

# ASR1- Système d'exploitation

# I. Introduction aux OS

Un OS exploite une machine pour faire interagir un ensemble de programme, il fait la liaison entre les ressources matérielles et les applications de la machine. Il dirige le fonctionnement afin d'initialiser le matériel, les fonctions de bases pour piloter les périphériques, la gestion des fichiers, assure la gestion, l'ordonnancement et la commande des processus.

Ils sont présent partout et il en existe des centaines (Windows, Linux, Solaris, AIX, HP-UX, FreeBSD, MAC OS, GCOS, VMS, AS400)

Les inconnus du grand public sont les mainframes ainsi que ceux qui sont dans les systèmes embarqués.

Nous utiliserons DEBIAN « Wheezy » 7.1 en dual boot avec Windows 7.

## II. Le cours

- Codage de l'information

- Architecture

- Type et caractéristiques des OS

- Applis clientes réseau, messagerie, transfert de fichiers, terminal virtuel, répertoires partagés

- Langage de commande, introduction à la programmation des scripts SHELL

- Gestion des processus (création, destruction, suivi ...) des fichiers et de l'utilisateur.

- Filtres

-Pour utiliser plusieurs OS sur un même ordinateur nous pouvons utiliser une machine virtuelle (VirtualBox), un programme (WUBI) ou les mettre en dual boot.

-Pour accéder à distance à nos fichier, un système de SSH est mis en place, voici la commande : `ssh nomutilisateur@info-ssh1.iut.u-bordeaux1.fr` , on peut également le mettre dans filezilla en mode sftp ou utiliser la commande `filezilla sftp://nomutilisateur@info-ssh1.iut.u-bordeaux1.fr`.

## III. Utiliser linux

Ctrl + F2 = Win + R

-Linux est créer par Linus Torvalds en 1992, il y a plusieurs distributions GNU/Linux (Debian, Ubuntu, Redhat, Mandriva, Fedora, Backtrack...) avec la norme POSIX. Son bureau que nous utilisons est Gnome Classic.

- La console (terminal) utilise des scripts Shell en Bash

- Les commandes sont du types ;

Nom -option paramètre

Chmod -w →redonner les droits d'accès

# Commande

## Utiliser les commandes

Le terminal fonctionne comme une boucle, une fois qu'une commande est exécutée, il en demande une autre.

### Pour voir les premières commandes :

Ls /bin (binary ce sont les commandes de bases.)

Ls /usr/bin (commandes ajoutées par l'utilisateur)

Ls /usr/local/bin mais encore ls /usr/local/games

Pour trouver toutes les listes de commandes

Echo \$PATH

Pour trouver le répertoire d'une commande : which [nomdelacommande]

### L'interprète de commande :

Le classement /opt/ pour toutes les commandes commerciales.

Les commandes superusers : dans /sbin pour les commandes de bases ou dans /usr/sbin

### Interprète avec boucle

Date -d « Sunday »

## Historique et complétion

Ctrl +r → ?

/.bash\_history pour voir l'historique des commandes

Sauf qu'aujourd'hui la loi n'autorise pas une personne tierce à voir ce que l'on a tapé donc inutile de vider son historique.

L'auto complétion de ce que l'on entre se fait par la touche tabulation.

## Navigation dans les répertoires

Cd → permet de retourner dans son répertoire personnel

Cd - → permet de retourner dans l'ancien répertoire de son historique

Cd [destination] → nous place dans le répertoire souhaité

Pwd → nous indique le répertoire actuel

Mkdir [nomdunrepertoire] → permet de créer un ou plusieurs répertoires

Rmdir → efface un ou plusieurs répertoires

Cp → avec ses options et ses sources permet de copier quelque chose.

Ls → permet de voir le contenu du dossier

-l (détaillé), -a (avec les dossiers cachés), -lh (détaillé en format humain), -t (par historique), -lt (par date avec le plus récent en premier), -ltr (par date avec le plus vieux en premier), -r (ordre alphabétique inversé), -R (récursive), -lR (récursive avec taille), -tree (affichage en arbre)

Du (diskusage) permet de voir ce qui prend de la place

-s (en résumant), -h (humain)

Du -s \*| sort -nr | head -h | affiche les 4 plus gros fichiers

Cat → affiche le contenu d'un fichier

-n (en numérotant les lignes), -b (en enlevant les lignes vides)

## Agir sur le contenu des fichiers

Less [nomdudossier] affiche page par page

## Développement

Pour indexer correctement un programme en C++ :

Astyle -style=[nomdunstyle]

Pour indexer correctement un programme en HTML

Tidy -im

## Metacaractère

Un meta caractère remplace n'importe quel caractère comme \*.cc signifie tous les fichiers terminant en .cc

## Création et utilisation d'archives

Archives : fichier contenant une arborescence comme zip, rar, tar anciennement sur bandes magnétiques, tgz, zoo, 7z

Créer une archive : tar -cf [nomsortie] [nomdelasource]

Tout extraire ou partie : tar -xf [nomdelasource] [partiedelasource]

Lister (voir) le contenu : tar -tf [nomdelasource]

Compresser : tar -czf [nomdesortie.tgz] [nomdelasource]

Avec z -->.tgz, avec j -->bzip2, avec Z (obsolète).

Décompresser : tar -tzf [nomsortie] [nomdelasource]

En zip :

Créer une archive : zip -r [nomsortie] [nomdelasource]

Voir une archive sans la déballer zip -l [nomdelasource]

Extraire une archive : unzip archive.zip -d [repertoire]

Extraire un fichier de l'archive : unzip archive.zip chemindufichier

## Le FTP :

On utilise la commande ftp voire sa version améliorée ncftp.

Pour demander un fichier et l'enregistrer sur le répertoire local on fait : get .....

Lls affiche le contenu du répertoire courant local et lcd rep change le répertoire courant local, lpwd affiche le chemin d'accès sur le répertoire courant local.

La commande get ne reconnaît pas \* pour le métalangage, on utilise alors mget.

Help donne la liste des principales commandes disponibles, pour les voir toutes, on fait help showall

! → Permet de lancer une commande Unix sur la machine locale

## Utilisation efficace de l'éditeur de texte Geany

Il faut connaître tous les raccourcis.

## Travailler à distance

En ssh : ssh users@nomduserveur

Pour une utilisation graphique : ssh (-X ou -Y)

Pour générer des clés : ssh -keygen -t [rsa ou dsa] Il faut créer deux paires de clés privés/publics qui faut copier dans ~/.ssh/authorized\_keys. D'abord les générer sur le compte local et les mettre ensuite sur le compte distant

## Copier des fichiers entre deux machines :

Scp [repertoirede la source ssh ou local] [cible ssh ou local] → il est inutile de se connecter avant.

Pour synchroniser deux répertoire distant et local → rsync -avz [repertoirede la source ssh ou local] [cible ssh ou local] et pour supprimer des fichiers qui ne sont plus sur la source mais qui sont encore sur la cible il faut ajouter - -delete-after.

Si l'on met time à côté d'une commande ça nous affiche le temps d'exécution.

## Filtres et redirection

Filtre : Commande qui prend un texte en entrée et qui fournit un texte en sortie.

Filtres usuels :

more / less ; cat ; find ; wc ; tr -d, -s ; sort ; head (-n) /tail ; grep (-n)

Faire un script: taper une ligne de commande dans un fichier .sh et le rendre exécutable.

Anytopnm convertir une image en pnm.

## Droits

- Droits → Système Multics (Multi-user pour faire plusieurs choses en même temps)
- ACL (access control list)
- Unix → simplification des utilisateurs et des groupes avec des droits différents

Interprétation : r -> Lire, w -> écrire, x -> exécuté

-rw (propriétaire) -rw (groupe) ..... (Le masque) + (présence d'ACL)

## Les codages

Codage Latin-1 : Un caractère par octet

ASCII → Lettres et chiffres non accentués (7bit) -> Ranger en hexadécimal en fonction des caractères accentués. Il y a différent codage pour unicode.

UCS 32 : 32 bits par caractères, UTF-8 → Un ou plusieurs octets

Od : afficher le contenu d'un fichier en octal

Pour savoir le type de fichier : file

BOM : Byte Order Marker

Plusieurs représentations d'un même mot : Big Indian et Little Indian

## Virtualisation

C'est un programme qui se comporte comme un second ordinateur (dans un autre ordinateur) like VirtualBox.