



Linux

Introduction

➤ Les origines

A l'origine Linux est un noyau UNIX créé en 1991 dans le cadre d'un projet initialisé et coordonné par **Linus TORVALDS**.

Par la suite Linux a désigné les distributions construites autour de ce noyau.

➤ Linux sera le noyau qui manque au système d'exploitation GNU.

➤ Quelques distributions :



Introduction

- **Linux reprend les fonctionnalités d'Unix sans reprendre de code propriétaire**
- **Le développement de Linux s'est fait avec "la même philosophie" que le projet GNU**
- **GNU est un système d'exploitation libre créé en 1983 par Richard Stallman, maintenu par le projet GNU. Il reprend les concepts et le fonctionnement d'UNIX**

Introduction

- **La GPL (*General Public License*) est une license qui spécifie les conditions de distribution de tous les logiciels GNU.**

Introduction

- **Logiciel Libre** (selon <http://www.gnu.org>) ?
L'expression “Logiciel libre” fait référence à la liberté et non pas au prix
 - ✓ **La liberté d'exécuter le programme, pour tous les usages (liberté 0).**
 - ✓ **La liberté d'étudier le fonctionnement du programme, et de l'adapter à vos besoins (liberté 1). Pour ceci l'accès au code source est une condition requise.**

Introduction

➤ **Logiciel Libre (selon <http://www.gnu.org>) ?**

- ✓ **La liberté de redistribuer des copies, donc d'aider votre voisin, (liberté 2).**
- ✓ **La liberté d'améliorer le programme et de publier vos améliorations, pour en faire profiter toute la communauté (liberté 3).**
Pour ceci l'accès au code source est une condition requise.

Introduction

- **Un programme est un logiciel libre si les utilisateurs ont toutes ces libertés. Ainsi, vous êtes libre de redistribuer des copies, avec ou sans modification.**
- **La seule condition est que les sources de ces logiciels doivent être disponibles gratuitement pour quiconque en fait la demande.**
- **Linux a donc été développé selon les termes d'une licence GPL**

Introduction

➤ L'Open source:

L'expression Open Source, un logiciel Open Source est un programme informatique dont le code source est distribué sous une licence permettant à quiconque de lire, modifier ou redistribuer ce logiciel.

Le Shell

Présentation

Un shell est un terminal permettant de taper des commandes pour exécuter des tâches, tel que :

- Se déplacer dans le système de fichiers (répertoire)
- Modifier des fichiers
- Télécharger des fichiers
- Configurer le système
- etc.

Le Shell

Les différents shells Unix

Bash

- Le plus répandu

Bourne Shell

- Tout premier shell Unix

Tcsh

- Second shell de référence
- Le plus moderne

- C shell

- Plus proche de C

- Korn Shell

- AIX & HP/UX

- Z Shell

- Le plus innovant
 - Synthèse de tout les autres

Le Shell

Accéder au Shell

Démarrage en mode texte

- Shell accessible directement après authentification

Avec environnement de bureau

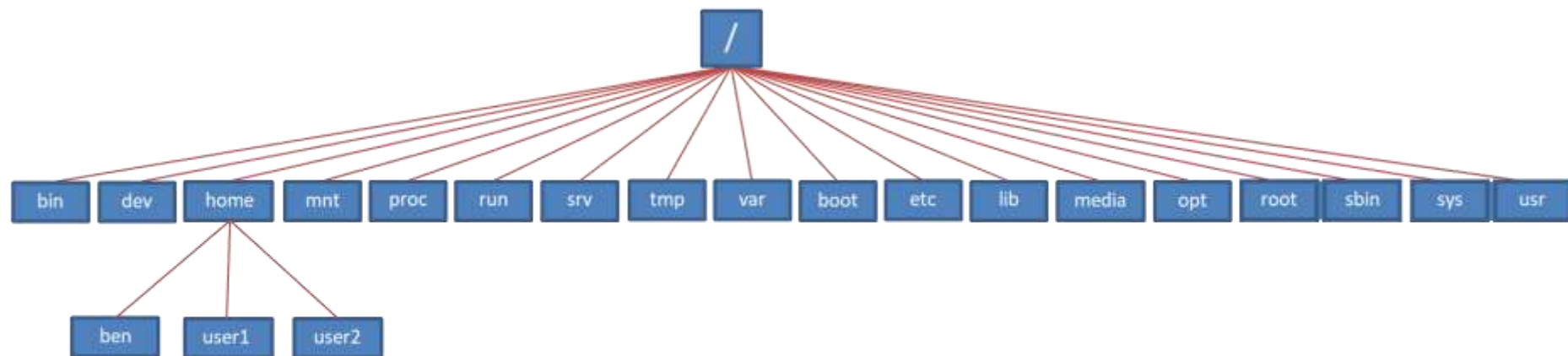
- Basculer sur une autre console (Ctrl + Alt + Fx)
- Ou utiliser un émulateur de terminal
 - Konsole, Gnome Terminal, xterm
 - Terminator

Le Système de fichiers

Le contenu d'un système de fichiers Linux

Il n'existe pas de norme d'organisation du système de fichiers, mais un standard est à peu près suivi par les différentes distributions de Linux.

Architecture du système:



/ est la racine du système il est représenté dans l'invite de commande par /
Pour savoir dans quel répertoire vous vous trouvez : **pwd (print working directory)**

10.30.10.100 - PuTTY

```
root@debian:/#  
root@debian:/# pwd  
/  
root@debian:/# ls  
bin      home      lib32     media     root      sys       vmlinuz  
boot     initrd.img  lib64     mnt       run       tmp       vmlinuz.old  
dev      initrd.img.old  libx32    opt       sbin      usr  
etc      lib        lost+found  proc      srv       var  
root@debian:/#
```

Le Système de fichiers

/bin	Contient les fichiers exécutables essentiels pour tous les utilisateurs
/boot	Les fichiers nécessaires au gestionnaire de démarrage
/dev	Contient les fichiers périphériques du système (dev = device)
/etc	Contient les fichiers de configuration du système et programmes
/home	Contient tous les répertoires personnels des utilisateurs
/lib	Contient les bibliothèques du système et les modules du noyau
/media	Périphériques amovible (montage)
/root	Contient le répertoire de base de l'administrateur (ROOT)
/sbin	Contient les fichiers exécutables pour l'administration du système donc surtout utilisé par root.
/tmp	Contient les fichiers temporaires créés par certains programmes
/usr	Contient les programmes, les librairies et les fichiers accessibles pour l'utilisateur
/var	Contient les données variables liées à la machine (log, backup, Mail, . . .)

Le Système de fichiers

Comment se repérer dans le système de fichiers

- La ligne de commande donne des informations :

Utilisateur courant Répertoire courant

```
root@fredon:/home/paul# |
```

Nom de la machine # : Superutilisateur
\$: Utilisateur normal

- Autre exemple

Sous répertoire du home de paul
« /home/paul/Documents/software/ »

```
paul@fredon:~/Documents/software$ |
```

~ : Répertoire home de paul (home/paul)

Le Système de fichiers

- La commande « **pwd** » permet de savoir quel est le répertoire courant
- La commande « **ls** » permet de lister les fichiers contenus dans un répertoire
- La commande « **cd** » permet de changer de répertoire
- Les symboles suivants ont une signification particulière :

◆ « **.** » : Le point désigne le répertoire courant

- Exemple : Exécuter un script depuis le répertoire courant

```
root@fredon:/home/paul# ./script.sh
```

◆ « **..** » : Les 2 points désignent le répertoire parent

- Exemple : Se déplacer dans le répertoire parent

```
root@fredon:/home/paul/Docs# cd ..
```

◆ « **~** » : Désigne le répertoire home de l'utilisateur courant

```
root@fredon:/home/paul/Docs# cd ~
```

Le Système de fichiers

Chemin relatif et absolu

- Il existe 2 méthodes pour spécifier un chemin dans le système de fichiers
 - ◆ Chemin relatif : Dépend du répertoire courant
 - ◆ Chemin absolu : Débute à la racine du système (« / »)

Exemples relatif

```
root@fredon:/home/paul# cd Documents
```

```
root@fredon:/home/jean# cd Documents
root@fredon:/home/jean/Documents/# |
```

```
root@fredon:/etc/apache# cd ../
root@fredon:/etc/# |
```

```
paul@fredon:~ # cd Docs
paul@fredon:~/Docs# |
```

absolu

```
root@fredon:/home/paul# cd /home/paul/Documents
```

```
root@fredon:/home/jean# cd /home/paul/Documents
root@fredon:/home/paul/Documents/# |
```

```
root@fredon:/etc/apache# cd /etc
root@fredon:/etc/# |
```

```
root@fredon:/etc# cd /home/paul/Docs
root@fredon :/home/paul/Docs # |
```

- Attention aux chemins relatifs à l'intérieur d'un script
 - ◆ Le script peut-être exécuté depuis n'importe où
 - ◆ Le répertoire courant est donc différent à chaque fois

Le Système de fichiers

Exercices

- Dans quel répertoire je suis situé ?

```
paul@fredon:~/Documents/rep1$
```

- Par quel chemin relatif équivalent peut-on remplacer celui-ci ?

```
paul@fredon:/etc$ cd /etc/ppp/peers/
```

- Que m'indiquera le résultat de la commande suivante?

```
mathilde@fredon:~/Documents/pub$ pwd
```

Le Système de fichiers

Exercices

- Dans quel répertoire je suis situé ?

```
paul@fredon:~/Documents/rep1$
```

◆ Réponse : Dans le répertoire « /home/paul/Documents/rep1 »

- Par quel chemin relatif équivalent peut-on remplacer celui-ci ?

```
paul@fredon:/etc$ cd /etc/ppp/peers/
```

◆ Réponse : «cd ppp/peers »

- Que m'indiquera le résultat de la commande suivante

```
?mathilde@fredon:~/Documents/pub$ pwd
```

◆ Réponse : « /home/mathilde/Documents/pub »

Le Système de fichiers

Les commandes

- Chaque commande Unix a une page de manuel qui la décrit. Pour afficher la page de manuel d'une commande, il suffit d'utiliser la commande suivante :

man nom_commande

- Par exemple, pour afficher l'aide sur la commande **cp**, il suffit de taper :

man cp

- Lorsqu'une page de man est affichée, il est possible de faire défiler son texte à l'aide des touches du curseur. Pour quitter l'aide, il suffit d'appuyer sur la touche **q**.

Le Système de fichiers

Les commandes

- **ls [options] [répertoire]**: lister le contenu d'un répertoire. Elle dispose d'un grand nombre d'options :
 - [répertoire] est le nom du répertoire que l'on désire lister. Si ce paramètre est absent, ls affichera tous les fichiers du répertoire courant. Les principales options sont
 - [Options] :
 - l, qui permet d'afficher des informations étendues (notamment les propriétaires, les groupes, les droits, la taille et éventuellement les liens),
 - a, qui permet d'afficher tous les fichiers, y compris les fichiers cachés
 - larth, pour voir l'intégralité du contenu du dossier et une foultitude d'informations
- **cd [chemin]** : changer de répertoire courant. Sa syntaxe est très simple :
 - [chemin] est un chemin de répertoire Unix valide. Ce chemin est constitué des noms des répertoires et sous-répertoires successifs, séparés par des barres obliques « / ».

Le Système de fichiers

Les commandes

- **mkdir [répertoire]** : créer un répertoire : (-p pour créer plusieurs répertoires)
Exemple : créer un répertoire TSSR , et à l'intérieur un répertoire STAGIAIRES
réponse: mkdir -p TSSR/STAGIAIRES
- **touch [fichier]** : crée un fichier vide
- **rmdir [répertoire]** : supprimer un répertoire , il faut qu'il soit vide (c'est-à-dire qu'il ne contienne ni fichier, ni répertoire).
- **rm [fichier]** pour supprimer un fichier
- **rm -r [répertoire]** pour supprimer un répertoire et son contenu (r pour récursif)
- **rm -ri [répertoire]** demande une confirmation avant la suppression

Le Système de fichiers

Les commandes

- La commande « cp » copie des fichiers source vers une ou plusieurs destinations
- Syntaxe :

```
cp [OPTION]... SOURCE... Destination
```

- Exemples

◆ Copie le fichier « tri.txt » vers « /home/paul »

```
cp tri.txt /home/paul
```

◆ Copie tous les fichiers du répertoire « /home/jean » vers « /home/paul »

```
cp /home/jean/* /home/paul
```

◆ Copie le rep « /home/jean » tout entier vers « /home/paul »

```
cp -r /home/jean/ /home/paul
```

◆ Copie en conservant les droits et l'appartenance (-a)

```
cp -a /home/jean/rapport.odt /home/paul
```


Le Système de fichiers

Les commandes

- Copier un fichier revient à créer un nouveau fichier
 - ◆ Identique à l'original
 - ◆ C'est donc l'utilisateur qui a effectué la copie qui devient le propriétaire
 - ◆ A moins d'utiliser l'option `-a`
 - ◆ Exemple :

Je suis connecté en tant que root

```
root@fredon:/home/jean# ls -l
total 0
lrwxrwxrwx 1 jean jean 26 2008-08-21 18:04 Examples -> /usr/share/example-content
-rw-r--r-- 1 jean jean 0 2008-08-21 18:14 rapport.odt
root@fredon:/home/jean# cp rapport.odt /home/paul/
root@fredon:/home/jean# ls -l /home/paul/
total 4
-rw-r--r-- 1 root root 0 2008-08-21 18:14 rapport.odt
drwxr-xr-x 2 paul paul 4096 2008-08-21 17:59 rep1
drwxr-xr-x 2 paul paul 4096 2008-08-21 18:00 rep2
drwxr-xr-x 2 paul paul 4096 2008-08-15 14:41 vidéos
root@fredon:/home/jean#
```

Je liste le répertoire de Jean

Ce fichier appartient à Jean

Je le copie chez Paul

La copie m'appartient (root)

Le Système de fichiers

- **mv [source] [destination]**, Déplacer les fichiers et les répertoires
[source] est le nom du fichier source
[destination] est le nom du répertoire destination.
- Notez que mv est une commande très puissante, puisqu'elle permet également de renommer des fichiers et des répertoires.
- Pour renommer un fichier ou un répertoire, il suffit d'indiquer le nouveau nom de ce fichier ou de ce répertoire à la place de destination.

Exemple:

On est dans «/home», Renommer le fichier tri.txt (qui se trouve dans /home) en tdi.txt

Réponse:

- mv tri.txt tdi.txt

Le Système de fichiers

Les commandes

- **cat** : affiche l'intégralité du contenu d'un fichier
- **touch [fichier]** : crée un fichier vide
- **head** : affiche le début du fichier (10 premières lignes)
- **tail** : affiche le fin du fichier (10 dernières lignes)(beaucoup utiliser pour surveiller les LOG avec option -f)
- **Find,locate** Trouver un fichier dans l'arborescence

Le Système de fichiers

Les commandes

find est une commande puissante sur linux, il permet de chercher des fichiers et des dossiers en utilisant différents critères : le nom, la taille, le type, propriétaire.

1-Rechercher les fichiers par nom dans le répertoire actuel :

```
# find . -name document.txt
```

2-Rechercher les fichiers par nom dans un répertoire spécifié :

```
# find /var/log -name *.log
```

3-Rechercher un répertoire par son nom :

```
# find /home -type d -name document
```

4-Rechercher les fichier vides :

```
# find / -type f -empty
```

6-Rechercher les fichiers cachés :

```
# find /var -type f -name ".*"
```

Le Système de fichiers

Les commandes

Locate

Contrairement à **find**, la commande **locate** ne cherche pas les fichiers au sein d'une arborescence mais dans une base de données.

Cette base de donnée stocke les informations des noms des fichiers existants et leurs arborescences.

```
root@serLinux:~# locate document.txt  
/home/admin/labo/document.txt
```

L'inconvénient de la commande **locate** est que les informations des fichiers récents que vous venez de créer ne sont pas stocké dans la base immédiatement. c'est pour cela qu'il faut mettre à jour la base de locate avec la commande **updatedb** :

```
root@serLinux:~# updatedb
```

Note : Le système se charge de mettre à jour cette base de données chaque jours.

Le Système de fichiers

Lien Symbolique & Physique

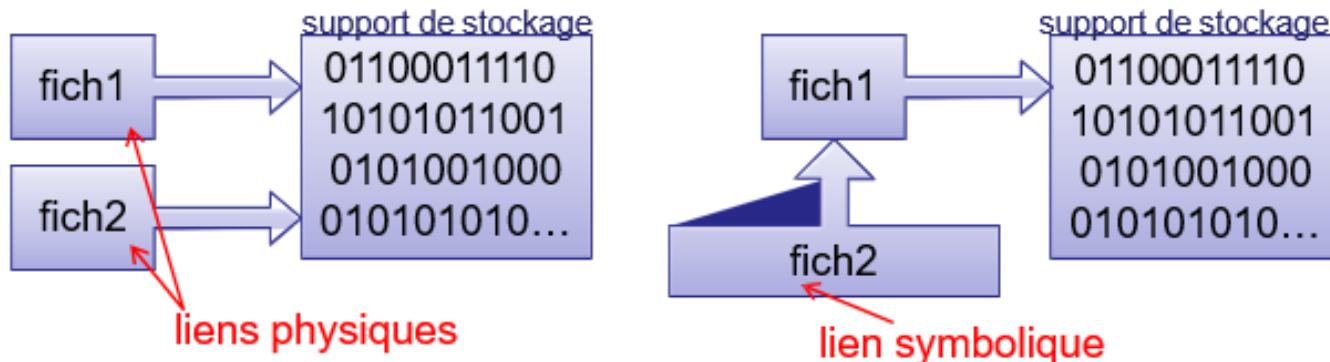
- Un lien est un type spécial de fichier qui fait référence à un autre fichier
- Axe central du fonctionnement de Linux, le lien permet :
 - ◆ De créer des raccourcis vers des fichiers existants
 - La compatibilité des logiciels entre les distributions Linux est assurée par les liens
 - ◆ D'éviter de stocker plusieurs fois le même fichier dans des répertoires différents

Syntaxe de la commande: `ln [option] [destination] [nom-lien]`

Option : `-s` □ symbolique

sans option □ physique

- Un petit dessin :



Le Système de fichiers

Lien Symbolique & Physique

- Le lien symbolique est une référence vers un fichier cible
 - ◆ Lorsque le fichier cible est effacé, le lien est rompu
 - ◆ Lorsque le lien est effacé, le fichier cible n'est pas effacé
- Exemple :

```
root@fredon:~/Documents$ ls -l
total 8
lrwxrwxrwx 1 root root 29 2008-08-25 14:23 ip -> /proc/sys/net/ipv4/ip_forward
drwxr-xr-x 3 root root 4096 2008-06-02 14:20 software
drwxr-xr-x 3 root root 4096 2008-07-29 15:54 vmware-tools
```

Nom du lien

Indique que c'est un lien

Emplacement du vrai fichier

- La commande « ln » avec l'option « -s » est utilisée pour créer un lien symbolique

```
root@fredon:~/Documents$ ln -s /proc/sys/net/ipv4/ip_forward ip
```

Cible (Target)

Nom du lien (link name)

- Afficher les informations complémentaires sur les liens : ls -l nom-lien**

Le Système de fichiers

Lien Symbolique & Physique

- Un lien physique est associé à un emplacement sur le support de stockage
 - ◆ 2 liens peuvent être associés au même « inode »
 - ◆ Deux liens physiques sont considérés comme 2 fichiers indépendants
 - Même si leur contenu est au même emplacement sur le support
 - ◆ Le lien physique est vu comme un fichier régulier
- Créer un lien physique avec la commande « ln » :

```
root@fredon:~/Documents$ ln /home/paul/Documents/rapport2007-2008.doc rap0708
root@fredon:~/Documents/essais$ ls -il
total 176
470930 -rw-r--r-- 2 root root 84091 2008-08-25 14:48 rap0708
470930 -rw-r--r-- 2 root root 84091 2008-08-25 14:48 rapport-annee2007_2008.doc
```

nom fichier

cible

L'« inode » est identique.
Il s'agit bien de liens
physiques

Nombre de liens vers cet inode.
C'est un indice permettant de
supposer qu'il s'agit d'un lien

Le Système de fichiers

Les Inodes

- Un « inode » est une structure de données concernant un fichier
 - ◆ Contient des informations sur :
 - Les droits, le propriétaire et le groupe
 - Le périphérique qui le contient
 - Des données relatives au système de fichiers et à l'emplacement du fichier sur le support de stockage
- A chaque fichier, correspond un « inode »
 - ◆ Il est unique pour le périphérique de stockage qui contient le fichier
- Pour connaître l'inode d'un fichier, la commande « ls » avec l'option « -i ».

```
root@fredon:~/Documents/essais$ ls -i /root/Documents
total 12
 8246 drwxr-xr-x 2 root rout 4096 2008-08-25 15:08 essais
467165 drwxr-xr-x 3 root rout 4096 2008-06-02 14:20 software
475969 drwxr-xr-x 3 root rout 4096 2008-07-29 15:54 vmware-tools
```

inode

Editeur de texte

- Le système Linux repose sur de nombreux fichiers texte modifiable avec un simple éditeur de texte.
- Les 2 éditeurs historiques du monde Unix :
 - ✓ **vi** ou **vim** : (créé par Billy Joy) pour les petites modifications. En cas de gros développements il est préférable d'utiliser...
 - ✓ **nano** : Prise en main facile pour débutant

Apt-get

- **Apt-get est la couche qui apporte une certaine intelligence et une grande facilité d'utilisation au système de gestion des packages Debian. Avec apt-get, on définit les sources des packages dans un fichier de configuration et il gère l'installation et le retrait des packages en tenant compte des dépendances ainsi que le téléchargement des packages s'ils sont sur une source réseau.**

Apt-get

Les commandes à savoir sont les suivantes :

apt-get update

Met à jour la liste des packages disponibles (pour les sources réseau ou fichier, il doit aller voir si elles ont été mises à jour)

apt-get upgrade

Met à jour tous les packages déjà installés à la dernière version disponibles dans les sources

apt-get dist-upgrade

Idem que la commande précédente, mais cette commande est optimisée pour les migrations vers une version supérieure de Debian ou beta

Apt-get

Les commandes à savoir sont les suivantes :

apt-get install package1 package2

Installe les packages package1 et package2 et tous les packages dont ils dépendent :

apt-get remove package1

Désinstalle le package package1 sans effacer ses fichiers de configuration :

apt-get remove --purge package1

Idem que la commande précédente mais ses fichiers de configuration sont supprimés :

apt-get clean

Efface du disque dur les packages téléchargés pour être installés

apt-cache search mot

La commande search recherche une chaîne de caractères dans la liste des paquets connus

La Sécurité – Les utilisateurs

- **Linux est un système multiutilisateurs. Il doit donc permettre l'administration des comptes.**
- **Commandes pour ...**

- ✓ **Créer un utilisateur :**

- adduser nom_user**

- ✓ **Affecter un mot de passe :**

- passwd nom_user**

L'utilisateur peut évidemment modifier son mot de passe avec cette même commande

La Sécurité – Les utilisateurs

- Résultats de cette création :
 - ✓ Ajout d'un compte pour l'utilisateur (uid) avec un mot de passe éventuellement
 - ✓ Création d'un groupe dit primaire (gid) qui par défaut porte son nom
 - ✓ Création d'un répertoire personnelle

/home/nom_user

Répertoire initialisé avec un contenu situé dans /etc/skel

La Sécurité – Les utilisateurs

- Les 2 premiers points se traduisent concrètement par l'ajout d'une ligne dans les fichiers :

`/etc/passwd`

`/etc/group`

- `/etc/passwd` et `/etc/group` sont deux fichiers texte que l'administrateur peut modifier avec un éditeur de texte.

La Sécurité – Les utilisateurs

➤ Une ligne du fichier `/etc/passwd` se présente ainsi (séparateur de champs `':'`) :

`sandra:*:1002:1002:Sandra Garnier:/home/sandra:/bin/bash`

✓ Respectivement

- Nom de l'utilisateur (login)
- Mot de passe crypté (cf. fichier `/etc/shadow`)
- Numéro de l'utilisateur (uid)
- Numéro de groupe (gid)
- Champ commentaire (nom par exemple)
- Le répertoire personnel
- Interpréteur de commande (shell)

La Sécurité – Les utilisateurs

Les paramètres de la commande *adduser*

-u uid	Pour fixer le numéro de l'utilisateur
-g gid	Pour fixer le numéro du groupe primaire. Le groupe doit exister dans /etc/group
-G liste	Appartenance de l'utilisateur à une liste de groupes (séparateur ",", " – sans espace)
-s shell	Attribution du shell par défaut /bin/bash
-c commentaire	
-d répertoire personnel	Par défaut le répertoire porte le nom de l'utilisateur et est placé dans /home
-m	Pour créer le répertoire personnel

La Sécurité – Les utilisateurs

Les paramètres de la commande *adduser*

-p mot de passe	Mot de passe crypté
-k rep-skel	Recopie le contenu de rep-skel dans le répertoire personnel, par défaut utilisation de /etc/skel.
-c commentaire	
-e date	Fixe une date d'expiration du compte (YYY-MM-DD)

Il existe également la commande **useradd** qui présente quelques différences avec *adduser*, en particulier elle n'initialise pas le profil de l'utilisateur, Ce travail est laissé à la charge de l'administrateur

La Sécurité – Les utilisateurs

- **Une ligne du fichier /etc/group** se présente ainsi (séparateur de champs " : ") :

comédie:*:1002:sandra,ingrid,michelle

sandra:*:1001:

users :*:1003: sandra,ingrid,michelle,alain

✓ **Respectivement**

- **Nom du groupe**
- **Mot de passe du groupe**
- **Numéro du groupe (gid)**
- **Liste des utilisateurs appartenant au groupe séparés par des virgules**

La Sécurité – Les utilisateurs

- Traditionnellement, pour des raisons de sécurité les mots de passe cryptés ne figurent pas dans le fichier */etc/passwd*.

Les utilisateurs ont accès à ce fichier lors de leur connexion, il est donc accessible en lecture.

- Les informations confidentielles sont stockées dans le fichier */etc/shadow*.
- Une ligne du fichier */etc/shadow* se présente ainsi (séparateur de champs " : ") :

nom_user:passwd:last_modif:may_modif:must_modif:avertir:expiration:desactiv:reserved

Par exemple - username:Npge08pfz4wuk:9479:0:10000:7:::

La Sécurité – Les utilisateurs

**nom_user:passwd:last_modif:may_modif: must_modif:
avertir:avant_expiration:date_expiration:réservé**

Où :

nom_user	Le Nom de l'Utilisateur
passwd	Le mot de passe crypté.
last_modif	Date de la dernière modification (en nombre de jours depuis le 1er janvier 1970).
may_modif	Nombre de jours avant que le mot de passe puisse être modifié (un 0 indique qu'il peut être changé à n'importe quel moment).
must_modif	Nombre de jours avant que le mot de passe ne doive être modifié
avertir	Nombre de jours durant lesquels l'utilisateur est prévenu de l'expiration de son mot de passe.

La Sécurité – Les utilisateurs

**nom_user:passwd:last_modif:may_modif: must_modif:
avertir: avant_expiration:date_expiration :réservé**

Où :

avant_expiration	Nombre de jours entre l'expiration du mot de passe et la fermeture du compte.
date_expiration	Date de la fermeture du compte
réservé	Champ réservé à une utilisation future



La Sécurité – Les utilisateurs

- Pour supprimer un mot de passe, l'administrateur supprime simplement le contenu du champs "mot de passe" avec un éditeur de texte.

La Sécurité – Les utilisateurs

➤ Autres commandes ...

- ✓ Modification des informations sur un utilisateur :

usermod paramètres nom_user

Les paramètres sont identiques à ceux de la commande *adduser*, par exemple :

usermod -u 1250 -c directrice sandra

Modification du numéro et du commentaire sur l'utilisateur sandra

La Sécurité – Les utilisateurs

➤ Autres commandes ...

✓ Suppression d'un utilisateur :

userdel nom_user

supprime la ligne de nom_user dans /etc/passwd

userdel -r nom_user

-r afin de supprimer également le répertoire

La Sécurité – Les utilisateurs

➤ Les commandes pour...

- ✓ Connaître les groupes d'un utilisateur :

groups nom_user

- ✓ Créer un groupe :

groupadd nom_groupe

Supprimer un groupe :

groupdel nom_groupe

- ✓ Ajouter un utilisateur à un groupe :

- Par usermod (usermod -G grp1,grp2 nom_user)
- Par l'édition du fichier /etc/group

La Sécurité – Les Fichiers

➤ Avec la commande `ls -l` on obtient un ensemble de lignes comme :

```
-rw-r--r-- 1 sandra sandra 1200 oct 19 12 : 39 page.html
```

Où respectivement on trouve :

- Les droits sur le fichier ou le répertoire (**-rw-r--r--**)
- Le nombre de liens sur le fichier (**1** signifie que le fichier n'a aucun lien qui pointe vers le même inode, **2** (ou plus) signifiant qu'il existe un lien (ou plus) vers le même inode.
- le nom du propriétaire du fichier (**sandra**)
- le nom du groupe propriétaire (**sandra**)

La Sécurité – Les Fichiers

-rw-r--r-- 1 sandra sandra 1200 oct 19 12 : 39 page.html

- La taille du fichier (**1200**)
- la date et l'heure de la dernière modification (**oct 19 12 : 39**)
- le nom complet du fichier (**page.html**)

➤ Les droits se décomposent en 4 zones :

.|...|...|... soit -|rw-|r--|r- -

- ✓ **"."** La première zone indique la nature du fichier
- "-"** pour un fichier normal,
- "d"** pour un répertoire,
- "l"** pour un lien symbolique

La Sécurité – Les Fichiers

- ✓ Les 3 dernières zones correspondent aux 3 catégories d'utilisateurs possible du fichier.

... | ... | ...
u **g** **o**

- La section **u** (u=user) fixe les droits accordés au propriétaire du fichier.
- La section **g** (g=group) fixe les droits accordés aux utilisateurs faisant partie du groupe auquel appartient le fichier.
- La section **o** (o=other) fixe les droits des autres utilisateurs.

La Sécurité – Les Fichiers

✓ Chaque zone propose à la catégorie correspondante 3 droits. Dans l'ordre

rwX

- **"r"** droit de **lecture** autorisé ; **"-"** non permise
- **"w"** droit **d'écriture** autorisé ; **"-"** non permise
- **"x"** droit **d'exécution** autorisé ; **"-"** non permise

La Sécurité – Les Fichiers

- Signification de ces droits sur les fichiers :
- **"r"** : Droit de lire le contenu du fichier
 - **"w"** : Droit d'écrire dans le fichier
 - **"x"** : Droit d'exécuter si le fichier est un script ou un fichier binaire

La Sécurité – Les Fichiers

- **Signification des droits sur les répertoires :**
 - **"r"** : Droit de lire le contenu du répertoire, afficher son contenu (commande ls)
 - **"w"** : Droit de modifier le contenu, créer, supprimer des fichiers et des répertoires (commandes cp, mv, rm)
 - **"x"** : Droit d'accéder aux fichiers du répertoire et droit d'exécution

La Sécurité – Les Fichiers

- **Le créateur d'un fichier en devient le propriétaire.**
- **Le groupe primaire auquel appartient le créateur au moment de la création devient le groupe du fichier.**
- **Le propriétaire a le droit de modifier tous les droits sur les fichiers et les répertoires dont il est propriétaire, y compris de transmettre son droit de propriété.**

La Sécurité – Les Fichiers

➤ Changer les droits d'accès :

chmod nouveau_droits fichier(s)

On utilise généralement les notations :

[**u**, **g**, **o**, **a**] pour désigner le bénéficiaire (a=all)

[**r**, **w**, **x**] pour désigner le droit

[**+**, **-**, **=**] pour désigner l'opération (ajouter, retirer ou affecter)

La Sécurité – Les Fichiers

➤ Changer les droits d'accès :

- ✓ Fixer de nouveaux droits (remplacement)

`chmod u=rwx,g=rw,o=r liste.txt`

pour obtenir : `rw-rw-r--`

`chmod g=rwx liste.txt`

`---rwx---`

les autres restent

inchangés

`chmod u=rwx,g=r,o= liste2.doc`

La Sécurité – Les Fichiers

➤ Changer les droits d'accès :

✓ Modifier les droits

`chmod g+rw liste.txt`

avant : -rwx--xr- résultat : -rwxrwxr--

`chmod go-rx document`

avant : -rwxrwxrw- résultat : -rwx-w--w-

`chmod a-x liste2.doc`

avant : -rwxrwxr-x résultat : -rw-rw-r--

La Sécurité – Les Fichiers

➤ Changer les droits d'accès :

✓ Modifier les droits, récursité

chmod -R nouveau_droits répertoire

Exemple :

chmod -R go-rwx /home/herve

Cette commande supprime tous les droits sauf au propriétaire sur tous les fichiers, répertoires et sous-répertoires à partir de /home/herve****

La Sécurité – Les Fichiers

➤ Changer les droits d'accès : ...|...|...

✓ Utilisation de l'octal

En binaire les droits affectés aux différents groupes vont de 000 (pour "---") à 111 (pour "rwx") en passant par 101 (pour "r-x")

En décimal ces valeurs vont de 0 à 7.

chmod 764 lettre.txt

chmod u=rwx g=rw o=r lettre.txt

Ces deux commandes sont équivalentes

La Sécurité – Les Fichiers

➤ Les commandes pour...

✓ Changer le propriétaire

chown [-R] new_proprio fichier

✓ Changer le groupe propriétaire

chgrp [-R] new_groupe fichier

(avec -R : l'option de récursivité)

Seul le propriétaire actuel (ou root) peut exécuter ces commandes