
TP N° 1 : Introduction - Correction

Création de comptes

A) Consulter le manuel et relever les différences entre les 2 jeux de commandes :
adduser, deluser, addgroup, delgroup **et** useradd, userdel, groupadd, groupdel

Utiliser le premier jeu de commandes pour :

1. Créer un utilisateur **ens**, appartenant à un groupe **ens**, avec pour répertoire de connexion **/home/ens**, finger non renseigné, lançant un bash
adduser ens
2. Créer un groupe **etu** ayant pour GID:20000
addgroup --gid 20000 etu
3. Ajouter le groupe secondaire **etu** pour l'utilisateur **ens**
usermod --groups etu ens
4. Détruire tous les éléments créés
deluser ens ; delgroup etu ; rm -rf /home/etu

B) En utilisant les options (et non la saisie interactive) utiliser le jeu de commandes **adduser**, **deluser**, **addgroup**, **delgroup** pour :

- Créer un utilisateur avec:
 - pour nom de login **lp1**, avec un UID à **10001**,
 - ayant pour groupe de connexion le groupe **lpmrit** de GID **10000**,
 - dont le répertoire de connexion, sera **/home/lp1**,
 - et, comme nom complet : LP MRIT, numéro de bureau: 3A12, téléphone professionnel : 03.20.43.43.43
 - lançant un shell **bash**
- Donner la commande pour contrôler l'enregistrement des infos, et vérifier

```
root@stretch:/home/user# addgroup --gid 10000 lpmrit
Ajout du groupe « lpmrit » (GID 10000)...
Fait.
```

```
root@stretch:/home/user# adduser --uid 10001 --gid 10000 --disabled-login --gecos
LPMRIT,3A12,03.20.43.43.43,, lp1
Ajout de l'utilisateur « lp1 » ...
Ajout du nouvel utilisateur « lp1 » (10001) avec le groupe « lpmrit » ...
Création du répertoire personnel « /home/lp1 »...
Copie des fichiers depuis « /etc/skel »...
```

```
root@stretch:/home/user# getent passwd lp1
lp1:x:10001:10000:LPMRIT,3A12,03.20.43.43.43,:/home/lp1:/bin/bash
```

Deuxième jeu de commandes **useradd**, **userdel**, **groupadd**, **groupdel**:

- Créer un utilisateur avec:
 - pour nom de login, avec un UID à 10002,
 - ayant pour groupe de connexion le groupe **lpcgir**,
 - dont le répertoire de connexion, sera **/home/lp2**.
 - avec le même finger et lançant un bash

- Donner la commande pour contrôler l'enregistrement des infos, et vérifier

```
root@stretch:/home/user# useradd -u 10002 -g 10000 -m lp2
root@stretch:/home/user# chfn -f "LP MRIT" -r 3A12 -w "03.20.43.43.43" lp2
root@stretch:/home/user# chsh -s /bin/bash lp2
```

```
root@stretch:/home/user# getent passwd lp2
lp2:x:10002:10000:LP MRIT,3A12,03.20.43.43.43,:/home/lp2:/bin/bash
```

Faites en sorte de vous connecter en tant que **lp1** puis **lp2** pour créer un fichier vide **test.txt** dans leur répertoire de connexion respectif

NPO : attributions de mots de passe `passwd lp1 ; passwd lp2`
`lp1` et `lp2` : `touch ~/test.txt`

D) Modifications

- Changer le nom du groupe **lpmrit** en **mrity**
 - `groupmod -n mrit lpmrit`
- Vérifier que **lp2** appartient au groupe **mrity**
 - `id -g -n lp2`
- Créer un groupe **cgir** avec comme **gid 10002**
 - `addgroup --gid 10002 cgir`
- Déclarer le groupe **cgir** comme groupe principal pour **lp1** et **lp2**.
 - `usermod -g 10002 lp1`
 - `usermod -g 10002 lp2`
- A quel groupe appartiennent les répertoires de connexions de ces utilisateurs? Vérifier votre hypothèse
 - `root@stretch:/home/user# ls -l /home|grep 'lp[12]'`
 - `drwxr-xr-x 2 lp1 cgir 4096 sept. 17 11:39 lp1`
 - `drwxr-xr-x 2 lp2 cgir 4096 sept. 17 11:53 lp2`
- A quel groupe appartiennent les fichiers **test.txt** préalablement créés par ces utilisateurs? Vérifier votre hypothèse
 - `root@stretch:/home/user# ls -l /home/lp[12]/test.txt`
 - `-rw-r--r-- 1 lp1 cgir 0 sept. 17 13:54 /home/lp1/test.txt`
 - `-rw-r--r-- 1 lp2 cgir 0 sept. 17 13:54 /home/lp2/test.txt`
- Quelle commande doivent saisir **lp1** et **lp2** pour que les membres de leur groupe puissent créer des sous-répertoires dans leur répertoires de connexions ?
 - `chmod g+w ~`
- Quelle commande permet à **lp1** de créer un sous-répertoire **test** dans le répertoire de connexion de **lp2**
 - `mkdir ~/lp2/test => non autorisée`
- Ajouter le groupe **mrity** comme groupe secondaire pour **lp1** et **lp2**
- 3 solutions :
 1. Editer `/etc/group (VI)` : `mrity:x:10000:lp1,lp2`
 2. `usermod -G mrit lp1`
 3. `usermod --groups mrit lp2`
- Quelle commande permet à **lp2** de créer un sous-répertoire **test** dans le répertoire de connexion de **lp1**
 - `mkdir ~/lp1/test => OK`

Détruire tous les comptes utilisateurs et groupes créés pour l'exercice à l'aide des commandes de suppression appropriées.

```
deluser lp1;deluser lp2 ;delgroup cgir;delgroup mrit
```

Vérifier que les répertoires de connexion ont été supprimés.

```
rm -rf ~user[12]Changer d'identité
```

SUDO

- **sudo** permet d'exécuter une commande avec les droits d'un autre utilisateur en fournissant son propre mot de passe (à condition d'avoir été déclaré dans sudoers)
 - *pendant 15 mn il n'est plus nécessaire de fournir de mot de passe pour exécuter d'autres commandes sudo sous la même identité*

Exercice

Remarque: pour lancer un terminal, ou une application, au d'un autre utilisateur : changer d'identité puis lancer en tâche de fond, p.ex : su ... pluma &;... xterm& ;

Accorder à lp1 la possibilité d'exécuter toutes les commandes en tant que root

```
visudo :  
# User privilege specification  
root  ALL=(ALL:ALL) ALL  
lp1  ALL=(ALL:ALL) ALL
```

Vérifier en accédant au fichier shadow en étant lp1 :

```
Depuis le terminal de lp1 : sudo cat /etc/shadow (mdp:lp1)
```

En tant que lp1 :

1. modifier le masque de permission pour que seul le propriétaire puisse modifier les nouveaux fichiers créés

```
umask 0066
```
2. créer un fichier ~lp1/root appartenant à root,

```
sudo touch ~lp1/root
```
3. créer un fichier ~lp1/lp1 appartenant à lp1,

```
touch ~/lp1
```
4. créer un fichier ~lp1/lp2 appartenant à lp2.

```
sudo touch ~lp1/lp2  
sudo chown lp2:lp lp2
```

Contrôler :

```
ls -l ~
```

Modifier sudoers pour autoriser à lp2 l'usage de la commande cat en tant que lp1

```
lp2  ALL=(lp1:lp) /bin/cat
```

Vérifier en affichant le contenu du fichier ~lp1/lp1 depuis un terminal de lp2

```
sudo -u lp1 cat ~lp1/lp1
```

Les fichiers d'environnement

Pour tester les fichiers d'environnement nous allons voir la commande **alias** qui permet de créer des alias de commandes.

Sur la ligne de commandes saisir : **alias** liste="ls -l" # uniquement pour le shell courant

Vérifier en utilisant l'alias *liste* en lieu et place de la commande `ls -l`.

Exercice

A) Créer un alias **heure** affichant l'heure sous la forme : 10:12

alias heure='date +%H:%M'

Créer un alias **jour** affichant le jour sous la forme : 15/09

alias jour='date +%d/%m'

Créer un alias **quand** affichant le jour et l'heure utilisant les alias ci-dessus

alias quand='jour;heure'

Persistence : Placer la définition de ces alias dans le fichier **~lp1/.bash_profile** (ce fichier est à créer, le *.profile* ne sera plus lu par le bash)

Ouvrir une console **lp1** et tester.

Lancer un nouveau shell, les alias sont-ils encore connus ? Remédier au problème

copier dans `~lp1/.bashrc`

Que faire pour que dorénavant les nouveaux utilisateurs créés aient accès à ces alias ?

copier dans `/etc/skel`

B) Ajouter au début de tous (i.e. pour *commun*, *root*, *skel*, *lp*x) les fichiers d'environnement:

echo "lecture du fichier d'environnement : xxx "

où xxx est le chemin absolu fichier d'environnement

Deviner quelles sont les lignes qui s'affichent quand **lp1** se connecte ? puis vérifier

De même, **deviner**, puis vérifier, l'affichage lorsque depuis un terminal, lp1 tente de :

1. devenir root : su
2. devenir lp2 : su lp2.....
3. devenir root : su -.....
4. revenir à lp2 : Ctrl+D.....
5. ouvrir un nouveau terminal: xterm &.....
6. devenir lp3 : su - lp3.....
7. ouvrir une console, et se connecter en tant que lp4 : Ctrl-Alt-F1.....

Revenir au terminal ouvert pour lp1 dans l'environnement graphique : Alt-F7

1. créer un nouveau bash fils : **bash**
2. créer un nouveau shell fils : **sh**

c) Modifier le fichier d'environnement approprié pour que les nouveaux **identifiants** respectent le format suivant :

- comptent au moins 8 caractères
- commencent par un chiffre
- comptent au moins une majuscule

- se terminent par un chiffre
- ne contiennent ni "." ni ":"

Premier script

Ecrire un script **accueil.sh** qui affiche les informations sur le compte sous la forme suivante :

```
Bonjour, lp5
Vous appartenez aux groupes suivants : mrit
Répertoire courant : /home/lp5
Nombre de fichiers dans votre arborescence : 28
(sans compter les fichiers silencieux)
```

vi ~/accueil.sh

#!/bin/bash

```
#####
#   Script accueil.sh : affiche des informations sur le compte courant #
#####
```

```
echo -n "Bonjour, ";id -nu
echo -n "Vous appartenez aux groupes suivants : ";id -nG
echo -n "Répertoire courant : ";pwd
echo -n "Nombre de fichiers dans votre arborescence : "
find ~ -type f -name "[^.]*" 2>/dev/null | wc -l
```

chmod a+x ~/accueil.sh

~/accueil.sh

Faire en sorte que ce script soit exécuté à chaque ouverture de session pour lp5
insérer dans le fichier ~/lp5/.bash_profile la ligne : ~/lp5/accueil.sh

Faire en sorte que ce script soit exécuté à chaque ouverture de session pour tous les
utilisateurs
insérer dans le fichier /etc/profile la ligne : ~/lp5/accueil.sh
pour ~lp5 autoriser la lecture et l'exécution pour tous
chmod a+rx ~lp5 ~/lp5/accueil.sh