analyst@95cc38ed66c7:~/notes$ touch tasks.txt
analyst@95cc38ed66c7:~/notes$ ls
tasks.txt
analyst@95cc38ed66c7:~/notes$ [
```





```
analyst@95cc38ed66c7:~/notes$ cat tasks.txt
Completed tasks

1. Managed file structure in /home/analyst
analyst@95cc38ed66c7:~/notes$
```

# Summary

Here, I successfully found and searched files, created and removed a directory, moved and removed a file, and created and edited a file on Linux Terminal.