

## 1 La commande chmod

1. À l'aide d'un éditeur de texte, créer un fichier carte\_visite.txt dans votre répertoire personnel. Ce fichier contient sur trois lignes les informations suivantes : votre nom, prénom et adresse e-mail.

`nano carte_visite.txt`

→ nano permet d'écrire directement sur le fichier txt depuis le shell tout en le créant.

2. Afficher le contenu de ce fichier.

`Cat carte_visite.txt`

→ lis le fichier (son contenu)

3. À l'aide des commandes id et ls -l trouver si votre voisin peut lire ou éditer ce fichier.

`id` → montre les droit

`ls -l ~/carte_visite.txt`

`ls` → dresse la liste | -l montre les droits | ~/carte\_visite.txt (on spécifie le fichier)

4. Quels sont les droits d'accès du fichier carte\_visite.txt ?

Propriétaire : lecture/écriture

Groupe : lecture

Autres : lecture

Donc rw-r--r-- (octal : 644).

5. Donner aux utilisateurs qui appartiennent au même groupe que vous le droit de lire et modifier ce fichier.

Exprimer ces droits de deux manières : code octal et approche symbolique. Demander à un voisin de modifier votre fichier pour vérifier le bon fonctionnement.

`chmod 664 ~/carte_visite.txt`

# chmod 664 : permissions en octal => rw-rw-r--

`chmod g+rw ~/carte_visite.txt`

# chmod g+rw : ajoute lecture et écriture au groupe

Donc rw-rw- -(octal : 640)

Faut attendre un peu des fois mais normalement c'est :

`nano ../maxence_debailleul/carte_visite.txt`

6. À l'aide de la commande umask changer les droits par défaut sur vos fichiers pour qu'ils soient en lecture/écriture exclusives pour vous seulement. Exprimer les nouveaux droits d'accès de deux manières : code octal et approche symbolique.

**umask 077**

umask 077 : tous les fichiers créés n'auront que rw- --- --- (600)

touch test.txt

ls -l test.txt

Vérifie que test.txt a bien les permissions -rw-----

7. Donner une commande qui permet de trouver tous les fichiers dans votre compte qui sont accessibles par les autres utilisateurs en mode écriture.

find ~ -type f -perm -002

find : cherche dans le dossier perso (~)

-type f : seulement les fichiers

-perm -002 : permission d'écriture pour "others"

8. Faire une copie du fichier carte\_visite.txt en le renommant carte\_visite.copie. Quels sont les droits associés au nouveau fichier ? Justifier.

cp ~/carte\_visite.txt ~/carte\_visite.copie

ls -l ~/carte\_visite.copie

cp : copie le fichier. Les permissions du fichier copié sont les mêmes que le fichier original, sauf si umask s'applique

si umask=077, alors rw-----.

9. Remettre umask à sa valeur usuelle.

**umask 022**

10. Créez à la racine de votre compte un répertoire Public

mkdir ~/Public

11. Trouvez le moyen de rendre accessible en lecture ce répertoire à vos voisins.

chmod o+r ~/Public

Ou chmod 755 ~/Public

12. Trouvez le moyen de rendre accessible en écriture ce répertoire à vos voisins.

chmod o+rwx ~/Public

Ou chmod 773 ~/Public

13. Demandez ensuite à votre voisin de créer un fichier dans ce répertoire et affichez ensuite le contenu du répertoire pour vérifier qui est le propriétaire du fichier.

touch ~/Public/test.txt

ls -l ~/Public

→ Le propriétaire du fichier est le voisin qui l'a créé.

14. Effacez le répertoire Public une fois les manipulations terminées

rm -r ~/Public

rm -r : supprime récursivement le dossier et son contenu

## 2 << , < , > et >>

1. Donner une commande qui permet de mettre dans le fichier sortie.txt avec les noms des fichiers qui se trouvent dans le répertoire /usr/include et qui ont l'extension .h

ls /usr/include/\*.h > sortie.txt

ls .../\*.h : liste les fichiers

\*.h ; > ~/sortie.txt : écrase/écrit la sortie dans ~/sortie.txt

2. Visualisez le contenu du fichier sortie.txt

cat sortie.txt

# cat : affiche le contenu de ~/sortie.txt sur l'écran

3. Remplacez le contenu du fichier sortie.txt par le contenu de votre répertoire personnel.

ls ~ > sortie.txt

ls ~ : liste le contenu du répertoire personnel (~ est un raccourci vers ton dossier home).

> : redirige la **sortie standard** (tout ce que ls affiche) vers un fichier.

sortie.txt : nom du fichier dans lequel la sortie est **écrasée** (si le fichier existe déjà, son contenu est remplacé).

4. Ajouter à la fin du fichier sortie.txt le résultat de la commande pwd

pwd >> sortie.txt

pwd : affiche le chemin complet du répertoire courant (exemple : /home/ton\_user).

>> : redirige la sortie standard **en ajoutant** (append) à la fin du fichier sortie.txt sans effacer son contenu.

sortie.txt : fichier dans lequel on ajoute la sortie.

5. Lors de l'exécution de la commande find /var -name \*.log , il y a des erreurs. Trouver un moyen d'envoyer ces erreurs dans un fichiers nomé erreur.txt et les réponses à la commande dans trouve.txt !

find /var -name "\*.log" > trouve.txt 2> erreur.txt

find /var -name "\*.log" : cherche dans le dossier /var tous les fichiers dont le nom finit par .log.

Les guillemets autour de "\*.log" empêchent le shell d'interpréter le \* avant la commande.

> trouve.txt : redirige la **sortie standard** (les fichiers trouvés) dans le fichier trouve.txt, en écrasant ce fichier s'il existe.

2> erreur.txt : redirige la **sortie d'erreur** (messages d'erreur, par exemple permissions refusées) dans le fichier erreur.txt.

### 3. Patchs

1) Faire une copie du fichier sortie.txt en copie.txt

```
cp ~/sortie.txt ~/copie.txt
```

# cp : copie le fichier sortie.txt en copie.txt dans le même répertoire

2) Ajouter à la fin de copie.txt la ligne : "\*\*\*\* fin du fichier\*\*\*"

```
echo "*** fin du fichier***" >> ~/copie.txt
```

# echo : écrit la ligne ; >> ajoute cette ligne à la fin du fichier copie.txt

3) Utiliser wc (word count) pour comparer sortie.txt et copie.txt

```
wc ~/sortie.txt ~/copie.txt
```

# wc affiche nombre de lignes, mots et caractères pour chaque fichier

4) Comparer les deux fichiers avec diff

```
diff ~/sortie.txt ~/copie.txt
```

# diff affiche les différences entre les deux fichiers

5) Sauvegarder le résultat de diff dans diff.patch

```
diff ~/sortie.txt ~/copie.txt > ~/diff.patch
```

# > redirige la sortie de diff dans le fichier diff.patch

6) Appliquer le patch sur sortie.txt avec le fichier diff.patch

```
patch ~/sortie.txt ~/diff.patch
```

# patch applique les modifications contenues dans diff.patch sur sortie.txt

→ sortie.txt est maintenant identique à copie.txt.

7) Vérifier que sortie.txt a été modifié (par exemple afficher la fin)

```
tail ~/sortie.txt
```