TP13 – Les utilisateurs et les droits

SAOU Rayan

Table des matières

---

1. La gestion des utilisateurs

Les UID, GID et leurs groupes sont pour les utilisateurs :

- Daemon : UID = 1, GID = 1 et son groupe est daemon
- Luke : il n'existe pas donc il n'a pas de UID, GID ou de groupe

```
root@DEB13Server: ~#id daemon
uid=1(daemon) gid=1(daemon) groupes=1(daemon)
root@DEB13Server: ~#id luke
id: ‘luke’ : utilisateur inexistant
root@DEB13Server: ~#
```

Nous allons donc créer le groupe jedi et rebelles avec la commande ***groupadd***

```
root@DEB13Server: ~#groupadd jedi
root@DEB13Server: ~#groupadd rebelles
root@DEB13Server: ~#
```

Ensuite, nous créons les comptes utilisateurs : **luke, vador** et **solo** avec la commande ***useradd***

```
root@DEB13Server: ~#useradd -g jedi -G rebelles -m luke
root@DEB13Server: ~#useradd -g jedi -m vador
root@DEB13Server: ~#_
```

-g permet d'assigner un groupe, -G un groupe secondaire et -m de créer un répertoire

Nous affichons les dernières lignes de /etc/passwd et /etc/group avec la commande ***tail***

```
root@DEB13Server: ~#tail -3 /etc/passwd
luke:x:1002:1002::/home/luke:/bin/sh
vador:x:1003:1002::/home/vador:/bin/sh
solo:x:1004:1003::/home/solo:/bin/sh
root@DEB13Server: ~#tail -2 /etc/group
jedi:x:1002:
rebelles:x:1003:luke
root@DEB13Server: ~#
```

Nous définissons le mot de passe de l'utilisateur **luke** avec la commande ***passwd*** avec comme mot de passe « ***password*** » :

```
root@DEB13Server: ~#passwd luke
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
root@DEB13Server: ~#
```

Nous ouvrons une seconde console et nous nous connectons au compte de luke :

```
Debian GNU/Linux 13 DEB13Server tty2

DEB13Server login: luke
Password:
Linux DEB13Server 6.12.48+deb13-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.12.48-1 (2025-09-20) x86_

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
$ who
luke     seat0        2025-12-17 10:13
luke     tty2         2025-12-17 10:13
root     seat0        2025-12-17 09:39
root     tty1         2025-12-17 09:39
$
```

Nous nous déconnectons avec un ***logout*** et nous revenons à notre session root, nous allons définir son shell sh par bash

```
root@DEB13Server: ~#usermod -s /bin/bash luke
root@DEB13Server: ~#_
```

Nous nous reconnectons et nous observons le prompt de la commande ***id*** :

```
luke@DEB13Server:~$ id
uid=1002(luke) gid=1002(jedi) groupes=1002(jedi),1003(rebelles)
luke@DEB13Server:~$
```

Nous revenons dans notre première console et nous créons l'utilisateur leia :

```
root@DEB13Server: ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia)
root@DEB13Server: ~#_
```

Le répertoire de l'utilisateur leia n'a pas été créer car la commande useradd n'a pas été spécifié avec l'option -m donc pas de repertoire crée

```
root@DEB13Server: ~#ls -l /home
total 20
drwx------ 5 guest guest    4096 15 déc.  16:04 guest
drwx------ 2 luke  jedi     4096 17 déc.  10:07 luke
drwx------ 2 sio   sio      4096 18 oct.  22:47 sio
drwx------ 2 solo  rebelles 4096 17 déc.  10:10 solo
drwx------ 2 vador jedi     4096 17 déc.  10:07 vador
root@DEB13Server: ~#
```

Nous allons affecter *Leia* au groupe rebelles ***avec usermod -G***

```
root@DEB13Server: ~#usermod -G rebelles leia
root@DEB13Server: ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1003(rebelles)
root@DEB13Server: ~#
```

Ensuite Leia quitte le groupe rebelles pour aller au groupe *jedi* :

```
root@DEB13Server: ~#usermod -G jedi leia
root@DEB13Server: ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1002(jedi)
root@DEB13Server: ~#_
```

Ensuite ***Leia*** n'a plus aucun groupe secondaire :

```
root@DEB13Server: ~#usermod -G "" leia
root@DEB13Server: ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia)
```

Nous allons supprimer le compte ***Leia*** et le recréer mais cette fois ci nous allons créer son répertoire personnel et créer un répertoire et un fichier à partir de son compte

```
root@DEB13Server: ~#userdel leia
root@DEB13Server: ~#useradd -m leia
root@DEB13Server: ~#cd /home/leia
root@DEB13Server: /home/leia#su - leia
$ mkdir rep1
$ cd rep1
$ touch fichier1
$ ls -l
total 0
-rw-rw-r-- 1 leia leia 0 17 déc.  10:46 fichier1
$ exit
root@DEB13Server: /home/leia#cd
root@DEB13Server: ~#_
```

Nous supprimons encore une fois le compte utilisateur ***Leia*** avec ses fichiers dans son répertoire de connexion :

```
root@DEB13Server: ~#userdel -r leia
userdel : leia spool de courrier /var/mail/leia non trouvé
root@DEB13Server: ~#ls -l /home/leia
ls: impossible d'accéder à '/home/leia': Aucun fichier ou dossier de ce nom
root@DEB13Server: ~#id leia
id: ‘leia’ : utilisateur inexistant
root@DEB13Server: ~#
```

Nous recréons le compte Leia avec le même UID et GID :

```
root@DEB13Server: ~#groupadd -g 1007 leia
root@DEB13Server: ~#useradd -u 1007 -g leia -m -s /bin/bash leia
root@DEB13Server: ~#id leia
uid=1007(leia) gid=1007(leia) groupes=1007(leia)
root@DEB13Server: ~#passwd leia
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
root@DEB13Server: ~#_
```

Et nous créons aussi un compte *toor* avec:

```
root@DEB13Server: ~#useradd -u 0 -o -d /root -s /bin/bash toor
useradd attention: l'uid de toor, 0, est en dehors de la plage UID_MIN 1000 et UID_MAX 60000 .
root@DEB13Server: ~#id toor
uid=0(root) gid=1008(toor) groupes=0(root)
root@DEB13Server: ~#_
```

Nous ouvrons un second terminal et nous nous connectons à *toor*

Nous créons cette fois ci un utilisateur avec la commande adduser :

```
root@DEB13Server: ~#adduser palpatine
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
Modifier les informations associées à un utilisateur pour palpatine
Entrer la nouvelle valeur, ou appuyer sur ENTER pour la valeur par défaut
        NOM []:
        Numéro de chambre []:
        Téléphone professionnel []:
        Téléphone personnel []:
        Autre []:
Is the information correct? [Y/n]
root@DEB13Server: ~#id palpatine
uid=1005(palpatine) gid=1005(palpatine) groupes=1005(palpatine),100(users)
```

Nous recherchons avec la commande *grep* l'utilisateur *Luke* :

```
root@DEB13Server: ~#grep luke /etc/passwd
luke:x:1002:1002::/home/luke:/bin/bash
root@DEB13Server: ~#
```

2. La gestion des droits

Nous créons un répertoire *etoilenoir(e*)* :

```
root@DEB13Server: ~#mkdir /home/etoilenoir
root@DEB13Server: ~#cd /home/etoilenoir/
root@DEB13Server: /home/etoilenoir#echo "voici les plans" > plans
root@DEB13Server: /home/etoilenoir#echo "c'est ouvert" > entree_secrete
root@DEB13Server: /home/etoilenoir#cd
```

Ensuite nous modifions le propriétaire du répertoire ainsi que les droits

```
root@DEB13Server: ~#ls -ld /home/etoilenoir/
drwxr-xr-x 2 root toor 4096 18 déc.  15:18 /home/etoilenoir/
root@DEB13Server: ~#chown luke /home/etoilenoir/
root@DEB13Server: ~#chgrp jedi /home/
etoilenoir/ guest/      leia/       luke/       palpatine/  sio/
root@DEB13Server: ~#chgrp jedi /home/etoilenoir/
root@DEB13Server: ~#chmod 750 /home/etoilenoir/
root@DEB13Server: ~#ls -ld /home/etoilenoir/
drwxr-x--- 2 luke jedi 4096 18 déc.  15:18 /home/etoilenoir/
root@DEB13Server: ~#
```

Ensuite, nous utilisons la notation symbolique pour changer les droits de certains fichiers

```
root@DEB13Server: ~#chmod g=r,o=- /home/etoilenoir/*
root@DEB13Server: ~#chgrp jedi /home/etoilenoire/plans
chgrp: impossible d'accéder à '/home/etoilenoire/plans': Aucun fich
root@DEB13Server: ~#chgrp jedi /home/etoilenoir/
entree_secrete  plans
root@DEB13Server: ~#chgrp jedi /home/etoilenoir/plans
root@DEB13Server: ~#chgrp rebelles /home/etoilenoir/entree_secrete
root@DEB13Server: ~#ls -l /home/etoilenoir/
total 8
-rw-r----- 1 root rebelles 13 18 déc.  15:18 entree_secrete
-rw-r----- 1 root jedi     16 18 déc.  15:18 plans
root@DEB13Server: ~#_
```

Puis nous testons les commandes via le compte *luke (via su)*

```
luke@DEB13Server:~$ ls /home/etoilenoir/
entree_secrete  plans
luke@DEB13Server:~$ cat /home/etoilenoir/plans
voici les plans
luke@DEB13Server:~$ cat /home/etoilenoir/entree_secrete
c'est ouvert
luke@DEB13Server:~$ man who > /home/etoilenoir/fichier
luke@DEB13Server:~$ ls
luke@DEB13Server:~$ ls /home/etoilenoir/
entree_secrete  fichier  plans
luke@DEB13Server:~$ rm /home/etoilenoir/fichier
luke@DEB13Server:~$ ls
luke@DEB13Server:~$ ls /home/etoilenoir/
entree_secrete  plans
luke@DEB13Server:~$ echo "=====" >> /home/etoilenoir/plans
-bash: /home/etoilenoir/plans: Permission non accordée
luke@DEB13Server:~$ _
```

Puis nous testons le compte *solo* :

```
root@DEB13Server: ~#su - solo
$ ls /home/etoilenoir
ls: impossible d'ouvrir le répertoire '/home/etoilenoir': Permission non accordée
$ ncal > /home/etoilenoir/fichier
-sh: 2: cannot create /home/etoilenoir/fichier: Permission denied
$ rm -f /home/etoilenoir/entree_secrete
rm: impossible de supprimer '/home/etoilenoir/entree_secrete': Permission non accordée
$ cat /home/etoilenoir_secrete
cat: /home/etoilenoir_secrete: Aucun fichier ou dossier de ce nom
$ cat /home/etoilenoir/entree_secrete
cat: /home/etoilenoir/entree_secrete: Permission non accordée
$
```

Puis nous modifions l'accès à la commande *uptime* :

```
root@DEB13Server: ~#whereis uptime
uptime: /usr/bin/uptime /usr/share/man/man1/uptime.1.gz
root@DEB13Server: ~#whatisuptime
-bash: whatisuptime : commande introuvable
root@DEB13Server: ~#whatis uptime
uptime (1)           - Indiquer depuis quand le système a été mis en route
root@DEB13Server: ~#ls -l /usr/bin/uptime
-rwxr-xr-- 1 root root 14648 30 juil. 13:58 /usr/bin/uptime
root@DEB13Server: ~#chmod o-x /usr/bin/uptime
root@DEB13Server: ~#ls -l /usr/bin/uptime
-rwxr-xr-- 1 root root 14648 30 juil. 13:58 /usr/bin/uptime
root@DEB13Server: ~#su - luke
luke@DEB13Server:~$ uptime
-bash: /usr/bin/uptime: Permission non accordée
luke@DEB13Server:~$ _
```

Luke n'a pas accès à la commande uptime, puis nous nous reconnectons à *root* pour lui accorder l'accès :

```
root@DEB13Server: ~#chmod o+x /usr/bin/uptime
root@DEB13Server: ~#ls -l /usr/bin/uptime
-rwxr-xr-x 1 root root 14648 30 juil. 13:58 /usr/bin/uptime
root@DEB13Server: ~#su - luke
luke@DEB13Server:~$ uptime
 15:52:21 up 52 min,  2 users,  load average: 0,00, 0,01, 0,00
luke@DEB13Server:~$
```

3. La gestion des droits, compléments

Nous allons créer des fichiers dans le répertoire *etoilenoir* et mesurer l'impact de la modification des droits

```
root@DEB13Server: ~#chmod 3770 /home/etoilenoir/
root@DEB13Server: ~#ls -ld /home/etoilenoir/
drwxrws--T 2 luke jedi 4096 18 déc.  15:34 /home/etoilenoir/
root@DEB13Server: ~#echo "fichier un" > /home/etoilenoir/f1
root@DEB13Server: ~#su - luke

luke@DEB13Server:~$ rm /home/etoilenoir/f3
luke@DEB13Server:~$ echo "bonjour" > /home/etoilenoir/f2
luke@DEB13Server:~$ _
```

```
root@DEB13Server: ~#su - vador
$ echo "bonjour" > /home/etoilenoir/f3
$ exit
```

```
root@DEB13Server: ~#ls -l /home/etoilenoir/f?
-rw-r--r-- 1 root  jedi 11 18 déc.  15:57 /home/etoilenoir/f1
-rw-r--r-- 1 luke  jedi  8 18 déc.  15:58 /home/etoilenoir/f2
-rw-r--r-- 1 vador jedi  8 18 déc.  16:00 /home/etoilenoir/f3
root@DEB13Server: ~#
```

Nous essayons avec le compte *vador* de supprimer un fichier :

```
root@DEB13Server: ~#su - vador
$ rm /home/etoilenoir/f2
rm : supprimer '/home/etoilenoir/f2' qui est protégé en écriture et est du type « regular file » ? y
rm: impossible de supprimer '/home/etoilenoir/f2': Opération non permise
$ exit
root@DEB13Server: ~#
```

Vador n'a pas pu supprimer le fichier à cause du sticky bit, nous allons supprimer ce droit et réessayer :

```
root@DEB13Server: ~#chmod -t /home/etoilenoir/
root@DEB13Server: ~#ls -ld /home/etoilenoir/
drwxrws--- 2 luke jedi 4096 18 déc.  16:00 /home/etoilenoir/
root@DEB13Server: ~#su - vador
$ rm /home/etoilenoir/f2
rm : supprimer '/home/etoilenoir/f2' qui est protégé en écriture et est du type « regular file » ? y
$ ls -l /home/etoilenoir/f2
ls: impossible d'accéder à '/home/etoilenoir/f2': Aucun fichier ou dossier de ce nom
$ exit
root@DEB13Server: ~#
```

```
root@DEB13Server: ~#ls -l /dev/sda1
brw-rw---- 1 root disk 8, 1 18 déc.  15:00 /dev/sda1
root@DEB13Server: ~#
```

Ici, seul *root* ou le groupe*disk* peuvent formater cette partition.

Si nous essayons de copier les fichiers autre part, ils gardent leur propriété grâce à l'option *-p*

```
root@DEB13Server: ~#cp -p /home/etoilenoir/* /tmp
root@DEB13Server: ~#ls -l /tmp/plans /tmp/entree_secrete
-rw-r----- 1 root rebelles 13 18 déc.  15:18 /tmp/entree_secrete
-rw-r----- 1 root jedi     16 18 déc.  15:18 /tmp/plans
root@DEB13Server: ~#
```

Nous modifions la propriété du fichier pour *Luke* :

```
root@DEB13Server: ~#chown luke /tmp/entree_secrete
root@DEB13Server: ~#ls -l /tmp/entree_secrete
-rw-r----- 1 luke rebelles 13 18 déc.  15:18 /tmp/entree_secrete
root@DEB13Server: ~#_
```

Et si nous essayons :

```
luke@DEB13Server:~$ cat /tmp/entree_secrete
c'est ouvert
luke@DEB13Server:~$ echo "======" >> /tmp/entree_secrete
luke@DEB13Server:~$ cat /tmp/entree_secrete
c'est ouvert
======
luke@DEB13Server:~$ /tmp/entree_secrete
-bash: /tmp/entree_secrete: Permission non accordée
luke@DEB13Server:~$ exit_
```

Cependant si nous essayons avec le compte *solo*:

```
root@DEB13Server: ~#su - solo
$ cat /tmp/entree_secrete
c'est ouvert
======
$ echo "++++++" >> /tmp/entree_secrete
-sh: 2: cannot create /tmp/entree_secrete: Permission denied
$ exit
```

Puis sur le compte *root* :

```
root@DEB13Server: ~#echo "+=+=+=+=" >> /tmp/entree_secrete
-bash: /tmp/entree_secrete: Permission non accordée
root@DEB13Server: ~#cat /tmp/entree_secrete
c'est ouvert
======
root@DEB13Server: ~#/tmp/entree_secrete
/tmp/entree_secrete: ligne 1: fin de fichier (EOF) prématurée lors de la recherche du « ' » correspondant
```

Cela ne marche pas, car le fichier appartient à *Luke,* il faut réaliser un ***chown*** **root** :

```
root@DEB13Server: ~#chown root /tmp/entree_secrete
root@DEB13Server: ~#echo "+=+=+=+=" >> /tmp/entree_secrete
root@DEB13Server: ~#cat /tmp/entree_secrete
c'est ouvert
======
+=+=+=+=
root@DEB13Server: ~#/tmp/entree_secrete
/tmp/entree_secrete: ligne 1: fin de fichier (EOF) prématurée lors de la recherche du « ' » correspondant
root@DEB13Server: ~#
```

Et enfin nous visualisons le fichier */etc/shadow* et du fichier */usr/bin*

```
root@DEB13Server: ~#ls -l /etc/shadow
-rw-r----- 1 root shadow 1346 18 déc.  15:06 /etc/shadow
root@DEB13Server: ~#ls -l /usr/bin/passwd
-rwsr-xr-x 1 root root 118168 19 avril  2025 /usr/bin/passwd
root@DEB13Server: ~#
```