

Memo des commandes LINUX Part 1

ordinatous ¹

¹Self Institute

Le shell

Lorsque vous lancez le **terminal** , vous accédez à un interpréteur de commande, qui est très souvent **bash**; mais il en existe d'autres. (fish, kornshell, zsh..) C'est un outil vraiment puissant.

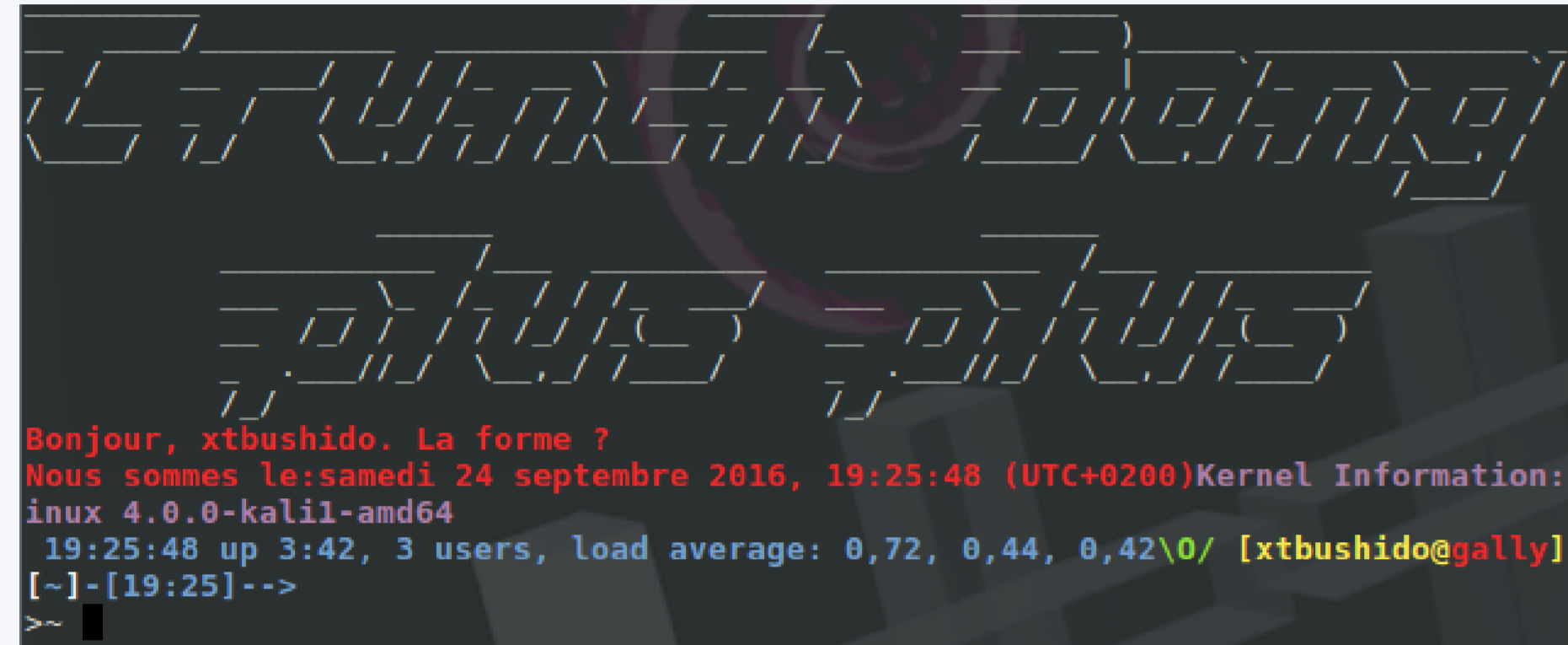


Figure 1:prompt

Le prompt de votre shell, ou session est personnalisable, ici une session d'un utilisateur standard. Le prompt indique en général l'utilisateur connecté à la machine , suivi de l'indication des droits root ou non, puis sa position dans l'arborescence.

A minima vous trouverez ceci: utilisateur@machine\$

C'est une variable (PS1) qui est définie dans votre .bashrc, on peut lui apporter quelques modifications pour colorer le prompt.

- **\$** -----> Signifie utilisateur standard
- **#** -----> Signifie utilisateur root
- **~** -----> Signifie Répertoire utilisateur

L'utilisateur standard a des droits uniquement sur son espace personnel, root a des droits privilégiés afin de modifier le système.

Il est possible de prendre la session root temporairement, ou d'exécuter une commande avec les droits nécessaire à la commande via l'utilitaire **sudo**.

Gestion des utilisateurs

Nous privilégierons l'usage de **adduser**, plus simple à utiliser que useradd. Par défaut l'utilisateur appartient également au groupe de son nom. Il peut être ajouter à un autre groupe. On peut également créer des utilisateurs système afin qu'ils exécutent des tâches sans que l'on puisse se connecter à leur session.

- **\$ sudo adduser toto** -----> Créé l'utilisateur toto (le mot de passe est défini à ce moment là)
- **\$ sudo adduser toto sudo** > Ajoute l'utilisateur toto au groupe sudo
- **\$ su toto** -----> Prend la session de l'utilisateur toto en restant dans son répertoire
- **\$ su - toto** -----> Prend la session de l'utilisateur toto en se déplaçant dans le répertoire de toto
- **\$ sudo chown ordinatous test-toto** > Change le propriétaire du fichier test-toto
- **\$ passwd** -----> Permet de changer son propre mot de passe
- **\$ sudo passwd toto** -----> Permet de changer le mot de passe de l'utilisateur toto sans connaître son mot de passe

Gestion des logiciels

- **\$ sudo apt update** -----> Met à jour la liste des paquets disponible
- **\$ sudo apt install nom-du-paquet** --> Installe le paquet

Manipulation des fichiers

- **ls**-----> Signifie list
- **ls** -----> Liste des fichiers du répertoire
- **ls -l** -----> Liste détaillée des fichiers du répertoire
- **ls -al** -----> Liste détaillée des fichiers du répertoire et les fichiers cachés
- **tree** -----> Affiche l'arborescence d'un répertoire
- **tree -L 3 projet** -----> Affichera les sous-répertoire du répertoire projet avec une profondeur de 3
- **pwd**-----> Signifie Print Working Directory
- **pwd** -----> Nom du répertoire courant
- **cd**-----> Signifie change directory
- **cd** -----> cd utilisé seul , fait revenir dans le répertoire utilisateur
- **cd /etc** -----> Positionnement sur le répertoire etc
- **cd ..** -----> Remonter dans l'arborescence
- **mkdir**-----> Signifie make directory
- **mkdir prog** -----> Création du répertoire prog
- **rmdir**-----> Signifie remove directory (seulement si il est vide)
- **rmdir prog** -----> Destruction du répertoire prog
- **Il existe plusieurs méthodes pour créer un fichier, via un editeur, ou la commande touch, soit une redirection**
- **vi test.txt** -----> Ouvre l'éditeur **vi** pour l'édition du fichier test.txt
- **touch test.txt** -----> Crée un fichier test.txt si il n'existe pas, sinon change la date de modification
- **mv** -----> Signifie move pour déplacer
- **mv test.txt test1.txt** -----> Déplace le fichier test.txt vers test1.txt , sert à renommer un fichier
- **cp** -----> Signifie copy , crée donc une copie
- **cp test1.txt test2.txt** -----> copie le fichier test1.txt vers test2.txt
- **rm** -----> Signifie remove , pour enlever un fichier
- **rm test2.txt** -----> Efface le fichier test2.txt
- **file test1.txt** -----> Indique le type de fichier
- **wc** -----> Signifie word count , compte le nombre de lignes, de mots, et caractères du fichier
- **wc test1.txt** -----> Résultat de la commande 7 4 19 test1.txt
- **ls > list.txt** -----> Envoie le résultat de la commande ls dans un fichier
- **ls /repertoire >> list.txt**>

Afficher le contenu d'un fichier

Les systèmes Linux et Unix-like disposent de plusieurs utilitaire, des **pager** pour afficher le contenu d'un fichier, tel que **more**, **less**, **tail**, il existe également **most** qui prend en charge la coloration syntaxique des man-pages par exemple.

La navigation se fait à l'aide de la barre d'espace pour aller de page en page, ou entrée pour défiler ligne par ligne. on peut également utiliser les flèches de navigation, o quitte d'ordinaire les pager avec la touche q (pour quitter)

Most est plus perfectionné et dispose d'un menu d'aide.

- **less test1.txt** -----> Ouvre le fichier test1.txt dans le pager **less**
- **more test1.txt** -----> Ouvre le fichier test1.txt dans le pager **more**, more dispose d'un compteur indiquant la position en pourcentage de progression
- **most test1.txt** -----> Ouvre le fichier test1.txt dans le pager **most**
- **tail test1.txt** -----> Affiche les 10 dernières lignes du fichier test1.txt
- **head test1.txt** -----> Affiche les 10 premières lignes du fichier test1.txt
- **cat test1.txt** -----> Affiche l'intégralité du document, cependant **cat** sert à concaténer é fichiers ensemble. **ATTENTION** en ouvrant un fichier très grand.

Modifications des droits sur un fichier

Il existe 3 types de droits : lecture **READ R**, écriture **write W**, et exécution **execute X**, par exemple lorsque vous récupérez un script , il faut parfois le rendre exécutable .

Souvent l'extension est en **.sh** , cependant c'est facultatif sous Linux . La nature du fichier étant indiqué dans son en-tête l'interpréteur de commande sait si il doit passer la main ou non. Pour bash c'est **#!/bin/bash**.

- **chmod +x script** -----> Ajoute les droits en exécution le fichier pour tous les utilisateurs.
- **chmod -r script** -----> Retire les droits en écriture le fichier pour tous les utilisateurs.

Recherche et modification d'une occurrence dans un répertoire (grep & sed)

```
$ grep -e occurrence -r /chemin/répertoire -l | xargs sed -i -e 's;occurrence;autre-occurrence;g'
```

Changer une occurrence dans un fichier

```
$ sed -i 's|jessie|stretch|' /etc/apt/sources.list
```

Installer un paquet .deb

```
$ sudo dpkg -i /chemin/du/package.deb ---> Installe un paquet.deb
```

Gestion des archives TAR

- **tar czf images.tar.gz images** -----> Création de l'archive TAR du répertoire images
- **tar xzf images.tar.gz images** -----> Dés archivage de l'archive TAR images

Commande d'aide

Les systèmes Linux et Unix-like disposent d'un manuel d'utilisation , il en existe 8 types différents.

1. Programmes exécutables ou commandes de l'interpréteur de commandes (shell)
2. Appels système (fonctions fournies par le noyau)
3. Appels de bibliothèque (fonctions fournies par les bibliothèques des programmes)
4. Fichiers spéciaux (situés généralement dans /dev)
5. Formats des fichiers et conventions. Par exemple /etc/passwd
6. Jeux
7. Divers (y compris les macropaquets et les conventions), par exemple man(7), groff(7)
8. Commandes de gestion du système (généralement réservées au superutilisateur)
9. Sous-programmes du noyau [hors standard]

- **man**-----> **man est le programme de visualisation des pages de manuel**
- **man ls** -----> Appel de l'aide pour la commande ls
- **-h** ou **--help** -----> Demande d'aide pour une commande
- **ls --help** -----> Demande d'aide pour la commande ls