



FORMATUX - Travaux Dirigés
GNU/Linux
Administration CentOS 6 (Version corrigée)

1.0, 03-08-2017

Table des matières

Préface	1
Crédits	1
Licence	1
Gestion des versions	2
1. Commandes de bases	3
1.1. ATELIER 1 : Affichage et identification	3
Exercice 1.1 : Identification	3
1.2. ATELIER 2 : A l'aide !	4
Exercices 2.1 : Utiliser le manuel	4
1.3. ATELIER 3 : Arborescence et fichiers	5
Exercice 3.1 : Création de répertoires	5
Exercice 3.2 : Gestion des fichiers	6
Exercice 3.3 : Gestion des répertoires	7
Exercice 3.4 : Suppression de fichiers et répertoires	7
1.4. ATELIER 4 : Recherches et filtres	8
Exercice 4.1 : Affichage et filtres	8
1.5. ATELIER 5 : tubes et redirections	9
Exercices	10
2. Gestion des utilisateurs	12
2.1. Gestion des utilisateurs et des groupes	12
Exercice 1.1 : Fichiers de configuration	12
Exercice 1.2 : Groupes et utilisateurs	12
Exercice 1.3 : Commentaires	13
2.2. ATELIER 2 : Gestion avancée des utilisateurs	14
Exercice 2.1 : Groupes secondaires	14
Exercice 2.2 : Mots de passe	15
Exercice 2.3 : Pérennité du compte et du mot de passe	15
Exercice 2.4 : Verrouillage du compte	16
Exercice 2.5 : Validation des changements	16
2.3. ATELIER 3 : Gestion avancée des comptes	16
Exercice 3.1 : Création par défaut	17
Exercice 3.2 : Message à la connexion	18
Exercice 3.3 : Ajout d'un utilisateur	18
3. Système de fichiers	20
3.1. ATELIER 1 : Ajout et préparation d'un disque	20
Exercice 1.1 : Ajouter un disque	20
Exercice 1.2 : Préparer le nouveau disque	20

Exercice 1.3 : Formater les nouvelles partitions	21
Exercice 1.4 : Monter les disques dans l'arborescence	21
3.2. ATELIER 2 : Les liens	22
Exercice 2.1 : Lien physique	22
Exercice 2.2 : Lien symbolique en absolu	24
Exercice 2.3 : Lien symbolique en relatif	25
3.3. ATELIER 4 : Droits particuliers	25
Exercice 4.1 : Création d'une boîte aux lettres	25
Exercice 4.2 : Droits d'endossement	26
3.4. ATELIER 5 : Sécuriser les fichiers	27
Exercice 5.1 : Evaluer ses droits sur un répertoire	27
Exercice 5.2 : Interdire la modification du contenu d'un répertoire	27
Exercice 5.3 : Interdire l'accès à un répertoire	28
Exercice 5.4 : Autoriser les droits en lecture seule sur un fichier	29
Exercice 5.5 : Droits d'exécution sur un fichier exécutable	30
Exercice 5.6 : Droits d'exécution sur un script	30
4. gestion des processus	32
4.1. ATELIER 1 : Gestion des processus	32
Exercice 1.1 : Identification des processus	32
Exercice 1.2 : Déconnexion immédiate	32
Exercice 1.3 : Arrière-plan / Avant-plan	33
5. Sauvegardes et restaurations	34
5.1. ATELIER 1 : La commande tar	34
Exercice 1.1 : Sauvegarde en mode absolu de /etc	34
Exercice 1.2 : Ajout d'un fichier	34
Exercice 1.3 : Sauvegarde en mode absolu de /home	35
Exercice 1.4 : Sauvegarde compressée en mode relatif	35
Exercice 1.5 : Vérification des sauvegardes	35
Exercice 1.6 : Restauration en mode absolu	36
Exercice 1.7 : Restauration en mode relatif	37
Exercice 1.8 : Restauration de la sauvegarde compressée	37
5.2. ATELIER 2 : La commande cpio	38
Exercice 2.1 : Sauvegarde en mode absolu	38
Exercice 2.2 : Sauvegarde en mode relatif avec « cpio »	39
Exercice 2.3 : Vérification des sauvegardes	39
Exercice 2.4 : Restauration en mode absolu	40
Exercice 2.5 : Restauration en mode relatif d'une sauvegarde en mode absolu	41
Exercice 2.6 : Restauration en mode relatif	41

Préface

GNU/Linux est un **système d'exploitation** libre fonctionnant sur la base d'un **noyau Linux**, également appelé **kernel Linux**.

Linux est une implémentation libre du système **UNIX** et respecte les spécifications **POSIX**.

GNU/Linux est généralement distribué dans un ensemble cohérent de logiciels, assemblés autour du noyau Linux et prêt à être installé. Cet ensemble porte le nom de “**Distribution**”.

- La plus ancienne des distributions est la distribution **Slackware**.
- Les plus connues et utilisées sont les distributions **Debian**, **RedHat** et **Arch**, et servent de base pour d'autres distributions comme **Ubuntu**, **CentOS**, **Fedora**, **Mageia** ou **Manjaro**.

Chaque distribution présente des particularités et peut être développée pour répondre à des besoins très précis :

- services d'infrastructure ;
- pare-feu ;
- serveur multimédia ;
- serveur de stockage ;
- etc.

La distribution présentée dans ces pages est la CentOS, qui est le pendant gratuit de la distribution RedHat. La distribution CentOS est particulièrement adaptée pour un usage sur des serveurs d'entreprises.

Crédits

Ce support de cours a été rédigé par les formateurs :

- Patrick Finet ;
- Antoine Le Morvan ;
- Xavier Sauvignon ;
- Nicolas Kovacs.

Licence

Formatux propose des supports de cours Linux libres de droits à destination des formateurs ou des personnes désireuses d'apprendre à administrer un système Linux en autodidacte.

Les supports de Formatux sont publiés sous licence Creative Commons-BY-SA et sous licence Art Libre. Vous êtes ainsi libre de copier, de diffuser et de transformer librement les œuvres dans le

respect des droits de l'auteur.

BY : Paternité. Vous devez citer le nom de l'auteur original.

SA : Partage des Conditions Initiales à l'Identique.

- Licence Creative Commons-BY-SA : <https://creativecommons.org/licenses/by-sa/3.0/fr/>
- Licence Art Libre : <http://artlibre.org/>

Les documents de Formatux et leurs sources sont librement téléchargeables sur framagit :

- <https://framagit.org/alemorvan/formatux.fr-support/>

Vous y trouverez la dernière version de ce document.

A partir des sources, vous pouvez générer votre support de formation personnalisé. Nous vous recommandons le logiciel AsciiDocFX téléchargeable ici : <http://asciidocfx.com/>

Gestion des versions

Table 1. Historique des versions du document

Version	Date	Observations
1.0	Août 2017	Version initiale.

Chapitre 1. Commandes de bases

1.1. ATELIER 1 : Affichage et identification

Objectifs

- ✓ Prendre en main un poste de travail,
- ✓ Se renseigner sur les utilisateurs connectés.

Exercice 1.1 : Identification

- Se connecter sur la console 1.
- Afficher les informations concernant le login courant.

```
[stagiaire]$ id
uid=1000(stagiaire) gid=100(users) groupes=100(users) ...
```

- Se connecter sur la console 2 avec le même utilisateur.
- Afficher les informations concernant le login courant.

```
[stagiaire]$ id
uid=1000(stagiaire) gid=100(users) groupes=100(users) ...
```

- Afficher les informations concernant l'utilisateur **patrick**.

```
[stagiaire]$ id patrick
uid=503(patrick) gid=501(GroupeP) groupes=501(GroupeP)
```

- D'autres utilisateurs sont-ils connectés sur le serveur ?

```
[stagiaire]$ who
stagiaire tty1 2016-01-04 13:05
stagiaire tty2 2016-01-04 13:10
```



Il n'y a que l'utilisateur stagiaire de connecté sur le serveur. Il est connecté sur le terminal 1 (**tty1**) Les terminaux physiques sont nommés **ttyX**, les terminaux virtuels (ceux de l'interface graphique) sont nommés (**pts/X**).

- Depuis quelle console êtes-vous connecté ?

```
[stagiaire]$ who am i
stagiaire tty2 2016-01-04 13:10
[stagiaire]$ whoami
stagiaire
```

► <https://www.youtube.com/watch?v=Ta7PXO3t3-0> (YouTube video)

1.2. ATELIER 2 : A l'aide !

Objectifs

- Rechercher dans la documentation,
- Se renseigner sur les utilisateurs connectés.

Exercices 2.1 : Utiliser le manuel

- Rechercher de l'aide sur la commande **passwd**

```
[stagiaire]$ whatis passwd
passwd (1) – Mettre à jour les marques d'authentification d'un utilisateur
passwd (5) – Fichier des mots de passe
```

```
[stagiaire]$ man passwd
```

- Rechercher des informations sur le formatage du fichier **passwd**

```
[stagiaire]$ man 5 passwd
```

- Quel est l'emplacement de l'**UID** dans le fichier **passwd** à l'aide du **man**.

```
[stagiaire]$ man 5 passwd
/UID
```

- Vérifier la date du jour.

```
[stagiaire]$ date
jeu. mars 14 15:15:25 CET 2015
```

- Effacer la console.

```
[stagiaire]$ clear
```

► <https://www.youtube.com/watch?v=NkwrnCtq7YE> (YouTube video)

1.3. ATELIER 3 : Arborescence et fichiers

Objectifs

- ✓ créer, supprimer, déplacer des fichiers ou des répertoires ;
- ✓ se déplacer dans l'arborescence.

Exercice 3.1 : Création de répertoires

- Afficher le répertoire courant.

```
[stagiaire]$ pwd  
/home/stagiaire
```

- Se déplacer de deux façons différentes sous le répertoire **/home**.

✓ chemin absolu :

```
[stagiaire]$ cd /home/
```

✓ chemin relatif :

```
[stagiaire]$ cd ..
```

- Vérifier que **/home** soit bien le nouveau répertoire courant.

```
[stagiaire]$ pwd  
/home
```

- Retourner dans le répertoire de connexion, et vérifier.

```
[stagiaire]$ cd  
[stagiaire]$ pwd  
/home/stagiaire
```

- Créer les répertoires suivants : `/home/stagiaire/tickets/` `/home/stagiaire/tickets/pierre/` `/home/stagiaire/tickets/jacques/`

```
[stagiaire]$ mkdir -p tickets/pierre tickets/jacques
[stagiaire]$ ls tickets/
jacques pierre
```

Exercice 3.2 : Gestion des fichiers

- Créer le fichier `/home/stagiaire/tickets/listing_en_cours`.

```
[stagiaire]$ touch tickets/listing_en_cours
```

- Copier ce fichier dans les répertoires `/home/stagiaire/tickets/pierre` et `/home/stagiaire/tickets/jacques`. Vérifier la taille de ces fichiers.

```
[stagiaire]$ cp tickets/listing_en_cours tickets/pierre/
[stagiaire]$ cp tickets/listing_en_cours tickets/jacques/
```

- ✓ Vérifier la copie en comparant les tailles :

```
[stagiaire]$ ls -lh tickets/listing_en_cours tickets/pierre/listing_en_cours
tickets/jacques/listing_en_cours
-rw-r--r-- 1 stagiaire users 0 [...] tickets/listing_en_cours
-rw-r--r-- 1 stagiaire users 0 [...] tickets/pierre/listing_en_cours
-rw-r--r-- 1 stagiaire users 0 [...] tickets/jacques/listing_en_cours
```



La taille des fichiers est identique, 0 octet (ils sont vides).

- Renommer le fichier `/home/stagiaire/tickets/jacques/listing_en_cours` en `listing_fini`.

```
[stagiaire]$ mv tickets/jacques/listing_en_cours tickets/jacques/listing_fini
```

- Déplacer et renommer le fichier `/home/stagiaire/listing_en_cours` en `/STAGE/commandes/archive_listing`.



Pour déplacer le fichier `listing_en_cours` du répertoire `/home/stagiaire/tickets` vers `/STAGE/commandes`, il faut d'abord créer ce dernier dossier :

```
[stagiaire]$ mkdir -p /STAGE/commandes
```

puis le déplacer :

```
[stagiaire]$ mv tickets/listing_en_cours /STAGE/commandes/archive_listing
```

Exercice 3.3 : Gestion des répertoires

- Copier le répertoire `/home/stagiaire/tickets/pierre/` et son contenu en le renommant `/home/stagiaire/tickets/sauvegarde/`.

```
[stagiaire]$ cp -r tickets/pierre/ tickets/sauvegarde
```

- Renommer le répertoire `/home/stagiaire/tickets/sauvegarde/` en `/home/stagiaire/tickets/archives/`.

```
[stagiaire]$ mv tickets/sauvegarde/ tickets/archives
```

- Copier le répertoire `/home/stagiaire/tickets/` dans le répertoire `/STAGE/commandes/`.

```
[stagiaire]$ cp -r tickets/ /STAGE/commandes/
```

Exercice 3.4 : Suppression de fichiers et répertoires

- Afficher le contenu des répertoires `/home/stagiaire/tickets/jacques/` et `/home/stagiaire/tickets/pierre/`.

```
[stagiaire]$ ls tickets/jacques/ tickets/pierre/
tickets/jacques/:
listing_fini

tickets/pierre/:
listing_en_cours
```

- Supprimer le répertoire `/home/stagiaire/tickets/jacques/` avec la commande `rmdir`.

```
[stagiaire]$ rmdir tickets/jacques/
rmdir : échec de suppression de « tickets/jacques/ » : Le dossier n'est pas vide
[root]# rm -f tickets/jacques/listing_fini
[root]# rmdir tickets/jacques/
```

- Supprimer le répertoire `/home/stagiaire/pierre/` en une seule commande.

```
[stagiaire]$ rm -rf tickets/pierre/
```

Vérifier les suppressions :



```
[stagiaire]$ ls -R tickets/
tickets/:
archives

tickets/archives:
listing_en_cours
```

► https://www.youtube.com/watch?v=QySrZv5_cis (YouTube video)

1.4. ATELIER 4 : Recherches et filtres

Objectifs

- rechercher un fichier ;
- rechercher du texte dans un fichier ;
- afficher un fichier, trier son contenu.

Exercice 4.1 : Affichage et filtres

- Copier dans le répertoire de connexion **/home/stagiaire** le fichier **/etc/passwd**.



Dorénavant, travailler sur cette copie.

```
[stagiaire]$ cp /etc/passwd ./
```

- Afficher les 7 premières lignes puis les 3 dernières.

```
[stagiaire]$ head -n 7 /home/stagiaire/passwd
[stagiaire]$ tail -n 3 /home/stagiaire/passwd
```

- Retrouvez la ligne contenant **alain**.

```
[stagiaire]$ grep "^alain" /home/stagiaire/passwd
alain:x:500:500::/home/GroupeA/alain:/bin/bash
```

Ou

```
[stagiaire]$ less /home/stagiaire/passwd
```

Puis

```
/alain
```

- Trier ce fichier par ordre d'`UID` croissant.

```
[stagiaire]$ sort -k3 -t: -n /home/stagiaire/passwd
```

- Combien y a-t-il d'utilisateurs créés sur le serveur ?

```
[stagiaire]$ wc -l /home/stagiaire/passwd
39 /home/stagiaire/passwd
```



Le fichier **passwd** contient 39 lignes, il y a donc 39 utilisateurs créés sur le serveur.

- Déplacer ce fichier dans le répertoire **/STAGE/commandes**.

```
[stagiaire]$ mv /home/stagiaire/passwd /STAGE/commandes
```

- Afficher les fichiers **passwd** présents dans le dossier **/STAGE** en précisant leur type.

```
[stagiaire]$ find /STAGE -name "passwd" -exec file {} \;
/STAGE/commandes/passwd: ASCII text
```

► <https://www.youtube.com/watch?v=0kiGdvjOrDA> (YouTube video)

1.5. ATELIER 5 : tubes et redirections

Objectifs

- utiliser un tube ;
- utiliser une redirection.

Exercices

- Créer un fichier `/home/stagiaire/suivi_admin`.

```
[stagiaire]$ touch /home/stagiaire/suivi_admin
```

- Se connecter sur le terminal 2 et suivre les modifications du fichier en direct.

Se connecter sur le terminal 2 avec **CTRL+SHIFT+ALT+F2** et afficher le fichier en temps réel :

```
[stagiaire]$ tail -f /home/stagiaire/suivi_admin
```



La suite de ce TP se fait sans éditeur de texte !

Retourner sous le terminal 1 et ajouter au fichier `suivi_admin` le texte **Voici les répertoires de /STAGE/commandes/gestion/ :**

- Retourner sur l'interface graphique avec **ALT+F1** et modifier le fichier :

```
[stagiaire]$ echo "Voici les répertoires de /STAGE/commandes/gestion/ :" > /home/stagiaire/suivi_admin
```

- Toujours dans `suivi_admin`, ajouter la liste des répertoires de `/STAGE/commandes/gestion/` en faisant apparaître les tailles avec l'indication Ko, Mo, Go ou To.

```
[stagiaire]$ find /STAGE/commandes/gestion/ -type d -exec ls -sdh '{}' \; >> /home/stagiaire/suivi_admin
```

- Vérifier le contenu du fichier en basculant sur le terminal 2.

Se connecter sur le terminal 2 avec **CTRL+SHIFT+ALT+F2**

- Retourner sous terminal 1 et ajouter au fichier `suivi_admin` le texte **Voici les personnes ayant un fichier listing_en_cours sous /STAGE/commandes/gestion/ :**

Retourner sur l'interface graphique avec **ALT+F1**.

```
[stagiaire]$ echo "Voici les personnes ayant un fichier listing_en_cours sous /STAGE/commandes/gestion/:" >> /home/stagiaire/suivi_admin
```

- Tapez la commande :

```
[stagiaire]$ find /STAGE/commandes/tickets -listing_en_cours >>
/home/stagiaire/suivi_admin 2>/home/stagiaire/erreur
```

- Basculer sur le terminal 2 et vérifier que la commande se soit bien exécutée.

Sur le terminal 2, rien n'a été modifié. En fait, la commande saisie comporte une erreur. Son affichage a donc été redirigé sur le canal d'erreur, le fichier **erreur**, et non **suivi_admin**.

- Corriger la commande pour remplir le fichier **suivi_admin**.

Il faut donc corriger la commande :

```
[stagiaire]$ find /STAGE/commandes/tickets -name listing_en_cours >>
/home/stagiaire/suivi_admin 2> /home/stagiaire/erreur
```

- Afficher parmi les 3 dernières lignes du fichier **suivi_admin** celles qui contiennent **pierre**.

```
[stagiaire]$ tail -n3 /home/stagiaire/suivi_admin | grep "pierre"
/STAGE/commandes/tickets/pierre/listing_en_cours
```

- Retourner sous le terminal 2 et se déconnecter.

taper **Ctrl+d** puis :

```
[stagiaire]$ exit
```

► <https://www.youtube.com/watch?v=m2WXIRYnWgU> (YouTube video)

Chapitre 2. Gestion des utilisateurs

2.1. Gestion des utilisateurs et des groupes

Objectifs

- créer un utilisateur ;
- créer un groupe.

Prérequis

- Se connecter sur la console 1 avec l'utilisateur root (mot de passe : mdparoot).

Exercice 1.1 : Fichiers de configuration

- Sauvegarder les fichiers de configuration des groupes et des utilisateurs en les copiant dans le répertoire /STAGE/utilisateurs.

```
[root]# mkdir /STAGE/utilisateurs
[root]# cp /etc/passwd /etc/shadow /etc/group /etc/gshadow /STAGE/utilisateurs/
```

Exercice 1.2 : Groupes et utilisateurs

- Créer les groupes et les utilisateurs suivants.

Groupe	GID	Utilisateurs
LINUX	1001	antoine, xavier
WINDOWS	1002	vincent, david
GroupeD	503	
GroupeV	504	
GroupeX	505	

Utilisateur	UID	REPERTOIRE
antoine	2001	/home/linux/antoine
xavier	2002	/home/linux/xavier
vincent	2003	/home/windows/vincent
david	2004	/home/windows/david

Créer dans un premier temps les 5 groupes en précisant leur GID :

```
[root]# groupadd -g 1001 LINUX
[root]# groupadd -g 1002 WINDOWS
[root]# groupadd -g 503 GroupeD
[root]# groupadd -g 504 GroupeV
[root]# groupadd -g 505 GroupeX
```

Créer ensuite les répertoires parents des répertoires de connexion des utilisateurs :

```
[root]# mkdir /home/{linux,windows}
```

Ajouter enfin les 4 utilisateurs en précisant leur UID, le GID de leur groupe principal, leur répertoire de connexion ainsi que leur shell :

```
[root]# useradd -u 2001 -g 1001 -d /home/linux/antoine antoine
[root]# useradd -u 2002 -g 1001 -d /home/linux/xavier xavier
[root]# useradd -u 2003 -g 1002 -d /home/windows/vincent vincent
[root]# useradd -u 2004 -g 1002 -d /home/windows/david david
```

Exercice 1.3 : Commentaires

- Modifier les commentaires des utilisateurs et visualiser les modifications.

Utilisateur	Commentaire
antoine	Antoine LM
xavier	Xavier S
vincent	Vincent B
david	David B

```
[root]# usermod -c "Antoine LM" antoine
[root]# usermod -c "Xavier S" xavier
[root]# usermod -c "Vincent B" vincent
[root]# usermod -c "David B" david
```

Visualiser les modifications dans le fichier `/etc/passwd` :

```
[root]# tail -n 4 /etc/passwd
antoine:x:2001:1001:Antoine LM:/homme/linux/antoine:/bin/bash
xavier:x:2002:1001:Xavier S:/homme/linux/xavier:/bin/bash
vincent:x:2003:1002:Vincent B:/homme/linux/vincent:/bin/bash david:x:2004:1002:David
B:/homme/linux/david:/bin/bash
```

2.2. ATELIER 2 : Gestion avancée des utilisateurs

Objectifs

- gérer les groupes secondaires ;
- appliquer une gestion des mots de passe à un utilisateur ;
- verrouiller un compte utilisateur.

Exercice 2.1 : Groupes secondaires

Inviter les utilisateurs suivant dans le groupe adéquat.

Utilisateur	Groupe
antoine	GroupeA
xavier	GroupeX
vincent	GroupeV
david	GroupeD

Ajouter le groupe secondaire à l'utilisateur :

```
[root]# usermod -aG GroupeA antoine
[root]# usermod -aG GroupeX xavier
[root]# usermod -aG GroupeV vincent
[root]# usermod -aG GroupeD david
```

Ou ajouter l'utilisateur au groupe secondaire :

```
[root]# gpasswd -a antoine GroupeA
[root]# gpasswd -a xavier GroupeX
[root]# gpasswd -a vincent GroupeV
[root]# gpasswd -a david GroupeD
```

Inviter ces utilisateurs dans le groupe users :

```
[root]# gpasswd -a antoine users
[root]# gpasswd -a xavier users
[root]# gpasswd -a vincent users
[root]# gpasswd -a david users
```

Rediriger les informations à propos des groupes de ces utilisateurs vers le fichier /STAGE/utilisateurs/modifications.

Rediriger le résultat de la commande id :

```
[root]# id antoine >> /STAGE/utilisateurs/modifications
[root]# id xavier >> /STAGE/utilisateurs/modifications
[root]# id vincent >> /STAGE/utilisateurs/modifications
[root]# id david >> /STAGE/utilisateurs/modifications
```

Exercice 2.2 : Mots de passe

- Définir les mots de passe des utilisateurs.

Utilisateur	MDP
antoine	tuxtux
xavier	tuxone
vincent	formatux
david	itstime

```
[root]# passwd antoine
[root]# passwd xavier
[root]# passwd vincent
[root]# passwd david
```

Exercice 2.3 : Pérennité du compte et du mot de passe

- Configurer les paramètres du mot de passe et du compte de l'utilisateur antoine.
 - Durée maximale du mot de passe : 60 jours
 - Durée minimale du mot de passe : 45 jours
 - Délai avant expiration du mot de passe : 5 jours
 - Inactivité du mot de passe : 10 jours
 - Durée de validité du compte : 365 jours

```
[root]# chage -m 45 -M 60 -I 10 -W 5 -E `date --date '1 year' +%Y-%m-%d` antoine
```

ou en mode interactif

```
[root]# chage antoine
```

Exercice 2.4 : Verrouillage du compte

- Verrouiller le compte david puis le compte de vincent avec deux commandes distinctes.

Verrouiller le compte utilisateur :

```
[root]# passwd -l david
```

et

```
[root]# usermod -L vincent
```

- L'utilisateur vincent devra modifier son mot de passe à la première connexion.

Demander un changement de mot de passe à la prochaine connexion :

```
[root]# chage -d 0 vincent
```

Exercice 2.5 : Validation des changements

Afficher les 4 dernières lignes du fichier /etc/shadow et visualiser les changements apportés par cet exercice.

```
[root]# tail -n 4 /etc/shadow
antoine:$6$. . . :16897:45:60:7:10:1726
xavier:$6$. . . :16897:0:99999:7:::
vincent:!!$6$. . . :0:0:99999:7:::
david:!!$6$. . . :16897:0:99999:7:::
```

2.3. ATELIER 3 : Gestion avancée des comptes

Objectifs

- modifier les paramètres systèmes de gestion des utilisateurs ;
- personnaliser les scripts de connexion.

Exercice 3.1 : Création par défaut

- Configurer les paramètres par défaut pour qu'à la création d'un utilisateur :
 - Le répertoire de connexion soit dans /home/utilisateurs
 - L'UID minimum soit 3000 et le GID minimum 3000
 - Le mot de passe ne soit valable que 60 jours
 - Les répertoires privé, travail et partage soient créés dans le répertoire de connexion

Dans un premier temps, vérifier la valeur de la variable HOME contenue dans le fichier /etc/default/useradd.

```
[root]# grep HOME /etc/default/useradd
HOME=/home
```

Sans utiliser un éditeur de texte, affecter la nouvelle valeur de HOME et vérifier.

```
[root]# useradd -D -b /home/utilisateurs
[root]# grep HOME /etc/default/useradd
HOME=/home/utilisateurs
```

- Modifier le fichier /etc/login.defs.

Modifier le fichier /etc/login.defs pour configurer les options d'UID, de GID et de mot de passe :

```
[root]# vim /etc/login.defs
UID_MIN      3000
GID_MIN      3000
PASS_MAX_DAYS 60
```

- Automatiser la création des dossiers privé, travail, partage lors de la création des comptes.

Créer les répertoires privé, travail et partage dans le répertoire /etc/skel :

```
[root]# mkdir /etc/skel/{privé,travail,partage}
```

- Rediriger les paramètres par défaut d'ajout d'utilisateurs vers le fichier /STAGE/utilisateurs/default.

Rediriger les paramètres par défaut de la commande useradd:

```
[root]# useradd -D > /STAGE/utilisateurs/default
```

Exercice 3.2 : Message à la connexion

Faire afficher le message suivant à la connexion du compte xavier : "Tout est fichier, sauf le café !"

- Solution 1 :

```
[root]# echo 'echo "Tout est fichier, sauf le café !"' >> /home/linux/xavier/.bashrc
```

- Solution 2 :

```
[root]# vim /home/linux/xavier/.bashrc
```

- Sur le terminal 3, connectez-vous en tant que xavier et vérifiez :

```
localhost login: xavier
Password:
Tout est fichier, sauf le café !
[xavier@localhost ~]$
```

Exercice 3.3 : Ajout d'un utilisateur

- Ajouter l'utilisateur tanguy.

```
[root]# useradd tanguy
```



Si vous n'avez pas pensé à créer le répertoire `/home/utilisateurs`, vous obtiendrez l'erreur suivante :

```
useradd : impossible de créer le répertoire /home/utilisateurs/tanguy
```

Vérifier que les modifications des exercices précédents ont bien été prises en compte.

```
[root]# id tanguy
uid=3000(tanguy) gid=3000(tanguy) groupes=3000(tanguy)
[root]# tail -n 1 /etc/passwd
tanguy:x:3000:3000::/home/utilisateurs/tanguy:/bin/bash
[root]# tail -n 1 /etc/shadow
tanguy:!!:16897:0:60:7:::
```

Chapitre 3. Système de fichiers

3.1. ATELIER 1 : Ajout et préparation d'un disque

Objectifs

- utiliser la console VirtualBox pour ajouter un disque dur à un serveur ;
- partitionner avec l'utilitaire cfdisk le nouveau disque dur ;
- formater les partitions avec l'utilitaire mkfs ;
- monter les partitions avec l'utilitaire mount.

Exercice 1.1 : Ajouter un disque

- Ajouter un disque de 10 Go à votre serveur virtuel.

Exercice 1.2 : Préparer le nouveau disque

Démarrer à nouveau votre machine virtuelle et connectez-vous avec l'utilisateur root.

- Visualiser les disques accessibles :

```
[root]# fdisk -l
```

- Créer 6 partitions: 4 de 2 Go et 2 de 1 Go.

```
[root]# cfdisk /dev/sdc
```

Créer une nouvelle partition :

Choisir l'option [Primaire] :

Donner une taille à la partition :

Répéter l'opération 3 fois :

Faire un premier lecteur logique :

Répéter l'opération 3 fois :

Penser à écrire la nouvelle table de partition :

Exercice 1.3 : Formater les nouvelles partitions

- Formater les nouvelles partitions en ext4.

```
[root]# mkfs -t ext4 /dev/sdc1
[root]# mkfs -t ext4 /dev/sdc2
[root]# mkfs -t ext4 /dev/sdc3
[root]# mkfs -t ext4 /dev/sdc5
[root]# mkfs -t ext4 /dev/sdc6
[root]# mkfs -t ext4 /dev/sdc7

[root]# ls -l /dev/sdc*
```

Exercice 1.4 : Monter les disques dans l'arborescence

- Monter les volumes sur /Disque[1-6].

Création des points de montage:

```
[root]# mkdir /Disque1 /Disque2 /Disque3 /Disque4 /Disque5 /Disque6
```

Montage des partitions:

```
[root]# mount /dev/sdc1 /Disque1
[root]# mount /dev/sdc2 /Disque2
[root]# mount /dev/sdc3 /Disque3
[root]# mount /dev/sdc5 /Disque4
[root]# mount /dev/sdc6 /Disque5
[root]# mount /dev/sdc7 /Disque6
```

Visualisation des systèmes de fichiers:

```
[root]# mount
...
/dev/sdc1 on /Disque1 type ext4 (rw)
...
```

- Faire en sorte qu'à chaque redémarrage, le disque ajouté (ses partitions) soit pris en compte.

```
[root]# vim /etc/fstab
/dev/sdc1    /Disque1    ext4    defaults    0 0
...
/dev/sdc7    /Disque6    ext4    defaults    0 0
```

- Rediriger le contenu du fichier de configuration des montages et la table associée dans le fichier « Resultat » que vous placerez sur la première partition du disque ajouté dans un répertoire « SuiviInstallation ».

```
[root]# cd /Disque1
[root]# mkdir SuiviInstallation
[root]# cat /etc/fstab /etc/mtab > SuiviInstallation/Resultat
```

3.2. ATELIER 2 : Les liens

Objectifs

- gérer les liens physiques et symboliques.

Pré-requis

- Créer le répertoire /Disque1/programmeTux.

```
[root]# mkdir /Disque1/programmeTux
```

- Changer le propriétaire de ce dossier pour stagiaire.

```
[root]# chown stagiaire /Disque1/programmeTux
```

Exercice 2.1 : Lien physique

- Se connecter sur une console avec l'utilisateur stagiaire.
- Créer l'arborescence src/version1 dans votre répertoire de connexion.

```
[stagiaire]$ mkdir -p src/version1
```

- Créer un fichier tux.exe dans src/version1. Il contiendra le texte « version 1 »

```
[stagiaire]$ echo "version 1" > src/version1/tux.exe
```

- Créer un lien physique tux-v1.exe dans le répertoire src vers le fichier src/version1/tux.exe.

```
[stagiaire]$ ln src/version1/tux.exe src/tux-v1.exe
```

- Que pouvons-nous remarquer à propos des propriétés du lien et de sa source ?

Les deux fichiers ont le même numéro d'inode.

```
[stagiaire]$ ls -li src/version1/tux.exe src/tux-v1.exe
263883 -rw-r--r-- 2 stagiaire users 0 fev 11 14:37 src/tux-v1.exe
263883 -rw-r--r-- 2 stagiaire users 0 fev 11 14:37 src/version1/tux.exe
```

- Détruire le fichier src/version1/tux.exe. Que se passe-t-il pour le lien src/tux-v1.exe ?

Le compteur revient à 1.

```
[stagiaire]$ rm -f src/version1/tux.exe
[stagiaire]$ ls -li src/tux-v1.exe
263883 -rw-r--r-- 1 stagiaire users 0 fev 11 14:37 src/tux-v1.exe
```

- Recréer tux.exe dans src/version1/.

```
[stagiaire]$ touch src/version1/tux.exe
```

- Le lien physique est-il toujours un lien vers src/version1/tux.exe ? Vérifier le contenu du fichier src/version1/tux.exe.

```
[stagiaire]$ ls -li src/version1/tux.exe src/tux-v1.exe
263883 -rw-r--r-- 1 stagiaire users 10 fev 11 14:37 src/tux-v1.exe
263884 -rw-r--r-- 1 stagiaire users 0 fev 11 14:38 src/version1/tux.exe
```

Non car les numéros d'inode sont différents et le compteur vaut 1.

```
[stagiaire]$ cat src/version1/tux.exe
[stagiaire]$ cat src/tux-v1.exe
version1
```

Le fichier source est maintenant vide.

- Créer un lien physique latest vers src/tux-v1.exe dans le répertoire « /Disque1/programmeTux ».

```
[stagiaire]$ ln src/tux-v1.exe /Disque1/programmeTux/latest
```

- Que se passe-t-il ? Pourquoi ?

Impossibilité de créer le lien physique car la source et le lien sont sur deux systèmes de fichiers différents.

Exercice 2.2 : Lien symbolique en absolu

- Copier le dossier src/version1 vers src/version2. Le fichier src/version2/tux.exe contiendra le texte « version 2 »

```
[stagiaire]$ cp -R src/version1 src/version2
[stagiaire]$ echo "version 2" > src/version2/tux.exe
```

- Créer un lien symbolique latest dans le répertoire « /Disque1/programmeTux/ » vers src/version2/ en utilisant un chemin absolu.

```
[stagiaire]$ ln -s /home/stagiaire/src/version2/ /Disque1/programmeTux/latest
```

- Afficher le contenu du fichier /Disque1/programmeTux/latest/tux.exe

```
[stagiaire]$ less /Disque1/programmeTux/latest/tux.exe
version 2
```

- Renommer le dossier src en sources.

```
[stagiaire]$ mv src sources
```

- Le lien fonctionne-t-il toujours ?

```
[stagiaire]$ less /Disque1/programmeTux/latest/tux.exe
/Disque1/programmeTux/latest/tux.exe: Aucun fichier ou dossier de ce type
[stagiaire]$ ll /Disque1/programmeTux/latest
lrwxrwxrwx 1 stagiaire users 29 fev 11 16:20 /Disque1/programmeTux/latest ->
/home/stagiaire/src/version2/
```

Non : le lien n'est plus actif.

- Pourquoi ?

Le chemin est absolu et la source a changé de place.

Exercice 2.3 : Lien symbolique en relatif

- Dans le répertoire sources, créer un lien symbolique latest.exe vers sources/version2/tux.exe en utilisant un chemin relatif.

```
[stagiaire]$ cd sources
[stagiaire]$ ln -s ./version2/tux.exe latest.exe
```

- Déplacer (ne pas copier) le répertoire sources et son contenu dans « /Disque1/programmeTux ».

```
[stagiaire]$ cd ..
[stagiaire]$ mv sources /Disque1/programmeTux/
```

- Le lien /Disque1/programmeTux/latest.exe fonctionne-t-il toujours ?

```
[stagiaire]$ ls -l /Disque1/programmeTux/sources/latest.exe
lrwxrwxrwx 1 stagiaire users 24 fev 11 16:20 /Disque1/programmeTux/sources/latest.exe
-> ./version2/tux.exe
```

Oui : le lien est actif.

3.3. ATELIER 4 : Droits particuliers

Objectifs

- positionner des droits particuliers ;
- positionner le suid.

Exercice 4.1 : Création d'une boîte aux lettres

- Créer l'arborescence /home/BAL/resultats/.

```
[root]# mkdir -p /home/BAL/resultats
```

- Faire en sorte que les utilisateurs puissent uniquement déposer des fichiers dans « resultats » en ne pouvant que passer dans BAL .

```
[root]# chmod 711 /home/BAL
[root]# cd /home/BAL
[root]# chmod 733 resultats
```

- Lister les droits de « BAL » ainsi que ceux de ses sous répertoires.

```
[root]# cd ..
[root]# ls -ld BAL
drwx--x--x 5 root root 1024 fev 11 08:21 BAL

[root]# cd BAL
[root]# ls -l
drwx-wx-wx 5 root root 1024 fev 11 08:21 resultats
```

Exercice 4.2 : Droits d'endossement

La commande « chfn » permet de formater le champ commentaire des comptes utilisateurs. Cette commande écrit dans le fichier « /etc/passwd ».

- Visualiser les droits de la commande et ceux du fichier.

```
[root]# ls -l /usr/bin/chfn
-rwx--x--x 1 root root 16464 oct 16 2007 /usr/bin/chfn

[root]# ls -l /etc/passwd
-rw-r--r-- 1 root root 1638 fev 5 13:23 /etc/passwd
```

- Faire en sorte que les utilisateurs puissent utiliser cette commande et ainsi changer le commentaire les concernant.

```
[root]# chmod u+s /usr/bin/chfn
```

- Listez à nouveau les droits de la commande et ceux du fichier.

```
[root]# ls -l /usr/bin/chfn
-rws--x--x 1 root root 16464 oct 16 2007 /usr/bin/chfn

[root]# ls -l /etc/passwd
-rw-r--r-- 1 root root 1638 fev 5 13:23 /etc/passwd
```

3.4. ATELIER 5 : Sécuriser les fichiers

Objectifs

- évaluer les droits sur un fichier ou un répertoire ;
- sécuriser l'accès à un fichier ou à un répertoire ;
- rendre un fichier exécutable.

Exercice 5.1 : Evaluer ses droits sur un répertoire

- Créer le répertoire « cours » dans votre répertoire de connexion.

```
[stagiaire]$ cd
[stagiaire]$ mkdir cours
```

- Observer les droits d'accès appliqués à ce répertoire :

```
[stagiaire]$ ls -lisad cours
drwxr-xr-x 5 stagiaire users 4096 fev 11 08:21 cours
```

- Quel utilisateur êtes-vous vis-à-vis de ce répertoire ?

Je suis le propriétaire

- De quels droits disposez-vous sur ce répertoire ?

Je dispose des droits de lecture, écriture et de passage dans le répertoire.

Exercice 5.2 : Interdire la modification du contenu d'un répertoire

- Modifier les droits du répertoire cours pour qu'ils deviennent dr-xr-xr-x.

```
[stagiaire]$ chmod u-w cours
```

- Créer un fichier linux.txt dans ce répertoire.

```
[stagiaire]$ touch cours/linux.txt
```

- Que se passe-t-il ?

cp: ne peut créer le fichier régulier 'cours/linux.txt' permission non accordée.

- Créer un sous répertoire « windows » dans le répertoire cours.

```
[stagiaire]$ mkdir cours/windows
```

- Que se passe-t-il ?

mkdir : ne peut créer le répertoire 'windows' : permission non accordée.

- Rétablir les droits drwxr-xr-x du répertoire cours. Créer un fichier linux.txt dans cours puis créer le répertoire cours/windows.

```
[stagiaire]$ chmod u+w cours
[stagiaire]$ touch cours/linux.txt
[stagiaire]$ mkdir cours/windows
```

- Remettre les droits dr-xr-xr-x sur le répertoire cours. Tenter de détruire le fichier linux.txt et le sous-répertoire windows.

```
[stagiaire]$ chmod u-w cours
[stagiaire]$ rm -f cours/linux.txt
rm: ne peut enlever 'cours/linux.txt' permission non accordée.

[stagiaire]$ rm -Rf cours/windows
rm: ne peut détruire le répertoire 'cours/windows' : permission non accordée.
```

- Copier le fichier cours/linux.txt dans le répertoire cours/windows en le renommant srv2k8.txt.

```
[stagiaire]$ cp cours/linux.txt cours/windows/srv2k8.txt
```

- Commenter le résultat.

Les droits du répertoire cours permettent le passage, et ceux sur windows la copie du fichier.

Exercice 5.3 : Interdire l'accès à un répertoire

- Modifier les droits du répertoire cours pour qu'ils deviennent drw-r-xr-x.

```
[stagiaire]$ chmod 655 cours
```

- Se positionner dans cours.

```
[stagiaire]$ cd cours
```

- Que se passe-t-il ?

```
bash : cd : cours : permission non accordée
```

Il manque le droit x qui permet le positionnement.

- Afficher le contenu du fichier cours/linux.txt.

```
[stagiaire]$ less cours/linux.txt
less: cours/linux.txt : permission non accordée
```

- Afficher ou tenter d'accéder au sous répertoire windows.

```
[stagiaire]$ cd cours/windows
```

- Que se passe-t-il ?

```
bash : cd : cours/windows : permission non accordée
```

Exercice 5.4 : Autoriser les droits en lecture seule sur un fichier

- Mettre les droits afin de pouvoir se positionner dans le répertoire cours et donner les droits de lecture seule au fichier « linux.txt ».

```
[stagiaire]$ chmod 755 cours
[stagiaire]$ cd cours
[stagiaire]$ chmod u-w linux.txt
```

- Afficher le contenu de ce fichier.

```
[stagiaire]$ less linux.txt
```

- Tenter de modifier son contenu (à l'aide d'un éditeur de texte ou par la commande « cat >>linux.txt »).

```
[stagiaire]$ vim linux.txt
[stagiaire]$ cat >>linux.txt
```

- Tenter de détruire le fichier « linux.txt ».

```
[stagiaire]$ rm -f linux.txt
```

- Que se passe-t-il ? Pourquoi ?

La destruction du fichier est possible car le répertoire où il est situé possède le droit « w » pour « stagiaire ».

Exercice 5.5 : Droits d'exécution sur un fichier exécutable

- Copier le fichier « /bin/ls » dans votre répertoire de travail. Renommer le fichier en « ls1 ».

```
[stagiaire]$ cd
[stagiaire]$ cp /bin/ls ls1
```

- Quel est son type ?

```
[stagiaire]$ file ls1
```

- Exécuter ce fichier par la commande « ./ls1 ».

```
[stagiaire]$ ./ls1
```

- Enlever les droits d'exécution à ce fichier.

```
[stagiaire]$ chmod a-x ls1
```

- Tenter de le lancer à nouveau « ./ls1 ». Que se passe-t-il ?

```
[stagiaire]$ ./ls1
bash: ./ls1: permission non accordée
```

Exercice 5.6 : Droits d'exécution sur un script

- Créer un script nommé « Prog » dans ce script vous écrirez ceci :

```
#!/bin/bash
clear
echo "essai de script Shell"
```

- Vérifier les droits du fichier « Prog ».

```
[stagiaire]$ ls -l Prog
```

- Afficher le type de ce fichier.

```
[stagiaire]$ file Prog
```

- Modifier ses droits pour qu'il soit exécutable par le propriétaire.

```
[stagiaire]$ chmod u+x Prog
```

- Exécuter ce fichier « ./Prog » et vérifier le résultat.

```
[stagiaire]$ ./Prog
```

Affichage à l'écran du message « essai de script Shell ».

Chapitre 4. gestion des processus

4.1. ATELIER 1 : Gestion des processus

Objectifs

- identifier et gérer les processus.

Exercice 1.1 : Identification des processus

- Lister les processus de votre utilisateur et rediriger le résultat vers le fichier « /STAGE/processus/stagiaire ».

```
[stagiaire]$ ps
[stagiaire]$ ps > /STAGE/processus/stagiaire
```

- Lister l'ensemble des processus du système et rediriger le résultat vers le fichier « /STAGE/processus/systeme ».

```
[stagiaire]$ ps -ef
[stagiaire]$ ps -ef > /STAGE/processus/systeme
```

- Identifier le processus de connexion et rediriger la ligne vers le fichier « /STAGE/processus/connexion ».

```
[stagiaire]$ ps -ef | grep "login" ## Noter le PID (ex: 4530)
[stagiaire]$ ps -ef | grep "login" > /STAGE/processus/connexion
```

- Identifier le processus parent du processus ci-dessus.

```
root    4530    1  0 15:10 ?        00:00:00  login -- stagiaire
```

Le processus parent est le processus init (pid 1)

Exercice 1.2 : Déconnexion immédiate

- Arrêter de façon brutale votre processus appelé « bash ».

```
[stagiaire] ps | grep bash
13858 tty1      00:00:00    bash
[stagiaire] kill -9 13858
localhost login:
```

Exercice 1.3 : Arrière-plan / Avant-plan

- Lancer en tâche de fond la commande sleep avec une durée de 1000 secondes. Récupérer son exécution au premier plan.

```
[stagiaire]$ sleep 1000 &
[1] 1664

[stagiaire]$ fg 1
```

- Comment faire pour envoyer de nouveau ce processus à l'arrière-plan ?

```
[CTRL]+[Z]
[1]+  Stopped          sleep 1000

[stagiaire]$ bg 1
[1]+  sleep 1000 &
```

Visualiser les différents travaux en cours :

```
[stagiaire]$ jobs
[1]+  Running          sleep 1000 &
```

Chapitre 5. Sauvegardes et restaurations

5.1. ATELIER 1 : La commande tar

Objectifs

- effectuer une sauvegarde avec l'utilitaire tar ;
- restaurer la sauvegarde.

Prérequis

- maîtriser le concept de chemin absolu et de chemin relatif.

Exercice 1.1 : Sauvegarde en mode absolu de /etc

- Réaliser une sauvegarde en mode absolu de « /etc ». Nommer cette sauvegarde de façon à connaître l'utilitaire utilisé. Elle sera placée dans le répertoire « /Disque1/sauvegardes ».

Vérifier l'existence de l'arborescence « /Disque1/sauvegardes », sinon la créer.



« /Disque1 » est une partition créée durant le TP « Ajout d'un disque ».

Afin de respecter la convention de nommage vu en cours, nous noterons « P » quand la sauvegarde est effectuée en mode absolu.

```
[root]# mkdir -p /Disque1/sauvegardes
```

Puis créer la sauvegarde en mode absolu :

```
[root]# tar cvfP /Disque1/sauvegardes/etc.P.`date +%j`.tar /etc
```

ou

```
[root]# cd /Disque1/sauvegardes  
[root]# tar cvfP etc.P.`date +%j`.tar /etc
```

Exercice 1.2 : Ajout d'un fichier

- Ajouter le fichier « /usr/bin/passwd » à la sauvegarde précédente.

La sauvegarde précédente ayant été faite en mode absolu, on ajoutera le fichier en mode absolu également :

```
[root]# tar rvfP /Disque1/sauvegardes/etc.P.`date +%j`.tar /usr/bin/passwd
```

ou

```
[root]# cd /Disque1/sauvegardes
[root]# tar rvfP etc.P.`date +%j`.tar /usr/bin/passwd
```

Exercice 1.3 : Sauvegarde en mode absolu de /home

- Réaliser une sauvegarde en mode absolu de /home. Elle portera un nom explicite. Cette sauvegarde sera placée dans le répertoire « /Disque1/sauvegardes ».

```
[root]# tar cvfP /Disque1/sauvegardes/home.P.`date +%j`.tar /home
```

ou

```
[root]# cd /Disque1/sauvegardes
[root]# tar cvfP home.P.`date +%j`.tar /home
```

Exercice 1.4 : Sauvegarde compressée en mode relatif

- Réaliser une sauvegarde compressée en mode relatif de « /etc ». Pour la compression, vous avez le choix de la méthode. Nommer cette sauvegarde de façon à connaître l'utilitaire utilisé. Cette sauvegarde sera placée dans le répertoire « /Disque1/sauvegardes ».

La sauvegarde est à réaliser en mode relatif et compressée (plus besoin de la clé « P »).

- 1ère méthode avec gzip

```
[root]# tar cvfz /Disque1/sauvegardes/etc.`date +%j`.tar.gz /etc
```

- 2ème méthode avec bzip2

```
[root]# tar cvfj /Disque1/sauvegardes/etc.`date +%j`.tar.bz2 /etc
```

Exercice 1.5 : Vérification des sauvegardes

- Lister le contenu des trois sauvegardes réalisées.

Mettre le résultat du listage dans « /Disque1/sauvegardes/ResultatDesSauvegardes ».

Chaque partie du fichier sera séparée par un commentaire.

```
[root]# cd /Disque1/sauvegardes
[root]# echo "Lecture de la sauvegarde etc.P.`date +%j`.tar" >ResultatDesSauvegardes
[root]# echo "-----" >>ResultatDesSauvegardes
[root]# tar tvfP etc.P.`date +%j`.tar >> ResultatDesSauvegardes
[root]# echo >> ResultatDesSauvegardes
[root]# echo "Lecture de la sauvegarde home.P.`date +%j`.tar" >>ResultatDesSauvegardes
[root]# echo "-----" >>ResultatDesSauvegardes
[root]# tar tvfP home.P.`date +%j`.tar >> ResultatDesSauvegardes
[root]# echo >> ResultatDesSauvegardes

[root]# echo "Lecture de la sauvegarde etc.`date +%j`.tar.gz" >>ResultatDesSauvegardes
[root]# echo "-----" >>ResultatDesSauvegardes
[root]# tar tvfz etc.`date +%j`.tar.gz >> ResultatDesSauvegardes
[root]# echo >> ResultatDesSauvegardes
```

ou

```
[root]# tar tvfj etc.`date +%j`.tar.bz2 >> ResultatDesSauvegardes
[root]# echo >> ResultatDesSauvegardes
```

Exercice 1.6 : Restauration en mode absolu

- Restaurer /etc en mode absolu.



Souvenez-vous que le fichier « /usr/bin/passwd » a été ajouté à cette sauvegarde. Pour ne restaurer que le répertoire /etc, il faut le préciser dans la commande.

```
[root]# tar xvfP /Disque1/sauvegardes/etc.P.`date +%j`.tar /etc
```

- Placer la commande utilisée dans le fichier « /Disque1/sauvegardes/CmdSauvegardes ». Utiliser un commentaire pour spécifier cette commande.

```
[root]# echo "Restauration de /etc en mode absolu" > CmdSauvegardes
[root]# echo "-----" >> CmdSauvegardes
```



À l'aide de la flèche haute du clavier, je recherche ma commande de sauvegarde précédente :

```
[root]# echo "tar xvfP /Disque1/sauvegardes/etc.P.`date +%j`.tar /etc"
>>CmdSauvegardes
```

Exercice 1.7 : Restauration en mode relatif

- Restaurer /home/GroupeA en mode relatif.

La sauvegarde du répertoire /home a été réalisée en mode absolu. Pour la restaurer en mode relatif, il suffit d'omettre la clé « P » lors de la restauration.



Attention, connaître le répertoire courant est important lors d'une restauration en relatif.

De plus, comme précédemment, il faut restaurer seulement une partie de la sauvegarde « /home/GroupeA ».

Se placer dans un répertoire neutre (ici « /tmp ») pour que « home/GroupeA » soit restauré. Notez bien, que c'est effectivement « home/GroupeA » qui sera restauré et non « /home/GroupeA » car la restauration est en mode relatif !!

```
[root]# cd /tmp
```

Notez que le « / » devant « home/GroupeA » est nécessaire car c'est ce que contient effectivement la sauvegarde, qui pour rappel a été réalisée en mode absolu.

```
[root]# tar xvf /Disque1/sauvegardes/home.P.`date +%j`.tar /home/GroupeA
```

- Placer la commande utilisée dans le fichier « /Disque1/sauvegardes/CmdSauvegardes ». Vous utiliserez un commentaire pour spécifier cette commande.

```
[root]# echo >> CmdSauvegardes
[root]# echo "Restauration de /home/GroupeA en mode relatif" >> CmdSauvegardes
[root]# echo "-----" >> CmdSauvegardes
[root]# echo "tar xvf /Disque1/sauvegardes/home.P.`date +%j`.tar /home/GroupeA" >>
CmdSauvegardes
```

Exercice 1.8 : Restauration de la sauvegarde compressée

- Restaurer la sauvegarde compressée.

La sauvegarde compressée ayant été réalisée en mode relatif, nous devons la restaurer dans ce mode. Il est donc impératif de bien se positionner au préalable. Ensuite, il faut penser à utiliser la bonne clé de décompression.

```
[root]# cd /tmp
[root]# tar xvfz /Disque1/sauvegardes/etc.`date +%j`.tar.gz
```

ou

```
[root]# tar xvfj /Disque1/sauvegardes/etc.`date +%j`.tar.bz2
```

- Placer la commande utilisée dans le fichier « /Disque1/sauvegardes/CmdSauvegardes ». Vous utiliserez un commentaire pour spécifier cette commande.

```
[root]# echo >> CmdSauvegardes
[root]# echo "Restauration de la sauvegarde compressée" >> CmdSauvegardes
[root]# echo "-----" >> CmdSauvegardes
[root]# echo "tar xvfz /Disque1/sauvegardes/etc.`date +%j`.tar.gz" >> CmdSauvegardes
```

ou

```
[root]# echo "tar xvfj /Disque1/sauvegardes/etc.`date +%j`.tar.bz2" >> CmdSauvegardes
```

5.2. ATELIER 2 : La commande cpio

Objectifs

- effectuer une sauvegarde avec l'utilitaire cpio ;
- restaurer la sauvegarde.

Prérequis

- maîtriser le concept de chemin absolu et de chemin relatif.

Exercice 2.1 : Sauvegarde en mode absolu

- Réaliser une sauvegarde de « /etc » en mode absolu. Nommer cette sauvegarde de façon à connaître l'utilitaire utilisé. Cette sauvegarde sera placée dans le répertoire « /Disque1/sauvegardes ».

Pour réaliser une sauvegarde en mode absolu, il faut lister le répertoire à sauvegarder en mode absolu. Afin de respecter la convention de nommage vu en cours, noter « A » quand la sauvegarde est effectuée en mode absolu.

```
[root]# find /etc | cpio -ov >/Disque1/sauvegardes/etc.A.`date +%j`.cpio
```

ou

```
[root]# find /etc | cpio -ovF /Disque1/sauvegardes/etc.A.`date +%j`.cpio
```

/etc est listé en absolu du fait de la présence du « / »

Exercice 2.2 : Sauvegarde en mode relatif avec « cpio »

- Réaliser une sauvegarde en mode relatif de /home/GroupeA.

Elle portera un nom explicite. Cette sauvegarde sera placée dans le répertoire «/Disque1/sauvegardes».

Pour réaliser une sauvegarde en mode relatif, il faut lister le répertoire à sauvegarder en mode relatif et donc il faut se trouver à l'endroit où se trouve le répertoire à sauvegarder.

```
[root]# cd /
[root]# find home/GroupeA | cpio -ov >/Disque1/sauvegardes/HomeGroupeA.`date +%j`.cpio
```

ou

```
[root]# find home/GroupeA | cpio -ovF /Disque1/sauvegardes/HomeGroupeA.`date +%j`.cpio
```

home/GroupeA est listé en relative car son chemin ne commence pas par un “/”.

Exercice 2.3 : Vérification des sauvegardes

- Lister le contenu des deux sauvegardes réalisées et mettre le résultat du listage dans « /Disque1/sauvegardes/ResultatDesSauvegardes ».

```
[root]# cd /Disque1/sauvegardes
[root]# echo >> ResultatDesSauvegardes
[root]# echo "Lecture de la sauvegarde etc.A.`date +%j`.cpio" >>ResultatDesSauvegardes
[root]# echo "-----" >>ResultatDesSauvegardes
[root]# cpio -tv <etc.A.`date +%j`.cpio >> ResultatDesSauvegardes
```

ou

```
[root]# cpio -tvF etc.A.`date +%j`.cpio >> ResultatDesSauvegardes

[root]# echo >> ResultatDesSauvegardes
[root]# echo "Lecture de la sauvegarde HomeGroupeA.`date +%j`.cpio"
>>ResultatDesSauvegardes
[root]# echo "-----" >>ResultatDesSauvegardes
[root]# cpio -tv <HomeGroupeA.`date +%j`.cpio >>ResultatDesSauvegardes
```

ou

```
[root]# cpio -tvF HomeGroupeA.`date +%j`.cpio >>ResultatDesSauvegardes
```

Exercice 2.4 : Restauration en mode absolu

- Restaurer « /etc » en mode absolu.



La sauvegarde ayant été réalisée en mode absolu, elle sera restaurée par défaut en mode absolu.

L'utilisation de `-u` permet d'écraser tous les fichiers sans demander le remplacement des fichiers récents par de plus anciens.

```
[root]# cpio -iuv <etc.A.`date +%j`.cpio
```

ou

```
[root]# cpio -iuvF etc.A.`date +%j`.cpio
```

- Placer la commande utilisée dans le fichier « /Disque1/sauvegardes/CmdSauvegardes ». Vous utiliserez un commentaire pour spécifier cette commande.

```
[root]# echo >> CmdSauvegardes
[root]# echo "Restauration de la sauvegarde etc.A.`date +%j`.cpio" >>CmdSauvegardes
[root]# echo "-----" >> CmdSauvegardes
[root]# echo " cpio -iuv <etc.A.`date +%j`.cpio " >> CmdSauvegardes
```

ou

```
[root]# echo "cpio -iuvF etc.A.`date +%j`.cpio " >> CmdSauvegardes
```

Exercice 2.5 : Restauration en mode relatif d'une sauvegarde en mode absolu

- Restaurer « /etc » en mode relatif.

Pour restaurer en mode relatif une sauvegarde ayant été réalisée en mode absolu, il faut préciser l'option « --no-absolute-filenames ».

En général quand une sauvegarde est restaurée en mode relatif, c'est pour la vérifier et donc la restaurer dans un répertoire autre que celui naturellement prévu.

```
[root]# cd /tmp
[root]# cpio --no-absolute-filenames -iv </Disque1/sauvegardes/etc.A.`date +%j`.cpio
```

ou

```
[root]# cpio --no-absolute-filenames -ivF /Disque1/sauvegardes/etc.A.`date +%j`.cpio
```

(Notez bien que le nom du fichier de sauvegarde doit se trouver directement après l'option « F », c'est pourquoi l'option « --no-absolute-filenames » est placé en premier).

- Placer la commande utilisée dans le fichier « /Disque1/sauvegardes/CmdSauvegardes ».

Vous utiliserez un commentaire pour spécifier cette commande.

```
[root]# cd /Disque1/sauvegardes
[root]# echo >> CmdSauvegardes
[root]# echo "Restauration en relatif de etc.A.`date +%j`.cpio">> CmdSauvegardes
[root]# echo "-----">> CmdSauvegardes
[root]# echo "cd /tmp" >> CmdSauvegardes
[root]# echo "cpio --no-absolute-filenames -iv </Disque1/sauvegardes/etc.A.`date +%j`.cpio" >> CmdSauvegardes
```

ou

```
[root]# echo "cpio --no-absolute-filenames -ivF /Disque1/sauvegardes/etc.A.`date +%j`.cpio" >> CmdSauvegardes
```

Exercice 2.6 : Restauration en mode relatif

- Restaurer « /home/GroupeA » en mode relatif.

```
[root]# cd /tmp
[root]# cpio -idv </Disque1/sauvegardes/HomeGroupeA.`date +%j`.cpio
```

ou

```
[root]# cpio -idvF /Disque1/sauvegardes/HomeGroupeA.`date +%j`.cpio
```

- Placer la commande utilisée dans le fichier « /Disque1/sauvegardes/CmdSauvegardes ». Vous utiliserez un commentaire pour spécifier cette commande.

```
[root]# cd /Disque1/sauvegardes
[root]# echo >> CmdSauvegardes
[root]# echo "Restauration en relatif de HomeGroupeA.`date +%j`.cpio" >>CmdSauvegardes
[root]# echo "-----" >> CmdSauvegardes
[root]# echo "cpio -idv </Disque1/sauvegardes/HomeGroupeA.`date +%j`.cpio"
>>CmdSauvegardes
```

ou

```
[root]# echo "cpio -idvF /Disque1/sauvegardes/HomeGroupeA.`date +%j`.cpio"
>>CmdSauvegardes
```