

Linux Commande de base

Exercice 1

1. Créez deux groupes group1 et group2 ? (addgroup)
2. Créez quatre utilisateurs user1, user2, user3 et user4 ?(adduser ou useradd)
3. Rendez les utilisateurs dans les groupes : (usermod name_gp name_user)
 - Les premier et deuxième utilisateurs sont membres du premier groupe.
 - Les troisième et quatrième utilisateurs sont membres du second groupe.
 - Le deuxième utilisateur est aussi membre du second groupe.
 - Le quatrième utilisateur est aussi membre du premier groupe.
4. Vérifier les membres du groupe group2 ? (utilisé la commande : lid -g grp_name)
5. Créer deux répertoires rep1, rep2 et rep3 en seul ligne ?(mkdir name_rep)
6. Créer dans rep1 un fichier nommé fich11 et dans rep2 un répertoire nommé rep21 ? (touch name_file)
7. Déplacez-vous au répertoire rep21 (cd)
8. Copiez le rep1 et son contenu dans le répertoire courant ? (cp -r src dest)
9. Copiez l'arbre rep2 dans le répertoire rp3 ? (cp -r src/* dest)
10. Visualisez le contenu de rep3 de façon détaillée ? (ls -l name_rep)
11. Supprimez l'arbre rep3 ? (rm -r name_rep)

Exercice 2

1. se connecter en administrateur linux (su)
2. afficher les fichiers relatifs aux configurations utilisateurs (consulter le répertoire /etc)
3. afficher le fichier /etc/passwd par trois commandes différentes (cat,more,less)
4. on veut afficher l'identité de l'utilisateur user (id)
5. Etablir l'utilisateur Tux1 avec conditions suivantes : (usermod)
6. son home directory /home/toto1 (créer la d'abord)
7. son id 508
8. son GId = identifiant du groupe rtoto (créer le groupe d'abord)
9. son mot de passe est 0/toto1 (passwd)
10. Changer son mot de passe par 0/guest1 (passwd)
11. Afficher la ligne relatif à Tux1 dans le fichier /etc/shadow (grep)

Exercice 3

1. Dans votre répertoire courant, créez en une commande les fichiers suivants :(touch) Ita1, ita1, Ita2, ita2, ita, Ita, itaadarissa, istaadarissa
2. Créer le répertoire «dossierv1» dans votre répertoire courant, et déplacez les fichiers précédemment créés dans le répertoire «dossierv1». (mv i* dossierv1/)
3. Lister tous les fichiers se terminant par «a» ls -l *a
4. Lister tous les fichiers commençant par «ita» ls -l ita*
5. Créer le répertoire «d_copie1» dans votre répertoire courant
6. Copier les fichiers dont l'avant dernier caractère est un s ou a dans le répertoire / d_copie1 en une seule commande. (cp *[sa]?/d_copie1/)

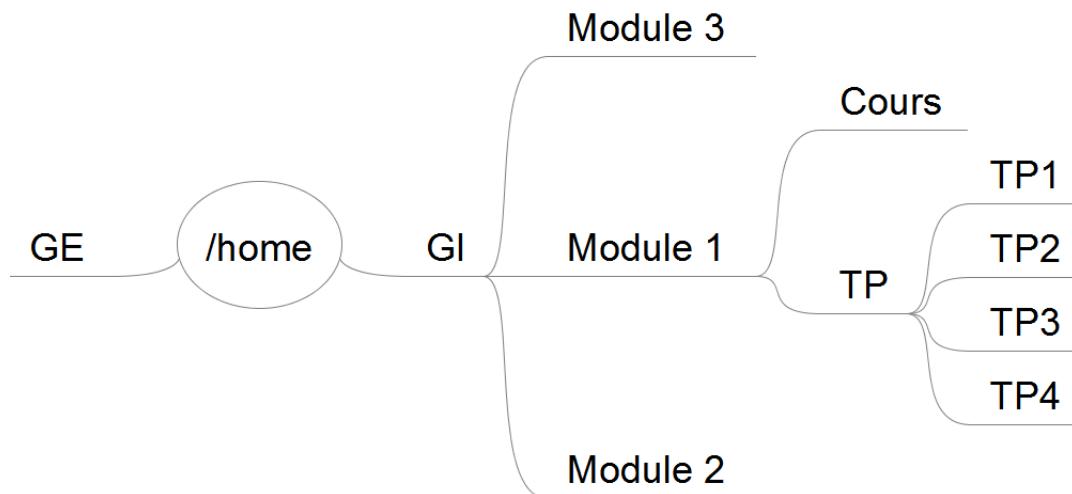
Exercice 4

On veut établir un fichier fichier1_droit dans le répertoire dossier_droit, Par défaut ce fichier est à 644 (rw-r-r-). En partant du répertoire courant, essayez d'accéder au répertoire dossier2_droit (commande cd), de faire un ls dans dossier_droit et de modifier le fichier2_droit avec un éditeur quelconque (vi par exemple).

1. Etablir le répertoire «dossier1_droit» avec les droits normal
2. Accéder à ce dossier
3. Etablir le fichier «fichier1_droit» avec les droits normal
4. Afficher le contenu de ce dossier
5. Modifier les droits du fichier pour qu'il soit modifiable par tout le monde
6. Tester d'ouvrir fichier par vi et ajouter vos nom et prénom à la fin
7. Afficher les droit du fichier et celle du dossier

Exercice 5

Soit la figure suivante :



1. Créer l'arborescence ci-dessus dans le dossier /home (mkdir)
2. Créer 2 utilisateurs TRI et TSGE avec mot de passe P@ssw0rd (passwd name_user)
3. Copier tous les fichiers du répertoire /dev qui ont l'extension .C vers le répertoire facture (en utilisant les chemins relatifs pour la source et pour la destination. (cp)
4. Renommer le répertoire Module3 par Examen (En utilisant un chemin relatif pour la source et un chemin absolu pour la destination) (mv)
5. Créer le répertoire Absence à l'intérieur du répertoire GE (mkdir) Créer un lien symbolique pour le répertoire TP1 dans /home et nommer le TP1_ls (ln)
6. Déplacer la branche Module2 dans le répertoire Examen en utilisant des chemins absolus (mv)
7. Donner les droits drwx- - - r - x pour le dossier GE (chmod)
8. On veut que tous les nouveaux répertoires créés auront par défaut le masque 750 (rwxr-x - - -) (umask)
9. Afficher les 13 dernières lignes du fichier /etc/passwd

Exercice 6

1. Ouvrez un éditeur de texte (gedit &) et créez un fichier texte.
2. Créez un lien symbolique et un lien physique dessus.
3. Vérifiez avec ls -li les numéros d'inode et le type de chaque fichier.
4. Ouvrez le fichier cible avec gedit et modifiez-le.
5. Ouvrez les fichiers liens. Est-ce que les contenus sont synchronisés ?
6. Fermez gedit.
7. (a) Supprimez le fichier cible (commande rm).
(b) A nouveau, essayez d'ouvrir les deux fichiers liens. Que se passe-t'il ?
8. Essayez de créer des liens physiques et symboliques sur des répertoires. Bilan ?

Exercice 7

Créer un répertoire essai-grep dans votre home directory. Dans ce répertoire créer les fichiers suivants : tomate, poire, pomme, cerise, Fraise, fraise, courgette, POMME3, afraise. Afficher les fichiers (sortie de la commande ls redirigée vers grep) avec les critères sur leur nom suivant :

- Critère 1 : Le nom doit être «Fraise» ou «fraise»
- Critère 2 : contient «se» est en fin de nom
- Critère 3 : «ai» est présent dans le nom
- Critère 4 : Nom contenant un chiffre numérique

Exercice 8

Créer un script qui demande à l'utilisateur de saisir une note et qui affiche un message en fonction de cette note :

- "**très bien**" si la note est entre 16 et 20 ;
- "**bien**" lorsqu'elle est entre 14 et 16 ;
- "**assez bien**" si la note est entre 12 et 14 ;
- "**moyen**" si la note est entre 10 et 12 ;
- "**insuffisant**" si la note est inférieur à 10.

Exercice 9

Reprenez l'exercice 1 et faites en sorte que le programme se répète tant que l'utilisateur n'a pas saisi une note négative ou 'q' (pour quitter). Le script doit calculer le nombre de notes de saisies et en faire la moyenne tout à la fin.

Exercice 10 Créer un script qui prend un nombre en saisie et l'élève à sa propre puissance. C'est un peu le même principe que la factorielle mais cette fois, l'usage de la boucle for est imposé.

Commande de base utilisable pour ce TP N°1

Gestion des utilisateurs

I. Crédation d'un utilisateur

La création d'un utilisateur se fait au moyen de la commande adduser [user]. Cette commande permet de donner un certain nombre d'informations au compte. Voici comme exemple la création d'un utilisateur appelé "test".

NB. pour initialiser le mot de passe de super_user (root) utiliser cette commande : sudo passwd puis entrer le mot de passe de votre root.

Consulter tous les utilisateurs du system : **cut -d ":" -f 1 /etc/passwd**

```
[root@localhost home]# adduser gil2
```

II. Changement du mot de passe

Pour changer le mot de passe d'un compte utilisateur, on utilise la commande passwd [user]. Seul le super-utilisateur peut changer le mot de passe d'un autre utilisateur. Si vous voulez changer votre propre mot de passe, vous devez utiliser la commande sans paramètre. Voici comme exemple le changement du mot de passe d'un utilisateur appelé "gil2".

```
[root@localhost home]# passwd gil2
```

III. Crédation d'un groupe

La création d'un groupe se fait au moyen de la commande addgroup [user]. Voici comme exemple la création d'un groupe appelé "gp_gil2".

```
[root@localhost home]# addgroup gp_gil2
```

IV. Modification d'un utilisateur

La modification d'un utilisateur se fait au moyen de la commande usermod. Cette commande permet, entre autre, de modifier le répertoire de l'utilisateur, les groupes auxquels appartient l'utilisateur, le nom d'utilisateur, le mot de passe, etc. Voici comme exemple l'ajout de l'utilisateur "gil2" au groupe "gp_gil2".

```
[root@localhost home]# usermod -aG gp_gil2 gil2
```

//dans cette commande on va supprimer gil2 depuis les anciens groupes et l'associer seulement à ce groupe.

```
[root@localhost home]# groups gil2
```

cette commande pour consulter la liste des groupes de l'utilisateur spécifier.

V. Suppression d'un utilisateur

La suppression d'un utilisateur se fait au moyen de la commande deluser [user]. Si vous voulez supprimer en même temps le répertoire de l'utilisateur, vous pouvez utiliser l'option --remove-home. Voici comme exemple la suppression de l'utilisateur "gil2".

```
[root@localhost home]# deluser --remove-home gil2
```

VI. Suppression d'un groupe

La suppression d'un groupe se fait au moyen de la commande delgroup [group]. Voici comme exemple la suppression du groupe "gp_gil2".

```
[root@localhost home]# delgroup gp_gil2
```

VII. Opérations sur les répertoires et les fichiers

Comme Linux possède un explorateur de fichiers, il est possible de faire toutes les opérations à partir de l'environnement graphique de manière intuitive. Voici quelques commandes pour réaliser les opérations à partir de la console.

- La commande pour lister le contenu d'un répertoire est :

```
[root@localhost home]# ls [option] [fichier ...]
```

Pour avoir la liste complète de toutes les options, tapez man ls ou info ls. Voici quelques options utiles :

- ✓ -F : affiche / derrière un répertoire, * derrière un fichier exécutable, @ derrière un lien symbolique, etc.
- ✓ -R : liste le contenu des sous-répertoires.
- ✓ -a : affiche les fichiers cachés.
- ✓ -l : affiche les informations détaillées de chaque fichiers.

Si vous ne donner aucun argument à la commande, elle affiche le contenu du répertoire courant.

- Afficher le nom du répertoire courant

La commande pour afficher le nom du répertoire courant est :

```
[root@localhost home]#pwd
```

Il s'agit de l'abréviation de "PrintWorking Directory". La commande va retourner le chemin absolu du répertoire courant.

- Changer de répertoire

La commande pour changer de répertoire est :

```
[root@localhost home]# cd [chemin]
```

[chemin] doit être un chemin de répertoire valide et peut être soit absolu, soit relatif. Si aucun chemin n'est spécifié, vous vous rendrez dans votre répertoire personnel.

- Créer un répertoire

La commande pour créer un répertoire est :

```
[root@localhost home]# mkdir [option] chemin
```

chemin peut être absolu ou relatif. S'il est relatif, le répertoire sera créé dans le répertoire courant. Pour obtenir la liste de toutes les options de la commande, utilisez les pages du manuel.

- Copier un répertoire

La commande pour copier un répertoire est :

```
[root@localhost home]#cp -R source destination
```

```
[root@localhost home]#cp -r source destination
```

source et destination peuvent être absous ou relatifs. S'ils sont relatifs, les répertoires à copier et de destination devront se trouver dans le répertoire courant.

- **Effacer un répertoire**

La commande pour effacer un répertoire est :

```
[root@localhost home]#rmdir chemin
```

chemin peut être absolu ou relatif. S'il est relatif, le répertoire à supprimer devra se trouver dans le répertoire courant. De plus le répertoire chemin doit être vide. Pour effacer un répertoire avec son contenu, utiliser la commande :

```
[root@localhost home]#rm -R chemin
```

- **Déplacer ou renommer d'un répertoire**

La commande pour déplacer ou renommer un répertoire est :

```
[root@localhost home]#mv source destination
```

- Afficher tout le contenu d'un fichier, la commande est :

```
[root@localhost home]#cat fichier
```

Le problème avec la commande cat, c'est que l'entièreté du contenu du fichier sera affiché. Vous n'aurez à l'écran que la fin des données. Pour effectuer un affichage page par page, vous pouvez utiliser la commande :

```
[root@localhost home]#more fichier
```

Vous pouvez également vouloir afficher seulement le début ou la fin du fichier. Dans ce cas, utilisez respectivement les commandes :

```
[root@localhost home]#head [option] fichier
```

```
[root@localhost home]#tail [option] fichier
```

Pour obtenir la liste des différentes options de ces commandes, référez-vous aux pages de manuel.
Voici une option bien utile :

-n nombre

affiche nombre lignes. Par défaut, la commande affiche les 10 premières ou dernières lignes.

- **Copier un fichier**

Pour copier un fichier, la commande est :

```
[root@localhost home]#cp [option] source destination
```

Pour obtenir la liste des différentes options de ces commandes, référez-vous aux pages de manuel.

- **Supprimer un fichier**

Pour supprimer un fichier, la commande est :

[root@localhost home]#rm [option] chemin

Pour obtenir la liste des différentes options de ces commandes, référez-vous aux pages de manuel.
Voici une option bien utile :

-R : si chemin est un répertoire, efface ce dernier ainsi que son contenu.

- **Déplacer ou renommer un fichier**

Pour déplacer ou renommer un fichier, la commande est :

[root@localhost home]#mv source destination

- **Rechercher un fichier**

Pour rechercher un fichier, la commande est :

[root@localhost home]#find répertoire-name nom_fichier -print

répertoire est le répertoire à partir duquel la recherche s'effectue et nom est le nom du fichier à retrouver.

La commande find est très puissante et possède de nombreuses options. Vous pouvez consulter les pages de manuels pour plus d'information.

- **Créer un lien physique ou symbolique**

Pour créer un lien vers un fichier, la commande est :

[root@localhost home]#ln [-s] source lien

Par défaut, ln crée un lien physique. Si l'option -s est spécifiée, c'est un lien symbolique qui sera créé.

- **Changer les droits sur un fichier**

Pour les droits sur un fichier, la commande est :

[root@localhost home]#chmod [options] droits fichier ...

Pour la liste des options, référez-vous aux pages du manuel.

droits peut être exprimé au format octal. Dans ce cas, le premier chiffre est à usage spécial et dans la plupart des cas sera égal à 0, le deuxième chiffre représentera les droits du propriétaire, le troisième chiffre les droits du groupe et le quatrième chiffre les droits des autres. Exemple pour donner les droits -rwxr-xr-x :

\$ chmod 0755 fichier

droits peut également être sous une forme symbolique. La syntaxe sera de droits sera : QOP avec Q représentant sur qui portera la permission (a pour tous, u pour l'utilisateur, g pour le groupe, o pour les autres), O représentant l'opérateur (+ pour activer la permission, - pour supprimer la permission) et P représentant la permission (r pour lecture, w pour écriture, x pour exécution). Exemple pour donner les droits -rwxr-xr-x :

[root@localhost home]#chmod u+rwx,go+rx fichier

Exemple pour supprimer le droit d'écriture à toutes personnes à part le propriétaire :

```
[root@localhost home]#chmod go-w fichier
```

- Changer le propriétaire d'un fichier

Cette opération ne peut être faite que par le super-utilisateur. Pour changer le propriétaire d'un fichier, la commande est :

```
[root@localhost home]#chown propr[:groupe] fichier
```

propr représente le nom du nouveau propriétaire et groupe son groupe. groupe doit être spécifié si vous voulez utiliser un autre groupe que celui par défaut.

- Changer le groupe d'un fichier

Pour changer le groupe d'un fichier, la commande est :

```
[root@localhost home]#chggrp groupe fichier
```

ou

```
[root@localhost home]#chown:groupe fichier
```

- Rechercher une chaîne de caractères dans un fichier

Pour rechercher une chaîne de caractères dans un fichier, la commande est :

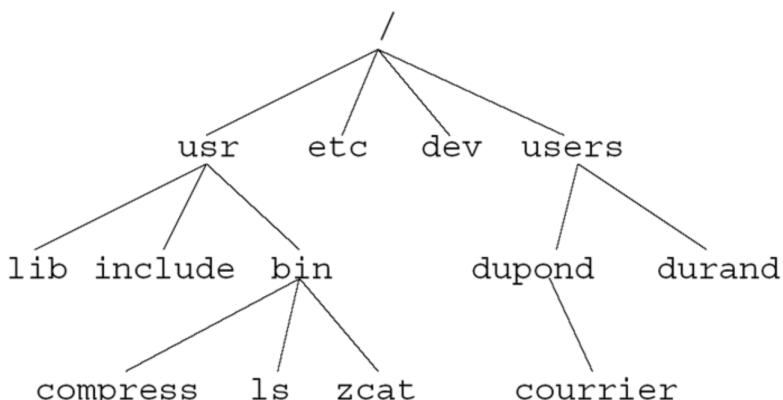
```
[root@localhost home]#grep texte fichiers
```

texte est la chaîne de caractère à retrouver et fichiers la liste des fichiers dans lesquels la recherche doit s'effectuer.

La commande grep est très puissante et possède de nombreuses options. Vous pouvez consulter les pages de manuels pour plus d'information.

Structure des fichiers

Les fichiers sont organisés en répertoires et sous-répertoires, formant une arborescence



Dans chaque répertoire, on trouve au moins deux fichiers :

1. . (point) permet de référencer le répertoire lui-même,
2. .. (pointpoint) permet de référencer le répertoire parent (répertoire du dessus).

Chemins absous et relatifs

Pour désigner un fichier quelconque, on peut utiliser soit un chemin absolu, soit un chemin relatif.

- Un chemin absolu spécifie la suite des répertoires à traverser en partant de la racine, séparés par des caractères / (et non \ comme sous DOS).

Par exemple, le chemin : /usr/bin/compress

- Un chemin relatif spécifie la suite des répertoires à traverser en partant du répertoire courant, séparés par des caractères /

Par exemple, si l'on travaille dans le répertoire dupond,

- ✓ On peut accéder au fichier durand en spécifiant le chemin : ../durand
- ✓ Du même endroit, on peut accéder au fichier compress via le chemin : ../../usr/bin/compress

Les commandes de base

- ✓ Pour toutes les commandes, il est possible d'obtenir de l'aide en tapant man suivi du nom de la commande.
- ✓ En tapant une commande suivie du paramètre --help, nous obtenons la liste des paramètres possibles.
- ✓ N'hésitez pas à recourir à la commande man ou au paramètre --help dès que vous avez besoin d'aide.

Gestion des répertoires

1. Se déplacer dans les répertoires:

cd [(Chemin absolu ou relatif) nom du répertoire] : change de répertoire

- Aller directement dans le répertoire de l'utilisateur: cd ~
- Revenir au répertoire précédent : cd -

2. Savoir où on est dans l'arborescence:

pwd : où suis-je dans l'arborescence ? Afficher le nom du répertoire courant

3. Lister les fichiers

•ls [opts] [rép]: liste le contenu d'un répertoire

4. Types de fichiers

Le premier caractère du résultat de la commande ls -l désigne le type du fichier:

- : désigne un fichier régulier(ordinaire) ce sont des fichiers dont les données sont stockés sous forme d'octets sur le disque dur.
- d : désigne un répertoire (dossier ou directory)
- l : désigne un lien symbolique (raccourci)
- c: Fichier spécial caractère
- b: Fichier spécial bloc
- p : pipe nommé

5. Créer un répertoire

mkdir [options] [nom répertoire] crée un répertoire

Options :

- -p : Crée les répertoires parents si nécessaire
- -m : Donne des droits d'accès spécifiques

6. Créer un fichier

touch file : crée un fichier vide file, si file n'existe pas, change la date de file si il existe

7. Supprimer un fichier

rm file: supprime le fichier file

8. Supprimer un répertoire

La commande rmdir (remove directory) permet de supprimer un répertoire vide.

rmdir [options] [nom répertoire]

Remarque: pour supprimer un répertoire non vide:

rm -r rep

9. Copier un fichier ou un répertoire

cp [options] source destination

Attention:

Gardez à l'esprit que si le fichier cible existe déjà, il sera purement et simplement écrasé, sans aucun message d'avertissement. Soyez donc très prudent en matière de copie de fichiers

cp -r source destination permet de copier le répertoire source

10. Renommer et/ou déplacer

La commande mv (move) remplit ces deux fonctions.

mv [options] source destination permet de renommer et/ou déplacer un fichier ou un répertoire

11. Les droits d'accès sur les fichiers

Sous Unix il existe trois familles d'utilisateurs pour chaque fichier :

- le propriétaire
- le groupe
- les autres

Et pour chacune de ses trois familles il existe trois types d'autorisation

r : lecture

w : écriture

x : exécution

Le premier champ du résultat de ls -l , former de 10 caractères, désigne le type et les droits

- le premier caractère désigne le type
- les trois suivants désignent les droits du propriétaire
- les trois suivants désignent les droits du groupe
- les trois derniers désignent les droits des utilisateurs

12.La commande ls

La commande *ls* permet d'afficher le contenu d'un répertoire.

- *ls -l* Afficher les informations de manière détaillée.
- *ls -a* Afficher les fichiers cachés.
- *ls -h* Afficher la taille des fichiers de façon lisible.
- *ls -r* Tri inversé.
- *ls -t* Trier les fichiers par date du plus récent au plus ancien.
- *ls -S* Trier par taille décroissante.
- *ls -la* Afficher tous les fichiers y compris les fichiers cachés.
- *ls -lhS* Afficher les informations des fichiers, avec des tailles lisibles le tout ordonné du plus grand au plus petit.

13.La commande cd

La navigation d'un répertoire à un autre s'effectue avec la commande *cd* suivie du nom du répertoire.

- *cd /* Permet de se retrouver à la racine du disque.
- *cd ~* ou *cd* Accéder directement au répertoire de l'utilisateur.
- *cd /var/www/* Aller dans le répertoire /var/www.
- *cd ..* Remonter dans le répertoire parent à partir de là où vous êtes.
- *cd -* Permet de revenir au répertoire précédent.

14.La commande pwd

Renvoyer le chemin absolu du répertoire courant ce qui est utile puisqu'en général le shell n'affiche que le nom du répertoire courant.

15.La commande cp

Pour créer une copie d'un fichier, on utilise la commande *cp*.

cp foo/bar.txt baz/ Copier le fichier bar.txt dans le répertoire baz.

cp -r foo/ baz/ Copier des répertoires entiers (note : si baz existe, la cible sera baz/foo/).

16.La commande mv

Déplacer un fichier est aussi simple que de le copier, pour cela il faut utiliser la commande *mv*. Cette commande permet aussi de renommer vos fichiers.

mv foo/bar.txt baz/ Déplacer le fichier bar.txt dans le répertoire baz.

`mv foo_bar.txt foo_baz.txt` Renommer le fichier `foo_bar.txt` en `foo_baz.txt`.

17. La commande rm

Pour effacer un fichier ou un répertoire on utilise la commande ***rm***.

`rm *.txt` Supprimer tous les fichiers ayant pour extension txt.

`rm foo.txt bar.txt` Supprimer les fichiers `foo.txt` et `bar.txt`.

`rm -rf baz/` Supprimer le répertoire `baz` et tout son contenu.

18.La commande mkdir

Pour créer un répertoire il suffit d'utiliser la commande ***mkdir***. Celle-ci vous permettra de créer un répertoire à l'emplacement où vous êtes ou l'emplacement précisé en argument de la commande.

`mkdir -v` Retourner des informations lors de la création d'un répertoire.

`mkdir -p` Cette option permet de créer une arborescence complète.

`mkdir foo` Créer le répertoire `foo`.

`mkdir -v foo /tmp/bar` Créer les répertoires `foo` et `/tmp/bar`.

`mkdir -p foo/bar/baz` Créer l'arborescence `foo/bar/baz`.

19.La commande chown

La commande ***chown*** permet de changer le propriétaire d'un fichier ou d'un répertoire.

`chownbob:admin foo.txt` Attribuer l'utilisateur `bob` et le groupe `admin` au fichier `foo.txt`.

20. La commande chmod

La commande ***chmod*** permet de changer les droits d'un fichier ou d'un répertoire. Pour exécuter cette commande vous devez être le propriétaire du fichier ou être logué en root.

`chmod u+w fichier` Ajouter les droits d'écriture au propriétaire (user, write)

`chmod g+r fichier` Ajouter les droits de lecture au groupe du fichier (group, read)

`chmod o-x fichier` Supprimer les droits d'exécution aux autres utilisateurs (other, execution)

`chmod a+rwx dossier` Ajouter les droits de lecture / écriture à tous (all)

`chmod -R a+rwx files` Ajouter les droits de lecture et d'exécution à tout ce que contient le répertoire dossier.

`chmod 764 dossier` Tous les droits pour le propriétaire (7xx), lecture et écriture pour le groupe (x6x) et lecture uniquement pour les autres (xx4).

chmod -R 755 dossier Donner au propriétaire tous les droits (7xx), alors que seuls les droits de lecture et d'accès seront donnés aux autres (x55). Grace à l'option -R ces droits seront appliqués à tous les fichiers et dossiers contenus dans ce répertoire.

Correspondances de représentation des droits

Droit	Valeur alphanumérique	Valeur octale
aucun droit	---	0
exécution seulement	--x	1
écriture seulement	-w-	2
écriture et exécution	-wx	3
lecture seulement	r--	4
lecture et exécution	r-x	5
lecture et écriture	rw-	6
tous les droits (lecture, écriture et exécution)	rwx	7

21.La commande grep

La commande *grep* permet de rechercher une chaîne de caractères ou un motif dans un fichier.

Quelques options :

- v: affiche les lignes ne contenant pas la chaîne.
- c: compte le nombre de lignes contenant la chaîne.
- n: retourne les lignes préfixées par leur numéro.
- x: ligne correspondant exactement à la chaîne.
- l: affiche le nom des fichiers qui contiennent la chaîne.

grep 'text' foo.txt Recherche l'occurrence 'text' dans le fichier foo.txt.

grep -nri 'foobar' /project Recherche toutes les occurrences de 'foobar' dans le répertoire /project.

grep -nri '(foo|bar|baz)' /project Recherche toutes les occurrences à 'foo', 'bar' et 'baz' dans le répertoire /project.

22.La commande find

La commande *find* permet de chercher des fichiers et éventuellement d'exécuter des commandes dessus.

Quelques options :

- name: Recherche d'un fichier par son nom
- iname: Même chose que -name mais insensible à la casse
- type: Recherche de fichier d'un certain type
- atime: Recherche par date de dernier accès
- mtime: Recherche par date de dernière modification
- user: Recherche de fichiers appartenant à l'utilisateur donné
- group: Recherche de fichiers appartenant au groupe donné
- size: Recherche par rapport à une taille de fichier.

- exec: Exécute la commande donnée aux fichiers trouvés.
- a: Opérateur ET
- o: Opérateur OU
- ! ou -not: Opérateur NOT

find myfile* -print Rechercher un fichier commençant par "myfile"

find -name *myfile*.txt -print Rechercher un fichier contenant "myfile" et ayant pour extention ".txt"

find /usr -type d -print Afficher tous les répertoires de /usr

find \$HOME \(-name '*.txt' -o -name '*.pdf' \) Afficher tous les fichiers .txt ou .pdf dans le répertoire home de l'utilisateur.

find \$HOME -name *.txt -atime +7 -execrm {} \; Supprimer tous les fichiers .txt qui n'ont pas été consultés depuis plus de 7 jours dans le répertoire home de l'utilisateur.

find \$HOME -name '*.txt' -size +4k -execls -l {} \; Afficher la taille de tous les fichiers de plus de 4 kilos

23. La commande Archives

Que ce soit pour compresser ou décompresser des fichiers ou des dossiers, cela se fait avec la commande *tar*.

Quelques options :

- c: créer
- t: tester / lister
- x: extraire
- v: description des fichiers désarchivés
- j: format de compression bzip2
- z: format de compression gzip

tar -cvf archive.tar fichier1 Création d'une archive nommée archive.tar contenant le fichier fichier1.

tar -cvf archive.tar fichier1 fichier2 Création d'une archive contenant deux fichiers fichier1 et fichier2.

tar -cvf archive.tar repertoire/ Création d'une archive à partir d'un répertoire.

tar -czvf archive.tar.gz repertoire/ Création d'une archive au format tar.gz.

tar -cjvf archive.tar.bz2 repertoire/ Création d'une archive au format tar.bz2.

tar -xzvf archive.tar.gz Extraction de l'archive tar.gz.

tar -xjvf archive.tar.bz2 Extraction de l'archive tar.bz2.

tar -tf mon_fichier.tar Liste tous les fichiers contenus dans une archive.

24.La commande pour Espace disque

du -sh dossier1 dossier2 connaitre l'espace disque utilisé des deux répertoires (*disk usage*).

du -hc --max-depth=1 afficher l'espace disque utilisé des fichiers et répertoires contenu dans un répertoire.

df -h afficher l'espace disque disponible (*disk free*).