TP:13

1. La gestion des utilisateurs.

1. Est-ce que les comptes utilisateurs daemon et luke existent et, si oui, quels sont leurs uid, gid et leur(s) groupe(s) ?

```
root@DEB12Server: ~# id daemon
uid=1(daemon) gid=1(daemon) groupes=1(daemon)
root@DEB12Server: ~#id luke
id: ‘luke’ : utilisateur inexistant
root@DEB12Server: ~#_
```

2. Créez les groupes jedi et rebelles.

```
root@DEB12Server: ~# groupadd jedi
root@DEB12Server: ~# groupadd rebelles
root@DEB12Server: ~#
```

3. Consultez le manuel en ligne à l’aide de la commande man afin de découvrir les options de la commande useradd (man useradd)

```
-g, --gid GROUP
    The name or the number of the user's primary group. The group name must exist. A group number must refer to

    If not specified, the behavior of useradd will depend on the USERGROUPS_ENAB variable in /etc/login.defs.
    -U/--user-group is specified on the command line), a group will be created for the user, with the same name
    to no (or -N/--no-user-group is specified on the command line), useradd will set the primary group of the
    GROUP variable in /etc/default/useradd, or 100 by default.

-G, --groups GROUP1[,GROUP2,...[,GROUPN]]]
    A list of supplementary groups which the user is also a member of. Each group is separated from the next by
    The groups are subject to the same restrictions as the group given with the -g option. The default is for
    group. In addition to passing in the -G flag, you can add the option GROUPS to the file /etc/default/useradd
    those supplementary groups.

-h, --help
    Afficher un message d'aide et quitter.
```

4. Créez des comptes utilisateurs luke, vador et solo. Visualisez-les ensuite. Le compte luke appartient au groupe jedi (comme groupe principal) et au groupe rebelles (comme groupe secondaire). Le compte vador appartient au groupe jedi. Le compte solo fait partie du groupe rebelle

```
root@DEB12Server: ~# useradd -g jedi -G rebelles -m luke
root@DEB12Server: ~# useradd -g jedi -m vador
root@DEB12Server: ~# useradd -g rebelles -m solo
root@DEB12Server: ~# id luke
uid=1002(luke) gid=1002(jedi) groupes=1002(jedi),1003(rebelles)
root@DEB12Server: ~# id vador
uid=1003(vador) gid=1002(jedi) groupes=1002(jedi)
root@DEB12Server: ~# id solo
uid=1004(solo) gid=1003(rebelles) groupes=1003(rebelles)
root@DEB12Server: ~#_
```

TP:13

5. Affichez les dernières lignes des fichiers /etc/passwd et /etc/group

```
root@DEB12Server: ~# 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@DEB12Server: ~# tail -2 /etc/group
jedi:x:1002:
rebelles:x:1003:luke
root@DEB12Server: ~#
```

6. Mettez le mot « password » comme mot de passe à l'utilisateur luke.

```
root@DEB12Server: ~# passwd luke
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
root@DEB12Server: ~#_
```

7. Ouvrez une seconde console (Ctrl+Alt+F2) et connectez-vous sous le compte de luke. Considérez le prompt ($)

8. Déconnectez-vous (logout ou exit) et retournez sur la première console (Ctrl+Alt+F1). Modifiez le compte utilisateur luke afin de remplacer le shell sh par bash :

```
root@DEB12Server: ~# usermod -s /bin/bash luke
root@DEB12Server: ~#_
```

9. Reconnectez-vous sous le compte de luke dans la seconde console. Observez de nouveau le prompt

```
luke@DEB12Server:~$ id
uid=1002(luke) gid=1002(jedi) groupes=1002(jedi),1003(rebelles)
luke@DEB12Server:~$ exit_
```

10. Créez l'utilisateur leia dans la première console avec la commande useradd (sous le compte root). Quel est son groupe principal ?

```
root@DEB12Server: ~# id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia)
root@DEB12Server: ~#_
```

11. Est-ce que le répertoire personnel de l'utilisateur leia a été créé ? Pourquoi ?

```
root@DEB12Server: ~# ls -l /home
total 20
drwx------ 4 guest guest    4096 16 déc.  11:13 guest
drwx------ 2 luke  jedi     4096 17 déc.  10:23 luke
drwx------ 2 sio   sio      4096 10 déc.  10:15 sio
drwx------ 2 solo  rebelles 4096 17 déc.  09:53 solo
drwx------ 2 vador jedi     4096 17 déc.  09:52 vador
root@DEB12Server: ~#_
```

12. Gérez les groupes secondaires.
a) Affectez l'utilisateur leia au groupe rebelles (comme groupe secondaire).

```
root@DEB12Server: ~# usermod -G rebelles leia
root@DEB12Server: ~# id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1003(rebelles)
root@DEB12Server: ~#_
```

b) Affectez leia au groupe jedi. Leia quitte le groupe rebelles.

```
root@DEB12Server: ~# usermod -G jedi leia
root@DEB12Server: ~# id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1002(jedi)
root@DEB12Server: ~#_
```

c) Affectez leia aux groupes jedi et rebelles.

```
root@DEB12Server: ~# usermod -G jedi,rebelles leia
root@DEB12Server: ~# id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1002(jedi),1003(rebelles)
root@DEB12Server: ~#
```

d) On veut que leia n'appartienne plus à aucun groupe secondaire.

```
root@DEB12Server: ~# usermod -G '''' leia
root@DEB12Server: ~# id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia)
root@DEB12Server: ~#_
```

e) On veut ajouter l'utilisateur à un groupe secondaire sans le retirer des autres groupes secondaires (option -a).

```
root@DEB12Server: ~# usermod -G jedi leia
root@DEB12Server: ~# usermod -aG rebelles leia
root@DEB12Server: ~# id leia
uid=1005(leia) gid=1005(leia) groupes=1005(leia),1002(jedi),1003(rebelles)
root@DEB12Server: ~#_
```

13. Supprimez le compte leia.

```
root@DEB12Server: ~# userdel leia
root@DEB12Server: ~#
```

14. Recréez le compte leia avec cette fois-ci un répertoire de connexion. A partir du compte leia, créez un répertoire ainsi qu'un fichier.

```
root@DEB12Server: ~# useradd -m leia
root@DEB12Server: ~# cd /home/leia
root@DEB12Server: /home/leia# su - leia
$ mkdir rep1
$ cd rep1
$ touch fichier1
$ ls -l
total 0
-rw-rw-r-- 1 leia leia 0 17 déc.  10:41 fichier1
$ exit
root@DEB12Server: /home/leia# cd
root@DEB12Server: ~#_
```

15. Supprimez un compte utilisateur et les fichiers de son répertoire de connexion.

```
root@DEB12Server: ~# userdel -r leia
userdel : leia spool de courrier /var/mail/leia non trouvé
root@DEB12Server: ~# ls -l /home/leia
ls: impossible d'accéder à '/home/leia': Aucun fichier ou dossier de ce nom
root@DEB12Server: ~# id leia
id: ‘leia’ : utilisateur inexistant
root@DEB12Server: ~#
```

16. On veut recréer le compte leia à l'identique (cf. 10.) avec les mêmes uid et gid

```
root@DEB12Server: ~# groupadd -g 1007 leia
root@DEB12Server: ~# useradd -u 1007 -g leia -m -s /bin/bash leia
root@DEB12Server: ~# id leia
uid=1007(leia) gid=1007(leia) groupes=1007(leia)
root@DEB12Server: ~# passwd leia
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
root@DEB12Server: ~#
```

17. Créez le compte toor ayant les mêmes droits que root

```
root@DEB12Server: ~# useradd -u 0 -o -d /root -s /bin/bash toor
useradd : l'utilisateur 'toor' existe déjà
root@DEB12Server: ~# id toor
uid=0(root) gid=1008(toor) groupes=0(root)
root@DEB12Server: ~# passwd toor
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd : mot de passe mis à jour avec succès
root@DEB12Server: ~#_
```

18. Ouvrez une seconde console et connectez-vous avec le compte toor.

```
Debian GNU/Linux 13 DEB12Server tty2

DEB12Server login: toor
Password:
Linux DEB12Server 6.12.48+deb13-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.12.48-1 (2025-09-20) 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.
root@DEB12Server: ~#
```

19. Créez un compte d’utilisateur respectant la charte Debian avec la commande adduser

```
root@DEB12Server: ~# 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] y
root@DEB12Server: ~# id palpatine
uid=1005(palpatine) gid=1005(palpatine) groupes=1005(palpatine),100(users)
root@DEB12Server: ~#
```

20. Affichez les caractéristiques de l’utilisateur local luke et du groupe local rebelles

```
root@DEB12Server: ~# grep luke /etc/passwd
luke:x:1002:1002::/home/luke:/bin/bash
root@DEB12Server: ~# grep rebelles /etc/group
rebelles:x:1003:luke
root@DEB12Server: ~#_
```

2. La gestion des droits.

TP:13

1. Création d’une arborescence de fichiers.

```
root@DEB12Server: ~# mkdir /home/etoilenoire
root@DEB12Server: ~# cd /home/etoilenoire
root@DEB12Server: /home/etoilenoire# echo "voici les plans" > plans
root@DEB12Server: /home/etoilenoire# echo "c'est ouvert" > entree_secrete
root@DEB12Server: /home/etoilenoire#
```

2. Changement des caractéristiques du répertoire etoilenoire. Son propriétaire sera luke, son groupe jedi. Il sera accessible en lecture, écriture et accès au propriétaire (droits en octal). Il sera accessible en lecture et accès au groupe mais pas aux autres.

```
root@DEB12Server: ~# ls -ld /home/etoilenoire
drwxr-xr-x 2 root root 4096 17 déc.  11:03 /home/etoilenoire
root@DEB12Server: ~# chown luke /home/etoilenoire
root@DEB12Server: ~# chgrp jedi /home/etoilenoire
root@DEB12Server: ~# chmod 750 /home/etoilenoire
root@DEB12Server: ~# ls -ld /home/etoilenoire
drwxr-x--- 2 luke jedi 4096 17 déc.  11:03 /home/etoilenoire
root@DEB12Server: ~#
```

3. Changement des 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.

```
root@DEB12Server: ~# chmod g=r,o=- /home/etoilenoire/*
root@DEB12Server: ~# chgrp jedi /home/etoilenoire/plans
root@DEB12Server: ~# chgrp rebelles /home/etoilenoire/entree_secrete
root@DEB12Server: ~# ls -l /home/etoilenoire/
total 4
-rw-r----- 1 root rebelles  0 17 déc.  11:03 entree_secrete
-rw-r----- 1 root jedi     16 17 déc.  11:02 plans
root@DEB12Server: ~#
```

4. Test des accès. a) A partir du compte luke : L’utilisateur luke, en tant que propriétaire, a tous les droits sur le répertoire etoilenoire : il peut le lister, créer ou supprimer des fichiers dedans et il a accès aux fichiers qu’il contient. En tant que membre du groupe jedi, il peut lire le fichier plans, et en tant que membre du groupe rebelles, il peut lire le fichier entree_secrete. Par contre, il ne peut pas modifier le fichier plans (ainsi que le fichier entree_secrete). Seul root peut le faire

```
root@DEB12Server: ~# su - luke
luke@DEB12Server:~$ ls /home/etoilenoire
entree_secrete  plans
luke@DEB12Server:~$ cat /home/etoilenoire/plans
voici les plans
luke@DEB12Server:~$ cat /home/etoilenoire/entree_secrete
luke@DEB12Server:~$ cal > /home/etoilenoire/fichier
luke@DEB12Server:~$ ls /home/etoilenoire/
entree_secrete  fichier  plans
luke@DEB12Server:~$ rm /home/etoilenoire/fichier
luke@DEB12Server:~$ ls /home/etoilenoire/
entree_secrete  plans
luke@DEB12Server:~$ echo "====" >> /home/etoilenoire/plans
-bash: /home/etoilenoire/plans: Permission non accordée
luke@DEB12Server:~$ exit
```

b) A partir du compte vador : L'utilisateur vador, en tant que membre du groupe jedi, peut lister le répertoire etoilenoire. Il a accès également aux fichiers qu'il contient. Par contre, il ne peut ni créer ni supprimer des fichiers dedans. En tant que membre du groupe jedi, il peut lire le fichier plans mais pas le fichier entree_secrete. Il ne peut pas modifier le fichier plans. Seul root peut le faire.

```
root@DEB12Server: ~# su - vador
$ ls /home/etoilenoire
entree_secrete  plans
$ rm /home/etoilenoire/plans
rm : supprimer '/home/etoilenoire/plans' qui est protégé en écriture et est du type « regular file » ? y
rm: impossible de supprimer '/home/etoilenoire/plans': Permission non accordée
$ cal > /home/etoilenoire/fichier
-sh: 3: cannot create /home/etoilenoire/fichier: Permission denied
$ cat /home/etoilenoire/plans
voici les plans
$ cat /home/etoilenoire/entree_secrete
cat: /home/etoilenoire/entree_secrete: Permission non accordée
$ echo "====" >> /home/etoilenoire/plans
-sh: 6: cannot create /home/etoilenoire/plans: Permission denied
$ exit_
```

c) A partir du compte solo : L'utilisateur solo (groupe rebelles) n'a aucun droit sur le répertoire etoilenoire (cf. droits other) : il ne peut pas connaître son contenu, il ne peut ni ajouter ni supprimer des fichiers à l'intérieur. Il n'a aucun accès aux fichiers de ce répertoire quels que soient les droits sur ces fichiers (cf. droit de lecture du groupe rebelles sur entree_secrete).

```
root@DEB12Server: ~# 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_
```

5. Suppression temporaire du droit d'exécution à la commande uptime. Testez les conséquences à partir du compte luke.

```
root@DEB12Server: ~# whereis uptime
uptime: /usr/bin/uptime /usr/share/man/man1/uptime.1.gz
root@DEB12Server: ~# whatis uptime
uptime (1)           - Indiquer depuis quand le système a été mis en route
root@DEB12Server: ~# uptime
 15:44:56 up 46 min,  1 user,  load average: 0,11, 0,05, 0,01
root@DEB12Server: ~# ls -l /usr/bin/uptime
-rwxr-xr-- 1 root root 14648 30 juil. 13:58 /usr/bin/uptime
root@DEB12Server: ~# chmod o-x /usr/bin/uptime
root@DEB12Server: ~# ls -l /usr/bin/uptime
-rwxr-xr-- 1 root root 14648 30 juil. 13:58 /usr/bin/uptime
root@DEB12Server: ~# su - luke
luke@DEB12Server:~$ uptime
-bash: /usr/bin/uptime: Permission non accordée
luke@DEB12Server:~$ exit
```

```
root@DEB12Server: ~# chmod o+x /usr/bin/uptime
root@DEB12Server: ~# ls -l /usr/bin/uptime
-rwxr-xr-x 1 root root 14648 30 juil. 13:58 /usr/bin/uptime
root@DEB12Server: ~# su - luke
luke@DEB12Server:~$ uptime
 15:52:35 up 54 min,  1 user,  load average: 0,00, 0,01, 0,00
luke@DEB12Server:~$ exit
```

3. La gestion des droits, compléments.

1. Ajoutez les droits SGID et sticky-bit au répertoire etoilenoire (soit 2000 et 1000 en octal). Ensuite, pour vérifier l'impact de ces droits, on crée des fichiers dans le répertoire etoilenoire. Sous le compte root, on crée le fichier f1. Sous le compte luke, on crée le fichier f2 et sous le compte vador on crée le fichier f3

```
root@DEB12Server: ~# chmod 3770 /home/etoilenoire/
root@DEB12Server: ~# ls -ld /home/etoilenoire/
drwxrws--T 2 luke jedi 4096 18 déc.  15:24 /home/etoilenoire/
root@DEB12Server: ~# echo "fichier un" > /home/etoilenoire/f1
root@DEB12Server: ~# su - luke
luke@DEB12Server:~$ echo "bonjour" > /home/etoilenoire/f2
luke@DEB12Server:~$ exit
```

```
root@DEB12Server: ~# su - vador
$  echo "bonjour" > /home/etoilenoire/f3
$ exit
root@DEB12Server: ~# ls -l /home/etoilenoire/f?
-rw-r--r-- 1 root  jedi 11 18 déc.  15:55 /home/etoilenoire/f1
-rw-r--r-- 1 luke  jedi  8 18 déc.  15:56 /home/etoilenoire/f2
-rw-r--r-- 1 vador jedi  8 18 déc.  15:57 /home/etoilenoire/f3
root@DEB12Server: ~#
```

2. Vador va essayer de détruire le fichier de luke.

a) On conserve le droit sticky-bit.

```
root@DEB12Server: ~# su - vador
$ rm /home/etoilenoire/f2
rm : supprimer '/home/etoilenoire/f2' qui est protégé en écriture et est du type « regular file » ? y
rm: impossible de supprimer '/home/etoilenoire/f2': Opération non permise
$ exit
root@DEB12Server: ~#
```

b) On supprime le droit sticky-bit.

```
root@DEB12Server: ~# chmod -t /home/etoilenoire/
root@DEB12Server: ~# ls -ld /home/etoilenoire/
drwxrws--- 2 luke jedi 4096 18 déc.  15:57 /home/etoilenoire/
root@DEB12Server: ~# su - vador
$ rm /home/etoilenoire/f2
rm : supprimer '/home/etoilenoire/f2' qui est protégé en écriture et est du type « regular file » ? y
$ ls -l /home/etoilenoire/f2
ls: impossible d'accéder à '/home/etoilenoire/f2': Aucun fichier ou dossier de ce nom
$ exit
root@DEB12Server: ~#
```

3. Qui peut formater la partition /dev/sda1 ?

```
root@DEB12Server: ~# ls -l /dev/sda1
brw-rw---- 1 root disk 8, 1 18 déc.  14:58 /dev/sda1
root@DEB12Server: ~#_
```

4. L’administrateur copie les fichiers du répertoire etoilenoire dans /tmp en conservant leurs attributs

```
root@DEB12Server: ~# cp -p /home/etoilenoire/* /tmp
root@DEB12Server: ~# ls -l /tmp/plans /tmp/entree_secrete
-rw-r----- 1 root rebelles  0 17 déc.  11:03 /tmp/entree_secrete
-rw-r----- 1 root jedi     16 17 déc.  11:02 /tmp/plans
root@DEB12Server: ~#
```

5. L'administrateur donne le fichier entree_secrete à luke.

```
root@DEB12Server: ~# chown luke /tmp/entree_secrete
root@DEB12Server: ~# ls -l /tmp/entree_secrete
-rw-r----- 1 luke rebelles 0 17 déc.  11:03 /tmp/entree_secrete
root@DEB12Server: ~#_
```

6. Test des accès (r,w,x) au fichier /tmp/entree_secrete. a) A partir du compte luke :

```
root@DEB12Server: ~# su - luke
luke@DEB12Server:~$ cat /tmp/entree_secrete
luke@DEB12Server:~$ echo "======" >> /tmp/entree_secrete
luke@DEB12Server:~$ cat /tmp/entree_secrete
======
luke@DEB12Server:~$ /tmp/entree_secrete
-bash: /tmp/entree_secrete: Permission non accordée
luke@DEB12Server:~$ exit
```

b) A partir du compte solo :

```
root@DEB12Server: ~# su - solo
$ cat /tmp/entree_secrete
======
$ echo "++++++" >> /tmp/entree_secrete
-sh: 2: cannot create /tmp/entree_secrete: Permission denied
$ exit_
```

c) A partir du compte root :

```
root@DEB12Server: ~# cat /tmp/entree_secrete
======
+=+=+=+=
root@DEB12Server: ~# echo "+=+=+=+=" >> /tmp/entree_secrete
root@DEB12Server: ~# cat /tmp/entree_secrete
======
+=+=+=+=
+=+=+=+=
root@DEB12Server: ~# /tmp/entree_secrete
-bash: /tmp/entree_secrete: Permission non accordée
root@DEB12Server: ~#
```

7. Visualisez les droits du fichier shadow et de la commande passwd.