TP10 : Les utilisateurs et les droits

1. La gestion des utilisateurs.

```
root@DS1 ~#id daemon
uid=1(daemon) gid=1(daemon) groupes=1(daemon)
root@DS1 ~#id luke
id: « luke » : utilisateur inexistant
```

1ère étape : Ici nous constatons avec la commande « **id daemon** » que l'utilisateur **daemon** existe, nous faisons de même avec l'utilisateur **luke,** « **id luke** » mais celui-ci n'existe pas.

```
root@DS1 ~#groupadd jedi
root@DS1 ~#groupadd rebelles
root@DS1 ~#_
```

2ème étape : Création des groupes **jedi** et **rebelles** avec les commandes « **groupadd jedi** » et « **groupadd rebelles** ».

```
root@DS1 ~#useradd -g jedi -G rebelles -m luke
root@DS1 ~#useradd -g jedi -m vador
root@DS1 ~#useradd -g rebelles -m solo
root@DS1 ~#id luke
uid=1002(luke) gid=1002(jedi) groupes=1002(jedi),1003(rebelles)
root@DS1 ~#id vador
uid=1003(vador) gid=1002(jedi) groupes=1002(jedi)
root@DS1 ~#id solo
uid=1004(solo) gid=1003(rebelles) groupes=1003(rebelles)
root@DS1 ~#
```

3ème étape : Par la suite nous créons les utilisateurs **luke** dans le groupe principal **jedi** et le groupe secondaire **rebelles**, **vador** dans le groupe **jedi** ainsi que **solo** dans le groupe **rebelles** avec les commandes suivantes :

- « **useradd -g jedi -G rebelles -m luke** »
- « **useradd -g jedi -m vador** »
- « **useradd -g rebelles -m solo** »

Puis avec la commande « **id** » nous observons que les utilisateurs font bien partie de leur groupe approprié.

```
root@DS1 ~#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@DS1 ~#tail -2 /etc/group
jedi:x:1002:
rebelles:x:1003:luke
```

4ème étape : Nous voulons afficher les trois dernières lignes du fichier **/etc/passwd** nous saisissons alors « **tail -3 /etc/passwd** », puis nous voulons les deux dernières lignes du fichier **/etc/group**, la commande est donc « **tail -2 /etc/group** ».

```
root@DS1 ~#passwd luke
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd: password updated successfully
```

5ème étape : Nous ajoutons le mot de passe « **password** » à l'utilisateur **luke** avec la commande « **passwd luke** ».

```
DS1 login: luke
Password:
Linux DS1 5.10.0-9-amd64 #1 SMP Debian 5.10.70-1 (2021-09-30) x86_64

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.
$
```

6ème étape : Via une seconde console nous vérifions donc que l'utilisateur **luke** ainsi que son mot de passe fonctionne bien en nous connectant sur ce compte.

```
root@DS1 ~#usermod -s /bin/bash luke
root@DS1 ~#
```

7ème étape : Après s'être déconnecté de l'utilisateur **luke**, nous modifions ce même compte afin de remplacer le shell sh par bash avec la commande « **usermod -s /bin/bash luke** ».

```
DS1 login: luke
Password:
Linux DS1 5.10.0-9-amd64 #1 SMP Debian 5.10.70-1 (2021-09-30) x86_64

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.
Last login: Wed Dec 15 11:21:13 CET 2021 on tty2
luke@DS1:~$ id
uid=1002(luke) gid=1002(jedi) groupes=1002(jedi),1003(rebelles)
luke@DS1:~$
```

8ème étape : De retour dans la seconde console nous observons le nouveau prompt accompagné de la commande « **id** ».

```
root@DS1 ~#useradd leia
root@DS1 ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia)
```

9ème étape : À nouveau sur la première console nous créons l'utilisateur **leia** avec la commande « **useradd leia** » puis en observant par la suite sont groupe principal avec « **id leia** ».

```
root@DS1 ~#ls -l /home
total 36
drwxr-xr-x 6 guest guest     4096  7 déc.  17:36 guest
drwx------ 2 root  root     16384 17 nov.  11:58 lost+found
drwxr-xr-x 2 luke      1004  4096 15 déc.  11:23 luke
drwxr-xr-x 2 sio   sio       4096 17 nov.  12:19 sio
drwxr-xr-x 2 solo  rebelles  4096 15 déc.  11:18 solo
drwxr-xr-x 2 vador jedi      4096 15 déc.  11:18 vador
```

10ème étape : Avec la commande « **ls -l /home** » nous vérifions si le répertoire personnel de l'utilisateur **leia** à été créé. Ici on constate que ce n'est pas le cas.

```
root@DS1 ~#usermod -G rebelles leia
root@DS1 ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1003(rebelles)
```

11ème étape : Nous ajoutons donc au groupe **rebelles** l'utilisateur **leia** avec la commande « **usermod -G rebelles leia** ».

```
root@DS1 ~#usermod -G jedi leia
root@DS1 ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1002(jedi)
```

12ème étape : Nous ajoutons donc au groupe **jedi** l'utilisateur **leia** avec la commande « **usermod -G jedi leia** », le compte **leia** quitte alors le groupe **rebelles**.

```
root@DS1 ~#usermod -G jedi,rebelles leia
root@DS1 ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1002(jedi),1003(rebelles)
```

13ème étape : Nous affectons cette fois-ci l'utilisateur **leia** au deux groupes avec la commande « **usermod -G jedi,rebelles leia** ».

```
root@DS1 ~#usermod -G "" leia
root@DS1 ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia)
```

14ème étape : Par la suite nous effectuons la commande « **usermod -G "" leia** » afin que l'utilisateur **leia** n'appartienne plus à aucun groupe secondaire.

```
root@DS1 ~#usermod -G jedi leia
root@DS1 ~#usermod -aG rebelles leia
root@DS1 ~#id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1002(jedi),1003(rebelles)
```

15ème étape : Ici nous voulons ajouter l'utilisateur à un groupe secondaire sans le retirer pour autant des autres, nous effectuons alors les commandes « **usermod -G jedi leia** » puis « **usermod -aG rebelles leia** » et pour finir « **id leia** » pour vérifier les actions que l'on vient d'effectuer.

```
root@DS1 ~#userdel leia
root@DS1 ~#_
```

16ème étape : Nous supprimons l'utilisateur leia avec la commande « **userdel leia** ».

```
root@DS1 ~#useradd -m leia
root@DS1 ~#cd /home/leia
root@DS1 /home/leia#su - leia
$ mkdir rep1
$ cd rep1
$ touch fichier1
$ ls -l
total 0
-rw-r--r-- 1 leia leia 0 15 déc.  11:30 fichier1
$ exit
root@DS1 /home/leia#cd
```

17ème étape : Nous recréons alors l'utilisateur **leia** avec un répertoire de connexion, « useradd -m leia ». Ensuite nous créons un répertoire ainsi qu'un fichier « **cd /home/leia** » et « **su - leia** ». La commande « **cd** » nous permet de sortir de ce mode.

```
root@DS1 ~#userdel -r leia
userdel: leia mail spool (/var/mail/leia) not found
root@DS1 ~#ls -l /home/leia
ls: impossible d'accéder à '/home/leia': Aucun fichier ou dossier de ce type
root@DS1 ~#id leia
id: « leia » : utilisateur inexistant
```

18ème étape : Nous supprimons le compte **leia** « **userdel -r leia** » ainsi que les fichiers de son répertoire de connexion avec « **ls -l /home/leia** », nous vérifions ensuite avec la commande « **id leia** ».

```
root@DS1 ~#groupadd -g 1007 leia
root@DS1 ~#useradd -u 1007 -g leia -m -s /bin/bash leia
root@DS1 ~#id leia
uid=1007(leia) gid=1007(leia) groupes=1007(leia)
root@DS1 ~#passwd leia
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd: password updated successfully
```

19ème étape : On recréer à l'identique l'utilisateur leia avec les mêmes uid et gid, les commandes sont « **groupadd -g 1007 leia** » et « **useradd -u 1007 -g leia -m -s /bin/bash leia** ». Nous vérifions le résultat avec la commande « **id leia** » et nous remettons le mot de passe avec « **passwd leia** ».

```
root@DS1 ~#useradd -u 0 -o -d /root -s /bin/bash toor
root@DS1 ~#id toor
uid=0(root) gid=0(root) groupes=0(root)
root@DS1 ~#passwd toor
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd: password updated successfully
```

20ème étape : Nous créons le compte toor en lui ajoutant les droits similaires à root, nous saisissons donc « **useradd -u 0 -o -d /root -s /bin/bash toor** », comme avant « **id toor** » pour vérifier et ajout du mot de passe **password** « **passwd toor** ».

```
DS1 login: toor
Password:
Linux DS1 5.10.0-9-amd64 #1 SMP Debian 5.10.70-1 (2021-09-30) x86_64

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.
Last login: Wed Dec 15 11:17:36 CET 2021 on tty1
root@DS1 ~#
```

21ème étape : Via la seconde console nous nous connectons à au compte **toor**.

```
root@DS1 ~#adduser palpatine
Ajout de l'utilisateur « palpatine » ...
Ajout du nouveau groupe « palpatine » (1004) ...
Ajout du nouvel utilisateur « palpatine » (1005) avec le groupe « palpatine » ...
Création du répertoire personnel « /home/palpatine »...
Copie des fichiers depuis « /etc/skel »...
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd: password updated successfully
Changing the user information for palpatine
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Cette information est-elle correcte ? [O/n]O
root@DS1 ~#id palpatine
uid=1005(palpatine) gid=1004(palpatine) groupes=1004(palpatine)
```

22ème étape : Toujours la seconde console nous voulons créer un utilisateur nommé **palpatine** respectant la charte Debian, la commande est « **adduser palpatine** », nous vérifions ensuite avec « **id palpatine** ».

```
root@DS1 ~#grep luke /etc/passwd
luke:x:1002:1002::/home/luke:/bin/bash
root@DS1 ~#grep rebelles /etc/group
rebelles:x:1003:luke
```

23ème étape : Nous affichons les caractéristiques de l'utilisateur **local luke** ainsi que celles du groupe local **rebelles**, « **grep luke /etc/passwd** » et « **grep rebelles /etc/group** ».

```
root@DS1 ~#getent passwd luke
luke:x:1002:1002::/home/luke:/bin/bash
root@DS1 ~#getent group jedi
jedi:x:1002:
root@DS1 ~#_
```

24ème étape : Cette fois-ci nous voulons les caractéristiques du simple utilisateur **luke** ainsi que celles du groupe **jedi**, les commandes sont donc « **getent passwd luke** » et « **getent group jedi** ».

2. La gestion des droits.

```
root@DS1 ~#mkdir /home/etoilenoire
root@DS1 ~#cd /home/etoilenoire
root@DS1 /home/etoilenoire#echo "voici les plans" > plans
root@DS1 /home/etoilenoire#echo "c'est ouvert" > entree_secrete
root@DS1 /home/etoilenoire#_
```

1ère étape : Nous créons une arborescence de fichier avec les commandes suivantes :

- « **mkdir /home/etoilenoir** »
- « **cd /home/etoilenoire** »

Puis :

- « **echo "voici les plans" > plans** »
- « **echo "c'est ouvert" > entree_secrete** »

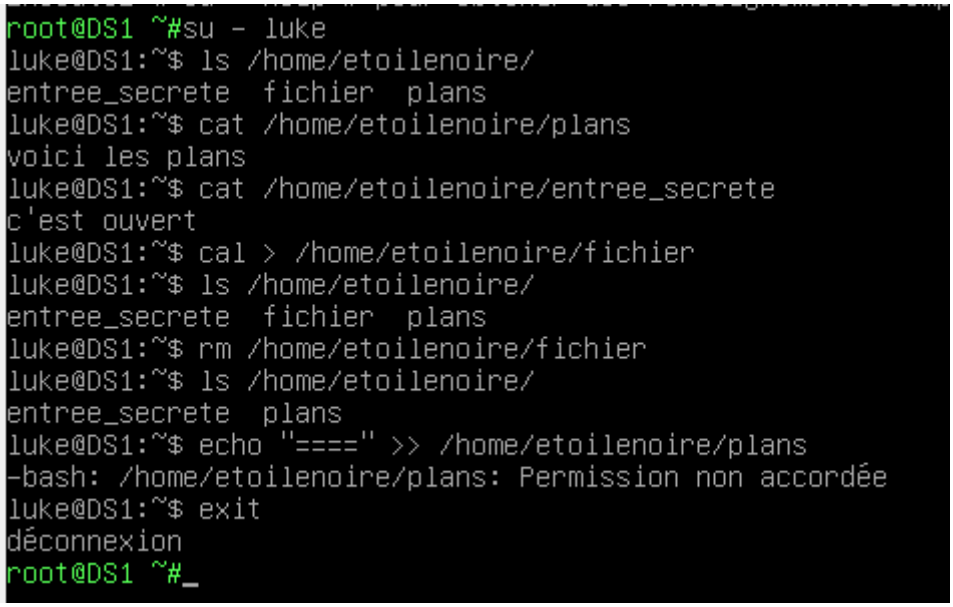
```
root@DS1 /home/etoilenoire#cd
root@DS1 ~#ls -ld /home/etoilenoire
drwxr-xr-x 2 root root 4096 15 déc.  11:47 /home/etoilenoire
root@DS1 ~#chown luke /home/etoilenoire
root@DS1 ~#chgrp jedi /home/etoilenoire
root@DS1 ~#chmod 750 /home/etoilenoire
root@DS1 ~#ls -ld /home/etoilenoire
drwxr-x--- 2 luke jedi 4096 15 déc.  11:47 /home/etoilenoire
root@DS1 ~#
```

2ème étape : Nous modifions les caractéristiques du répertoire **etoilenoire**, le propriétaire de celui-ci sera donc **luke** et son groupe **jedi**. Par la suite il sera accessible en lecture, écriture et accès au propriétaire. Il sera accessible en lecture et accès au groupe mais pas aux autres utilisateurs. Nous effectuons les commandes « **cd** », « **ls -l home/etoilenoire** », « **chown luke /home/etoilenoire** », « **chgrp jedi /home/etoilenoire** », « **chmod 750 /home/etoilenoire** » puis « **ls -ld /home/etoilenoire** ».

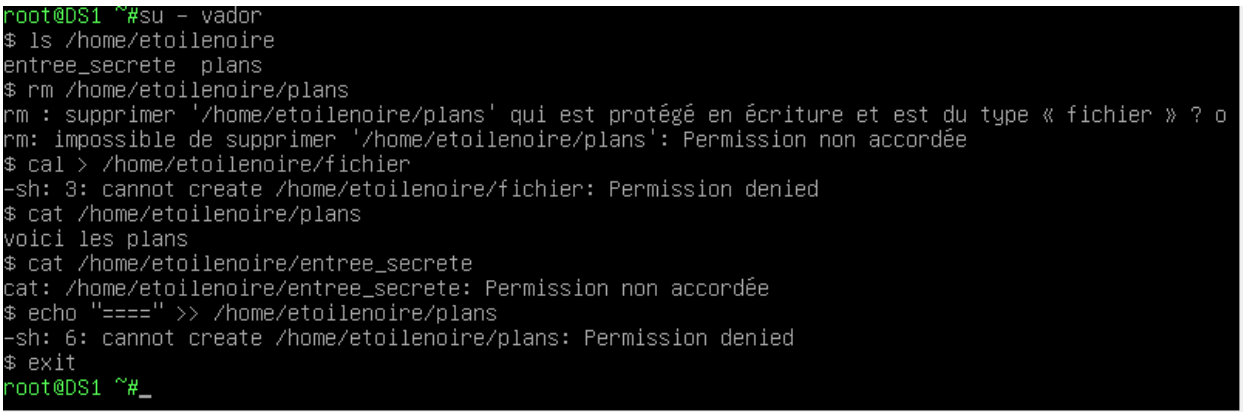
```
root@DS1 ~#chmod g=r,o=- /home/etoilenoire/*
root@DS1 ~#chgrp jedi /home/etoilenoire/plans
root@DS1 ~#chgrp rebelles /home/etoilenoire/entree_secrete
root@DS1 ~#ls -l /home/etoilenoire/
total 8
-rw-r----- 1 root rebelles 13 15 déc.  11:47 entree_secrete
-rw-r----- 1 root jedi     16 15 déc.  11:47 plans
root@DS1 ~#_
```

3ème étape : Nous modifions les caractéristiques des fichiers, ils seront accessibles en lecture seule pour le groupe et n'auront aucun droit pour les autres. On utilise la notation symbolique. On affilie le fichier plans au groupe jedi et le fichier entree_secrete au groupe rebelles. On utilise les commandes suivantes :

- « **chmod g=r ,o=- /home/etoilenoire/*** »
- « **chgrp jedi /home/etoilenoire/plans** »
- « **chgrp rebelles /home/etoilenoire/entree_secrete** »
- « **ls -l /home/etoilenoire/** »



4ème étape : Nous testons alors les accès. Tout d'abord avec le compte **luke,** on voit que l'utilisateur **luke** à presque toutes les permissions hormis celle pour la commande **« echo "====" ” /home/etoilenoire/plans** », il est donc interdit de modifier le fichier **plans** et **entree_secrete**.



5ème étape : Même manipulation mais cette fois avec le compte **vador**, celui-ci ne peut ni créer ni supprimer des fichiers, il ne peut pas lire le fichier **entree_secrete**, ne peut pas modifier le fichier **plans**. Nous le constatons avec les commandes « **rm /home/etoilenoire/plans** », « **cal > /home/etoilenoire/fichier** », « **cat /home/etoilenoire/entree_secrete** » et « **echo "====" >> /home/etoilenoire/plans** ».

```
root@DS1 ~#su - solo
$ ls /home/etoilenoire
ls: impossible d'ouvrir le répertoire '/home/etoilenoire': Permission non accordée
$ cal > /home/etoilenoire/fichier
-sh: 2: cannot create /home/etoilenoire/fichier: Permission denied
$ rm -f /home/etoilenoire/entree_secrete
rm: impossible de supprimer '/home/etoilenoire/entree_secrete': Permission non accordée
$ cat /home/etoilenoire/entree_secrete
cat: /home/etoilenoire/entree_secrete: Permission non accordée
$ exit
root@DS1 ~#_
```

6ème étape : Désormais depuis le compte **solo**, celui-ci ne possède aucun droit, tout les commandes effectuer lui donne comme résultat une **permission non accordée**.

```
root@DS1 ~# whereis uptime
uptime: /usr/bin/uptime /usr/share/man/man1/uptime.1.gz
root@DS1 ~#whatis uptime
uptime (1)           - Indiquer depuis quand le système a été mis en route
root@DS1 ~#uptime
 12:09:34 up 13 min,  1 user,  load average: 0,00, 0,00, 0,00
root@DS1 ~#ls -l /usr/bin/uptime
-rwxr-xr-x 1 root root 14560  6 avril  2021 /usr/bin/uptime
root@DS1 ~#chmod o-x /usr/bien/uptime
chmod: impossible d'accéder à '/usr/bien/uptime': Aucun fichier ou dossier de ce type
root@DS1 ~#chmod o-x /usr/bin/uptime
root@DS1 ~#ls -l /usr/bin/uptime
-rwxr-xr-- 1 root root 14560  6 avril  2021 /usr/bin/uptime
root@DS1 ~#su - luke
luke@DS1:~$ uptime
-bash: /usr/bin/uptime: Permission non accordée
luke@DS1:~$ exit
déconnexion
root@DS1 ~#chmod o+x /usr/bin/uptime
root@DS1 ~#ls -l /usr/bin/uptime
-rwxr-xr-x 1 root root 14560  6 avril  2021 /usr/bin/uptime
root@DS1 ~#su - luke
luke@DS1:~$ uptime
 12:12:08 up 16 min,  1 user,  load average: 0,00, 0,00, 0,00
luke@DS1:~$ exit
déconnexion
```

7ème étape : Nous supprimons temporairement le droit d'exécution pour la commande **uptime,** par la suite nous testons donc les conséquences de ces actions sur le compte **luke**. Les commandes sont les suivantes :

- « **whereis uptime** »
- « **whatis uptime** »
- « **uptime** »

Prise de connaissance et compréhension de la commande uptime

- **« ls -l /usr/bin/uptime »**
- « **chmod o-x /usr/bin/uptime** »
- « **ls -l /usr/bin/uptime** »

Suppression des droits d'exécution de la commande uptime.

- « **su – luke** »
- « **uptime** »

Test de la commande uptime après la modification

3. La gestion des droits, compléments.

```
root@DS1 ~#chmod 3770 /home/etoilenoire/
root@DS1 ~#ls -ld /home/etoilenoire
drwxrws--T 2 luke jedi 4096 15 déc.  12:01 /home/etoilenoire
root@DS1 ~#echo "fichier un" > /home/etoilenoire/f1
root@DS1 ~#su - luke
luke@DS1:~$ echo "bonjour" > /home/etoilenoire/f2
luke@DS1:~$ exit
déconnexion
```

1ère étape : Nous ajoutons des droits spéciaux au répertoire **etoilenoire**, par la suite nous créons des fichiers dans ce même répertoire avec le compte **root**, il sera nommé **f1**, le second avec le compte **luke** sera nommé **f2** et le troisième avec le compte **vador** sera nommé **f3**.
Nous effectuons les commande « **chmod 3370 /home/etoilenoire/** », « **ls -ld /home/etoilenoire** », « **echo "fichier un" > /home/etoilenoire/f1** », « **su - luke** » et « **echo "bonjour" > /home/etoilenoire/f2** ».

```
root@DS1 ~#su - vador
$ rm /home/etoilenoire/f2
rm : supprimer '/home/etoilenoire/f2' qui est protégé en écriture et est du type « fichier » ? o
rm: impossible de supprimer '/home/etoilenoire/f2': Opération non permise
$ exit
```

2ème étape : Sur le compte **vador** on essaie de détruire le fichier **f2** de **luke**, on conserve par ailleurs le droit sticky-bit avec la commande « **rm home/etoilenoire/f2** », nous obtenons **opération non permise**.

```
root@DS1 ~#chmod -t /home/etoilenoire/
root@DS1 ~#ls -ld /home/etoilenoire/
drwxrws--- 2 luke jedi 4096 15 déc.  12:14 /home/etoilenoire/
root@DS1 ~#su - vador
$ rm /home/etoilenoire/f2
rm : supprimer '/home/etoilenoire/f2' qui est protégé en écriture et est du type « fichier » ? o
$ ls -l /home/etoilenoire/f2
ls: impossible d'accéder à '/home/etoilenoire/f2': Aucun fichier ou dossier de ce type
$ exit
```

3ème étape : Cette fois-ci nous supprimons le droit sticky-bit, nous effectuons les commandes « **chmod -t /home/etoilenoire/** », « **ls -ld /home/etoilenoire/** », connexion au compte **vador** avec « **su - vador** » puis « **rm /home/etoilenoire/f2** » et « **ls -l /home/etoilenoire/f2** ».

```
root@DS1 ~#ls -l /dev/sda1
brw-rw---- 1 root disk 8, 1 15 déc.  11:56 /dev/sda1
```

4ème étape : Nous observons avec la commande « **ls -l /dev/sda1** » qui peut formater la partition **/dev/sda1** . Ici c'est root.

```
root@DS1 ~#cp -p /home/etoilenoire/* /tmp
root@DS1 ~#ls -l /tmp/plans /tmp/entree_secrete
-rw-r----- 1 root rebelles 13 15 déc.  11:47 /tmp/entree_secrete
-rw-r----- 1 root jedi     16 15 déc.  11:47 /tmp/plans
```

5ème étape : Nous copions avec l'administrateur **root** les fichiers du répertoire **etoilenoire** dans **/tmp** en conservant leurs attributs. Les commandes effectuer sont « **cp -p /home/etoilenoire/* /tmp** » et « **ls -l /tmp /plans /tmp/entree_secrete** ».

```
root@DS1 ~#chown luke /tmp/entree_secrete
root@DS1 ~#ls -l /tmp/entree_secrete
-rw-r----- 1 luke rebelles 13 15 déc.  11:47 /tmp/entree_secrete
```

6ème étape : **Root** donne donc le fichier **entree_secrete** à l'utilisateur **luke** avec la commande « **chown luke /tmp/entree_secrete** » puis « **ls -l /tmp/entree_secrete** ».

```
root@DS1 ~#su - luke
luke@DS1:~$ cat /tmp/entree_secrete
c'est ouvert
luke@DS1:~$ echo "======" >> /tmp/entree_secrete
luke@DS1:~$ cat /tmp/entree_secrete
c'est ouvert
======
luke@DS1:~$ /tmp/entree_secrete
-bash: /tmp/entree_secrete: Permission non accordée
luke@DS1:~$ exit
déconnexion
```

7ème étape : Nous testons les accès au fichier **/tmp/entree_secrete** depuis le compte **luke**, « **su -luke** », « **cat /tmp/entree_secrete** », « **echo "======" >> /tmp/entree_secrete** », « **cat /tmp/entree_secrete** » et « **/tmp/entree_secrete** ».

```
root@DS1 ~#su - solo
$ cat /tmp/entree_secrete
c'est ouvert
======
$ echo "++++++" >> /tmp/entree_secrete
-sh: 2: cannot create /tmp/entree_secrete: Permission denied
$ exit
```

8ème étape : Même opération mais depuis le compte solo, « **su - solo** », « **cat /tmp/entree_secrete** » et « **echo "++++++" >> /tmp/entree_secrete** ».

```
root@DS1 ~#cat /tmp/entree_secrete
c'est ouvert
======
root@DS1 ~#echo "+=+=+=+=" >> /tmp/entree_secrete
-bash: /tmp/entree_secrete: Permission non accordée
root@DS1 ~#cat /tmp/entree_secrete
c'est ouvert
======
root@DS1 ~#/tmp/entree_secrete
-bash: /tmp/entree_secrete: Permission non accordée
```

9ème étape : De même depuis le compte root, les commandes sont « **cat /tmp /entree_secrete** », « **echo "+=+=+=+=" >> /tmp/entree_secrete** », « **cat /tmp /entree_secrete** » et « **/tmp/entree_secrete** ».

```
root@DS1 ~#ls -l /etc/shadow
-rw-r----- 1 root shadow 1443 15 déc.  11:37 /etc/shadow
root@DS1 ~#ls -l /usr/bin/passwd
-rwsr-xr-x 1 root root 63960  7 févr.  2020 /usr/bin/passwd
```

10ème étape : Nous visualisons les droits du fichiers shadow ainsi que de la commande passwd, nous saisissons « **ls -l /etc/shadow** » et « **ls -l /usr/bin/passwd** ».