

# Fichiers sous Linux

## Description du projet

Dans ce scénario, l'équipe d'analystes m'a chargé d'exécuter quelques tâches. Les voici :

1. Trouver et rechercher les fichiers
2. Créer et supprimer un répertoire
3. Déplacez et retirez un fichier
4. Créer et modifier un fichier

Le système d'exploitation est Linux, ce qui indique que les tâches nécessitent une interface en ligne de commande (shell Bash Linux) via le terminal Linux.

```
analyst@ec0337e39025:~$ ls
logs project reports temp
analyst@ec0337e39025:~$ ls -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:~$
```

## Trouver et rechercher les fichiers

L'équipe d'analystes m'a demandé de localiser un journal dont les lignes contiennent la chaîne de texte

`error`. Le fichier `server_logs.txt` se trouve dans le répertoire `logs`. Nous pouvons examiner

les journaux système lignes en utilisant `cat server_logs.txt` après `cd logs / cat`

``/home/ analyst /logs/server_logs.txt`` est affichée. L'image ci-dessous illustre l'intégralité du processus. Lignes du texte.

```
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
2022-09-29 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
5:~/logs$
```

Maintenant, allons-y, filtrons-le, donc il renverra une liste des lignes qui correspondent à la chaîne de caractères « erreur » dans ce fichier. La commande `grep error server_logs.txt` fait se produire. Comme le montre le résultat ci-dessous, il y a six lignes qui correspondent à la chaîne de texte « error ».

```
analyst@ec0337e39025:~/logs$ grep error server_logs.txt
2022-09-28 13:56:22 error   The password is incorrect
2022-09-28 15:56:22 error   The username is incorrect
2022-09-28 16:56:22 error   The password is incorrect
2022-09-29 13:56:22 error   An unexpected error occurred
2022-09-29 15:56:22 error   Unauthorized access
2022-09-29 16:56:22 error   Unauthorized access
```

L'équipe d'analystes aimerait également que je localise les noms contenant `Q1` et `access` parmi les `users` du répertoire. Tout d'abord, commençons par saisir la commande `cd /home/analyst/reports/users` pour entrer dans les `users`. Les commandes `directory` et `ls/ls -la` permettent d'afficher tous les fichiers contenus dans le répertoire.

```
analyst@ec0337e39025:~/logs$ cd /home/analyst/reports/users
analyst@ec0337e39025:~/reports/users$ ls
Q1_access.txt      Q2_access.txt      Q3_access.txt      Q4_access.txt
Q1_added_users.txt Q2_added_users.txt Q3_added_users.txt Q4_added_users.txt
Q1_deleted_users.txt Q2_deleted_users.txt Q3_deleted_users.txt Q4_deleted_users.txt
analyst@ec0337e39025:~/reports/users$ ls -la
total 56
drwxr-xr-x 2 analyst root 4096 Dec 20 05:24 .
drwxr-xr-x 3 analyst root 4096 Dec 20 05:24 ..
-rw-r--r-- 1 analyst root 85 Dec 20 05:24 Q1_access.txt
-rw-r--r-- 1 analyst root 251 Dec 20 05:24 Q1_added_users.txt
-rw-r--r-- 1 analyst root 219 Dec 20 05:24 Q1_deleted_users.txt
-rw-r--r-- 1 analyst root 86 Dec 20 05:24 Q2_access.txt
-rw-r--r-- 1 analyst root 251 Dec 20 05:24 Q2_added_users.txt
-rw-r--r-- 1 analyst root 220 Dec 20 05:24 Q2_deleted_users.txt
-rw-r--r-- 1 analyst root 85 Dec 20 05:24 Q3_access.txt
-rw-r--r-- 1 analyst root 251 Dec 20 05:24 Q3_added_users.txt
-rw-r--r-- 1 analyst root 220 Dec 20 05:24 Q3_deleted_users.txt
-rw-r--r-- 1 analyst root 86 Dec 20 05:24 Q4_access.txt
-rw-r--r-- 1 analyst root 251 Dec 20 05:24 Q4_added_users.txt
-rw-r--r-- 1 analyst root 220 Dec 20 05:24 Q4_deleted_users.txt
```

Cette commande (`grep`) va, permettez-nous de trouver les dont les noms contiennent `Q1` :

`ls | grep Q1` (assurez-vous d'être déjà dans le répertoire) ou `ls`

`/home/analyst/reports/users | grep Q1`. Il y a trois fichiers associés avec `Q1`.

```
analyst@ec0337e39025:~/reports/users$ ls | grep Q1
Q1_access.txt
Q1_added_users.txt
Q1_deleted_users.txt
```

En utilisant le même logiquement, nous pouvons localiser un dont le nom contient

`accès` : `ls | grep access` (assurez-vous que vous êtes déjà dans le répertoire) ou `ls`

`/home/analyst/rapports/users | grep access` Comme le montrent les résultats, il existe quatre groupes associés avec `access`.

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

Enfin, l'équipe d'analystes souhaite rechercher des informations contenues dans les fichiers utilisateurs et établir un rapport sur les utilisateurs. qui étaient ajoutés et supprimés. Ils souhaitent effectuer une recherche dans le fichier `Q2_deleted_users.txt` au sein du système. Répertoire pour l'utilisateur `jhill`. Comme le montre le résultat, nous trouvons `jhill` dans ce `grep` : `grep jhill`

`Q2_deleted_users.txt`

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

Pour voir quels employés ont été ajoutés au service des `Human Resources` département, nous pouvons utiliser la commande. Pour plus d'un mot, il convient d'utiliser « » pour exécuter la commande.

`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$
```

## Créer et supprimer un répertoire

L'équipe d'analystes souhaiterait que je crée un nouveau répertoire nommé `logs` et suppression, Créer un répertoire `temp` système. Voici la ligne de commande pour créer et supprimer un répertoire :

`mkdir logs` (créer)

`rmdir temp` (supprimer)

```
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:~$
```

## Déplacez et retirez un fichier

L'équipe souhaite déplacer le fichier `Q3patches.txt` vers un autre répertoire, dans le répertoire `Notes`.

(s'il te plaît naviguer ici), la commande `mv Q3patches.txt /home/analyst/report` déplacera le fichier vers le répertoire `notes`.

```
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$
```

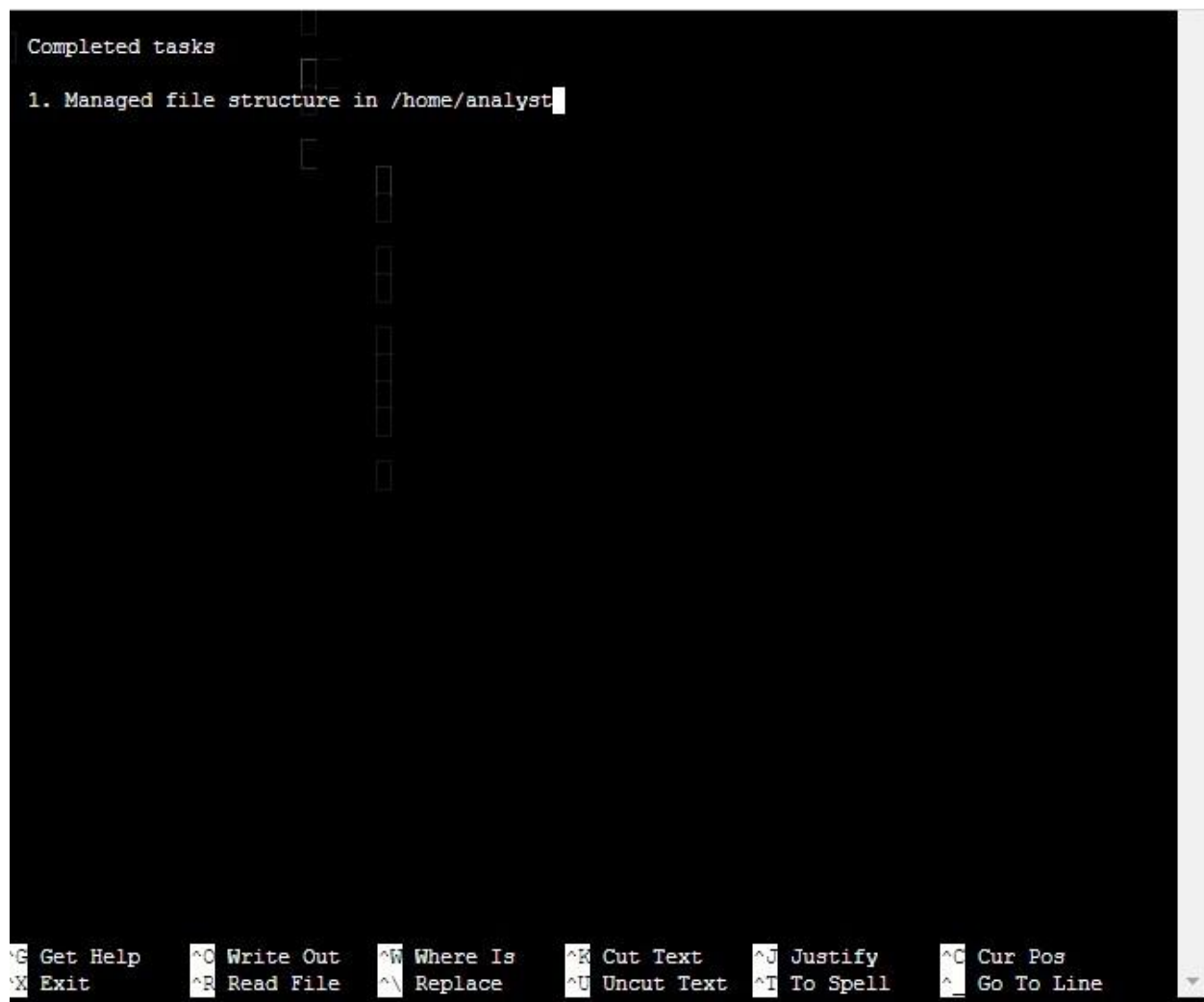
L'équipe souhaiterait que je retire les `temp notes` comme le fichier `tempnotes.txt` n'est plus nécessaire dans le répertoire `notes`. La commande `rm temp notes.txt` le supprimera. Nous permettre de supprimer, s'il vous plaît. Assurez-vous d'utiliser la commande `cd` pour accéder au répertoire des notes.

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

## Créer et modifier un fichier

L'équipe d'analystes souhaiterait que je crée un nouveau fichier et que je le modifie. La commande `touch tasks.txt` permet de créer un fichier. En revanche, la commande `nano tasks.txt` permet de modifier un fichier. N'oubliez pas d'appuyer sur CTRL + X pour quitter l'éditeur nano. A demandé « Sauvez Modified buffer », appuyez sur Y pour enregistrer les nouvelles données dans le. Ensuite, veuillez appuyer sur Entrée pour confirmer. Que le nom à écrire. Le fichier tasks.txt contient le contenu de tasks.txt. Utilisez la commande `cat tasks.txt` pour afficher ce contenu.

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



The screenshot shows the nano text editor interface. At the top, it says "Completed tasks". Below that, there is a list of tasks, with the first one being "1. Managed file structure in /home/analyst". The editor has a dark background with light-colored text. At the bottom, there is a status bar with various keyboard shortcuts and their corresponding actions, such as ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, ^C Cur Pos, ^X Exit, ^R Read File, ^\ Replace, ^U Uncut Text, ^T To Spell, and ^\_ Go To Line.



```
Completed tasks
1. Managed file structure in /home/analyst

Save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No      ^C Cancel
```

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

## Résumé

Ici, j'ai réussi : j'ai trouvé et recherché des fichiers, créé et supprimé un répertoire, déplacé et supprimé un fichier, et créé et modifié un fichier sur le terminal Linux.