

Administration Linux

Ipssi Janvier2023

Introduction

Introduction: GNU , Linux, GNU/Linux, free software, open source

- GNU signifie GNU's not Unix
- GNU est un système d'exploitation, qui utilise le noyau Linux pour former le système **GNU/Linux**.



Introduction: GNU , Linux, GNU/Linux, free software, open source

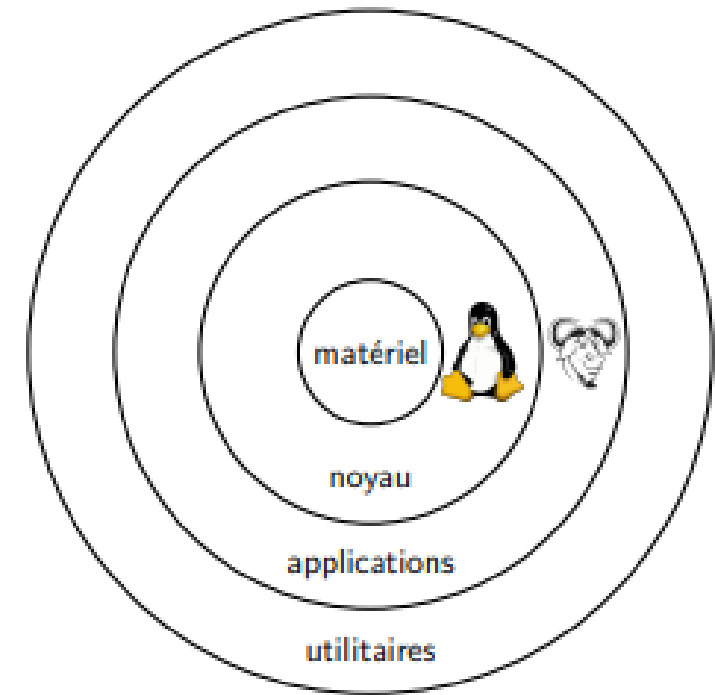
- À l'origine, le projet GNU désigne un système d'exploitation entièrement libre à base de noyau UNIX.
- Le projet **GNU** a initialement créé la plupart des composants et services utilisés dans **GNU/Linux** et a ensuite ajouté le noyau **Linux** pour créer le système d'exploitation

Introduction: GNU , Linux, GNU/Linux, free software, open source

- GNU et Linux sont du free software ; il s'agit de “logiciel libre”, et pas “logiciel gratuit”
- GNU définit quatre libertés essentielles :
 - le droit d'utiliser le logiciel sans restrictions ;
 - l'accès au code source et le droit d'étudier et modifier le logiciel ;
 - la redistribution sans restrictions du logiciel . . .
 - Et la modification (créer sa propre distribution)

Introduction: Noyau Linux (Kernel)

- Le noyau (ou kernel) Linux est le principal composant d'un système d'exploitation GNU/Linux et constitue l'interface entre le matériel d'un ordinateur et ses processus.
- Le noyau assure la communication entre les deux parties et gère les ressources aussi efficacement que possible.



Introduction: Noyau Linux (Kernel)

Le noyau Linux remplit quatre fonctions :

- 1. Gestionnaire de la mémoire** : suivi des contenus stockés, de leur emplacement et de l'espace mémoire utilisé
- 2. Gestionnaire des processus** : identification des processus susceptibles de solliciter le processeur, à quel moment et pour quelle durée
- 3. Pilote des périphériques** : médiateur/interprète entre le matériel et les processus
- 4. Gestionnaire des appels système et de la sécurité** : réception des demandes de service envoyées par les processus

Introduction: Distributions Linux



Introduction: Distributions Linux

Il existe plusieurs distributions GNU/Linux:

- Debian, la distribution de référence .
- Red Hat, l'autre distribution de référence maintenant commercial et pour les serveurs d'entreprise.
- Gentoo, la distribution à partir des sources.
- Slackware, la distribution la plus ancienne.
- Ubuntu, la distribution la plus aboutie basée sur la distribution Debian.

Introduction: Distributions Linux

- Fedora, la version communautaire et grand public de la distribution Red Hat.
- Centos, la version communautaire strictement équivalente à la version commerciale de la Red Hat Entreprise recompilée à partir des sources.
- OpenSuse, au préalable basée sur une distribution Slackware, maintenant autonome et passée sous contrôle de l'entreprise Novell.
- Mandriva, la distribution d'origine française basée sur une distribution Red Hat

Ligne de commandes

Introduction à la ligne de commandes

- La ligne de commande de Linux est un moyen d'interagir avec le système d'exploitation en utilisant des commandes textuelles au lieu d'une interface graphique
- Elle est souvent appelée "shell" ou "terminal"
- Elle peut être utilisée pour lancer des programmes, manipuler des fichiers et réaliser des tâches de configuration du système.

Introduction à la ligne de commandes

- les **commandes** sont les mots-clés qui indiquent à la shell ce qu'elle doit faire. Par exemple, la commande "ls" affiche la liste des fichiers et dossiers dans le répertoire courant.
- **Arguments** : les arguments sont des informations supplémentaires qui sont fournies avec la commande pour préciser comment celle-ci doit être exécutée. Par exemple, si vous utilisez la commande "ls" avec l'argument "-l", cela affichera la liste des fichiers et dossiers de manière détaillée.
- **Options** : les options sont des modifications qui peuvent être apportées à une commande pour en changer le comportement. Elles sont généralement indiquées par un tiret suivi d'une lettre ou d'un mot. Par exemple, l'option "-l" de la commande "ls" permet d'afficher la liste des fichiers et dossiers de manière détaillée.

Introduction à la ligne de commandes

- Les commandes basiques possèdent des options parfois très nombreuses ;
- Elles permettent de modifier le comportement du programme ;
- Si la commande que vous utilisez ne fait pas exactement ce que vous voulez, consultez le manuel (man) pour trouver la bonne option à utiliser ;
- La plupart des programmes GNU permettent de combiner les options de manière concise : `ls -lSh` au lieu de `ls -l -S -h`
- En général, l'ordre des options n'importe pas ;

Introduction à la ligne de commandes

- Shell est l'interface de la ligne de commande de Linux.
- Il existe plusieurs shells différents, chacun avec ses propres fonctionnalités et syntaxes.
- Le shell par défaut sur la plupart des distributions Linux est Bash (Bourne-Again Shell)

Principales commandes Linux/shell

- **pwd** : affiche le chemin du répertoire de travail actuel
- **ls**: affiche la liste des fichiers et dossiers dans le répertoire de travail actuel
- **cd**: permet de changer de répertoire de travail
- **mkdir**: permet de créer un nouveau dossier
- **touch**: permet de créer un nouveau fichier vide
- **cp**: permet de copier un fichier ou un dossier
- **mv**: permet de déplacer ou de renommer un fichier ou un dossier
- **rm**: permet de supprimer un fichier ou un dossier
- **cat**: permet d'afficher le contenu d'un fichier dans la sortie standard
- **less**: permet de lire le contenu d'un fichier de manière paginée

Principales commandes Linux/shell

- **grep**: permet de rechercher un motif dans un fichier
- **find**: permet de rechercher des fichiers en fonction de divers critères
- **sort**: permet de trier le contenu d'un fichier
- **uniq**: permet de supprimer les lignes en double dans un fichier
- **wc**: permet de compter le nombre de lignes, de mots et de caractères dans un fichier

Enchainement des commandes avec le Pipe

- Le Pipe '`|`' est une fonctionnalité de la plupart des shells qui permet de rediriger la sortie standard (stdout) d'une commande vers l'entrée standard (stdin) d'une autre commande.
- Cela permet de combiner plusieurs commandes en une seule commande en enchaînant leur exécution, ce qui est parfois appelé "enchainement de commandes".

Exemple:

```
ls | grep test
```

Structure du système de fichier

Gestion des dossiers/fichiers

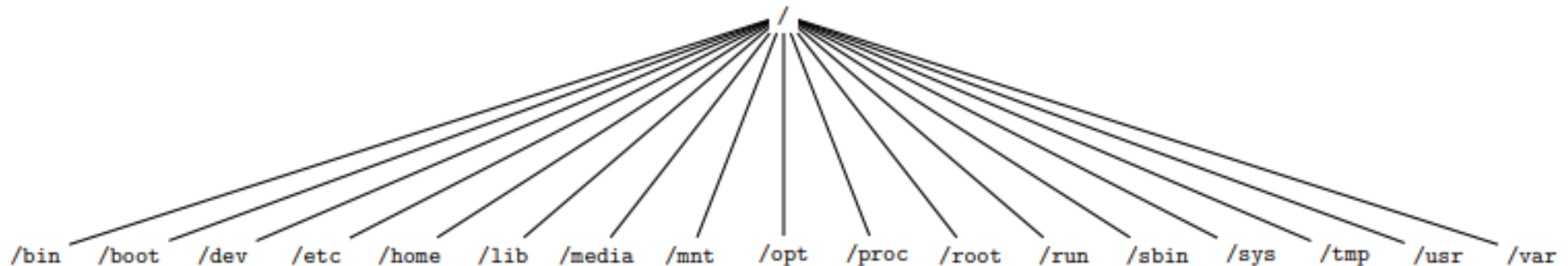
Pour Linux tout est un Fichier

Sous GNU / Linux, tout est un fichier :

- les fichiers ,
- les répertoires,
- les liens ,
- les périphériques,
- les entrées / sorties,
- ...

Structure du système du fichier

- Le système de fichiers structure les données sur le ou les disque(s)
- C'est une arborescence qui suit les conventions Unix, Il s'agit du standard FHS (File system Hierarchy Standard)
- Elle se présente comme suit :



Structure du système du fichier

- / est la racine (root)
- Le caractère / sépare les répertoires,
- Les majuscules et minuscules importent (Linux est case sensitive)
- Les commandes suivantes seront utiles :
 - cd** permet d'aller dans un répertoire ;
 - ls** affiche le contenu d'un répertoire ;
 - pwd** affiche le répertoire actuel (avec son chemin) ;

Quelques Répertoires importants

- `.` est le répertoire actuel ;
- `..` est le répertoire parent ;
- `/dev` (pour devices) contient le matériel (disques durs, processeurs, . . .) ;
- `/etc` contient les fichiers de configuration globaux ;
- `/home` contient les répertoires personnels des utilisateurs ;
- `/mnt` et `/media` contiennent les disques “montés” ;
- `/tmp` contient des fichiers temporaires : il est vidé à chaque redémarrage ;
- `/var` contient diverses données (en particulier des “logs” dans `/var/logs`)

Création, suppression et renommages des fichiers et des dossiers

- Pour créer un dossier sous Linux, on peut utiliser la commande **mkdir**, comme suit:
mkdir nom_du_dossier
- Pour créer un fichier vide, on peut utiliser la commande **touch**, comme suit:
touch nom_du_fichier
- Pour supprimer un dossier ou un fichier, on peut utiliser la commande **rm**
rm nom_du_dossier
rm nom_du_fichier
- Pour renommer un dossier ou un fichier, on peut utiliser la commande **mv**
mv ancien_nom nouveau_nom

Création, suppression et renommages des fichiers et des dossiers

- Il est important de noter que Linux ne propose pas de corbeille de suppression et que la plupart des distributions ne demandent à l'utilisateur de confirmer la suppression (ou le renommage) d'un fichier ou d'un dossier
- Il est donc recommandé d'être très vigilant lors de l'utilisation de ces commandes.

Les Utilisateurs et les Groupes

Les utilisateurs

- GNU est un système multi-utilisateurs : plusieurs utilisateurs peuvent s'y connecter en même temps et partager des ressources,
- Chaque utilisateur possède un nom (ex : alain) et un répertoire personnel dans /home (ex : /home/alain/) ;
- Le système peut théoriquement accueillir 232 utilisateurs
- On peut aussi regrouper les utilisateurs dans des groupes

Super-utilisateur

- Pour des raisons de sécurité, les utilisateurs ordinaires ne peuvent pas tout faire
- Seuls les super-utilisateurs (root) peuvent administrer le système ; par exemple, (dés)installer des programmes, ou accéder à des fichiers “critiques”
- Avec la distribution ubuntu la commande **sudo** permet d'exécuter une commande en tant que super-utilisateur ;

Les utilisateurs

Les attributs qui caractérisent un utilisateur Unix sont :

- un nom de connexion (login) ;
- un mot de passe ;
- un identifiant numérique unique (UID) ;
- un groupe primaire (GID) ;
- un commentaire (appelé gecos);
- le répertoire principal de l'utilisateur (home directory) ;
- un interpréteur de commandes (shell) par défaut.

Les utilisateurs

- L'ensemble de ces éléments est stocké dans le fichier **/etc/passwd** au format texte. Les champs sont séparés par le caractère « : ». Par exemple :
 alain:x:1001:1001:alain admin unix:/home/alain:/bin/bash
- Le super-utilisateur root a toujours l'UID 0
- Le fichier **/etc/shadow** contient les mots de passe cryptés des utilisateurs

Les groupes

- Les groupes sont identifiés par un nom et un **GID** (group identifier) ;
- Ils servent à définir des droits d'accès de manière plus globale et simple ; par exemple :
 - faire partie de audio permet d'utiliser les périphériques audio,
 - faire partie de lpadmin permet de configurer les imprimantes,
 - Etc.
- La commande **groups** affiche les groupes dont l'utilisateur fait partie
- Chaque utilisateur fait partie de son propre groupe ;

Les groupes

Les attributs qui caractérisent un groupe Unix sont :

- un nom ;
- un mot de passe (jamais utilisé) ;
- un identifiant numérique unique (GID) ;
- une liste d'utilisateurs membres.

La liste des utilisateurs peut être vide ou contenir un ou plusieurs noms d'utilisateurs séparés par un caractère « , ».

L'ensemble de ces éléments est stocké dans le fichier `/etc/group` au format texte. Les champs sont séparés par le caractère « : ». Par exemple :

```
groupe1:x:100:alain,sara,jean,nathalie
```


Gestion des utilisateurs

- Les commandes suivantes permettent de gérer les utilisateurs :
- **sudo adduser nom** : ajoute l'utilisateur avec le nom donné ;
- **sudo deluser nom** : supprime l'utilisateur avec le nom donné ;
 - son répertoire personnel n'est pas supprimé ;
 - Il faut utiliser l'option **--remove-home** ou **--remove-all-files** pour ce faire
- **sudo usermod nom_utilisateur** : modifie l'utilisateur avec le nom donné ;
- **passwd** : change le mot de passe de l'utilisateur actuel ;

Gestion des groupes

- Les commandes suivantes permettent de gérer les groupes :
 - **sudo addgroup nom_groupe** : ajoute le groupe avec le nom donné ;
 - **sudo delgroup nom_groupe** : supprime le groupe avec le nom donné ;
 - **sudo groupmod nom_groupe** : modifie le groupe avec le nom donné ;
- Si l'on veut ajouter ou supprimer un utilisateur d'un groupe, on utilise **usermod**

Gestion des permissions

Permissions

- Sous Linux chaque fichier ou dossier appartient à un utilisateur et à un groupe d'utilisateur.
- Chaque utilisateur a également un rôle de propriétaire ou de groupe
- Les permission ou droits d'accès aux fichiers et aux dossiers permettent de contrôler qui peut accéder à ces fichiers et dossiers et qui a le droit d'effectuer des actions sur ces fichiers et dossiers.

Permissions

- Il existe trois types de permissions sous Linux:

1- Lecture 'r' : permet de lire le contenu d'un fichier ou de lister les fichiers d'un dossier

2- Écriture 'w': permet de modifier le contenu d'un fichier ou de créer de nouveaux fichiers dans un dossier

3- Exécution 'x' permet d'exécuter un fichier ou de traverser un dossier (c'est-à-dire d'accéder aux fichiers et dossiers à l'intérieur)

Un fichier exécutable se lance avec la commande **./nom_fichier** ;

Permissions

- Il existe également des codes correspondants aux différentes permissions

Code	Signification
0	- - -
1	- - x
2	- w -
3	- wx
4	r - -
5	r- x
6	rw -
7	rw x

Permissions

Exemple de changement des droits d'accès d'un fichier en utilisant les codes

touch mon_fichier

chmod 123 mon_fichier

ls -l mon_fichier

Permissions

- Chaque fichier et dossier possède permissions pour le propriétaire, le groupe et tous les autres utilisateurs
- Par exemple, si un fichier a les droits '**rw- rw- r- -**' c'est-à-dire que le propriétaire a les droits de lecture et d'écriture, le groupe a les droits de lecture et d'écriture, et tous les autres utilisateurs n'ont que le droit de lecture.

Permissions

- Pour afficher les droits d'accès d'un fichier ou d'un dossier, vous pouvez utiliser la commande '**ls -l**'
- L'option **-l** de la commande **ls** affiche les informations détaillées

The diagram shows a single line of output from the `ls -l` command on a dark background. The text is `-rw-r--r-- 1 root root 27555 Jan 6 23:14 alternatives.log`. Brackets below the text group the fields into categories with labels: 'Permissions' (red) for `-rw-r--r--`, 'Propriétaire' (green) for `1 root`, 'Groupe' (yellow) for `root`, 'Taille' (black) for `27555`, 'Dernière Modification' (black) for `Jan 6 23:14`, and 'nom' (black) for `alternatives.log`. A blue arrow labeled 'Type' points to the first character of the permissions string.

```
-rw-r--r-- 1 root root 27555 Jan 6 23:14 alternatives.log
```

Type

Permissions Propriétaire Groupe Taille Dernière Modification nom

Permissions

- seul le propriétaire d'un fichier ou d'un dossier peut modifier ses permissions.
- Si on veut que d'autres utilisateurs puissent modifier les droits d'accès, il faut changer le propriétaire du fichier ou du dossier en utilisant la commande **chown**

Permissions

- Pour changer les permissions d'un fichier ou d'un dossier, vous pouvez utiliser la commande **chmod**

chmod droits_d_accès nom_du_fichier/dossier

- Par exemple, pour donner les droits de lecture et d'écriture au propriétaire et au groupe et seulement le droit de lecture aux autres utilisateurs pour un fichier nommé 'mon_fichier', on utilise la commande chmod comme suit:

chmod ug=rw,o=r mon_fichier

Visualisation et Edition des texte

Visualisation de texte

Pour visualiser le contenu d'un fichier sans modifier son contenu, on a les commandes suivantes:

- **cat :**
`cat nom_fichier` affiche le contenu d'un fichier en entier dans le terminal ;
- **Less:**
`less nom_fichier` affiche le contenu d'un fichier de manière interactive, on peut le faire défiler avec les boutons du clavier «page up » , « page down » ;
- **head :**
`head nom_fichier` affiche les 10 premières lignes d'un fichier ;
- **tail :**
`tail nom_fichier` affiche les 10 dernières lignes d'un fichier ;

Editeur de texte vi

- «*vi* » a été développé pour être utilisé sur des terminaux disposant d'un nombre limité de touches.
- *vi* possède différents modes de travail.
- Chaque mode permet d'exécuter différentes choses.
- Lorsqu'on démarre, on se retrouve trouve dans le mode "*commande*".
- on peut passer en mode "*édition*" (ou insertion) en tapant la touche « i »
- Pour revenir au mode commande on tape la touche «echap»

Editeur de texte vi

Passage en mode «édition »

Commande	Fonctionnalité
i	Insertion de texte devant le curseur.
a	Insertion de texte après le caractère.
I	Insertion de texte au début de la ligne.
A	Insertion de texte à la fin de la ligne.
o	Ajout d'une ligne avant la ligne courante et passage en mode " <i>édition</i> ".
O	Ajout d'une ligne après la ligne courante et passage en mode " <i>édition</i> ".
r	Remplace le caractère se trouvant par sous le curseur par le prochain caractère tapé.
R	Remplace tous les caractères jusqu'à la fin de la ligne ou jusqu'à ce que la touche <ESC> soit pressée.

Editeur de texte vi

Les commandes de recherche

Commande	Fonctionnalité
/texte	Recherche " <i>texte</i> " vers le bas.
n	Recommence la dernière recherche.

Editeur de texte vi

Les commandes pour effacer

Commande	Fonctionnalité
[n]x	Efface n caractère à partir du curseur. n est optionnel.
X	Efface la caractère précédant le curseur.
[n]dd	Efface n ligne à partir de la ligne en cours. n est optionnel.
[n]dw	Efface n mots à partir du mot en en cours. n est optionnel.
[n]x	Efface n caractère à partir du curseur. n est optionnel.

Editeur de texte vi

Les commandes d'édition

Commande	Fonctionnalité
[n]yy	Copie n lignes dans la mémoire tampon. n est optionnel.
p	Colle le texte contenu dans la mémoire tampon après le curseur.
P	Colle le texte contenu dans la mémoire tampon avant le curseur.
u	Annule la dernière action.
.	Répète la dernière action.

Editeur de texte vi

Les manipulations des fichiers

Commande	Fonctionnalité
:w [<i>fichier</i>]	Sauvegarde dans fichier. fichier est optionnel.
:wq	Sauvegarde le fichier en cours et quitte "vi".
:q	Quitte "vi". Si des modifications ont été effectuées mais non sauvegardées, la commande ne sera pas exécutée.
:q!	Quitte "vi". Si des modifications ont été effectuées mais non sauvegardées, elles seront perdues.

Travaux pratiques

- Dans les prochaines séances de TPs on va travailler avec des machines virtuelles sur lesquelles on va installer la distribution Ubuntu 22.04.1 LTS
- Dans cette première séance de TP, on va réaliser les installations et configurations nécessaires de VirtualBox et Ubuntu et s'initier avec les commandes Unix de base pour: gérer des dossiers/fichiers, gérer des utilisateurs/groupes, ...