

Système d'exploitation

Jean-Sébastien Coron

Université du Luxembourg

September 26, 2009

- Les commandes usuelles du shell.
- Les permissions de fichier.

- Les commandes de manipulation de fichier.
 - `cd`: change le répertoire courant.
 - `ls`: affiche la liste des fichiers.
 - `mkdir`: création d'un répertoire.
 - `rmdir`: suppression d'un répertoire.
 - `cp source destination`: copie d'un fichier.
 - `mv source destination`: renommer ou déplacer un fichier.
 - `rm`: supprimer un fichier.
 - `cat`: affiche le contenu d'un fichier.
 - `more`, `less`: affiche le contenu d'un fichier avec des pauses.

- Manipulation de fichier (suite)
 - `chmod mode fichier`: modifie les droits d'accès au fichier.
 - `pwd`: affiche le répertoire courant.
- Autre commandes:
 - `which`: affiche le chemin exécutable correspondant à une commande.
 - `which ls` donne `/usr/bin/ls`.
 - `date`: affiche la date et l'heure.
 - `man commande`: affiche le manuel pour commande.

- Informations sur un fichier (commande `ls -l`).

- `drwxrwxrwx 2 Administ None 4096 May 9 10:52 www`
- `d`: type de fichier. `d`=directory, `-`=fichier, `l`=lien.
- `rwXrwxrwx`: permissions.
- `2`: liens sur ce fichier.
- `Administ`: propriétaire.
- `None`: groupe.
- `4096`: taille du fichier.
- `May 9 10:52`: date de dernière modification.
- `www`: nom du fichier ou répertoire.

- Chaque fichier UNIX a trois types de permission qui déterminent ce que l'on peut faire avec.
- Lecture:
 - Fichier: l'utilisateur peut lire le contenu du fichier
 - Répertoire: l'utilisateur peut avoir la liste des fichiers dans le répertoire.
- Ecriture:
 - Fichier: l'utilisateur peut modifier le contenu du fichier.
 - Répertoire: l'utilisateur peut créer ou supprimer des fichiers du répertoire.

- Exécution:
 - Fichier: l'utilisateur peut exécuter le fichier.
 - Répertoire: l'utilisateur peut se déplacer dans le répertoire.
- 3 catégories d'utilisateur.
 - L'utilisateur qui possède le fichier.
 - Le groupe auquel appartient le fichier.
 - Tous les autres utilisateurs.

- Structure:

- `rw xrwxrwx` pour utilisateur, groupe, autre.

- Exemple:

- `-rw-r---` 1 dupond User 9 Oct 14 10:00 toto
- Le fichier toto appartient à l'utilisateur dupond et au group User.
- L'utilisateur dupond peut lire et modifier le fichier.
- Le group User peut lire le fichier mais pas le modifier.
- Les autres utilisateurs ne peuvent ni lire ni modifier le fichier.

Commandes de permissions

- Seul le propriétaire d'un fichier ou l'administrateur peuvent modifier les permissions.
- La commande `chmod` permet de modifier les permissions d'un fichier.
 - `chmod options files.`
- `chown` permet de changer le propriétaire d'un fichier.
- `chgrp` permet de changer le groupe auquel appartient un fichier.

- `chmod parametre fichier`
 - Permet de modifier les droit d'accès au fichier `fichier`.
 - Seul le propriétaire du fichier ou l'administrateur peut le faire.
- Deux possibilités pour `parametre`:
 - Représentation octale:
`chmod 755 fichier`
 - Représentation symbolique:
`chmod +x fichier`

- Représentation octale: `chmod parametre file`
 - `parametre` se décompose en trois chiffres allant de 0 à 7.
 - Le premier correspond au propriétaire
 - Le deuxième correspond au groupe
 - Le troisième correspond au reste du monde
 - Exemple: `chmod 754 fich`
 - Chaque chiffre est la somme de:
 - (1) exécution, (2) écriture et (4) pour la lecture.
 - `chmod 754 fich` donne les droits:
 - exécution, écriture, lecture pour le propriétaire.
 - exécution, lecture pour le groupe.
 - lecture pour le reste du monde.

- Notation symbolique: `chmod parametre file`
 - Première partie: à qui s'applique la modification.
 - u pour l'utilisateur (le propriétaire du fichier)
 - g pour le groupe
 - o pour le reste du monde
 - Deuxième partie: ajout ou retrait de droit.
 - +: l'ajout de nouveaux droits d'accès
 - -: suppression de droits d'accès
 - Troisième: quels droits
 - r: lecture.
 - w: écriture.
 - x: exécution.

Commande `chmod`: exemples

- `chmod o-rwx fich`
 - Enlève les droits en lecture, écriture et exécution au reste du monde.
- `chmod u+x fich`
 - Permet à l'utilisateur d'exécuter le fichier
- `chmod g+r fich`
 - Permet au groupe de lire le fichier.
- `chmod u+rw,go+r fich`
 - Permet à l'utilisateur de lire et écrire le fichier, permet au groupe et au reste du monde de le lire.

- `tar` permet d'archiver un répertoire avec tous ses sous-répertoires dans un unique fichier que l'on appelle fichier archive.
- Créer une archive:
 - `tar -zcvf nomarchive nomfichiers`
 - `tar -zcvf prog.tar.gz *.cpp`: archive tous les fichiers `.cpp` dans `prog.tar.gz`.
 - `tar -zcvf rep.tar.gz rep`: archive le répertoire `rep` et tous ses sous-répertoires.
- Restaurer une archive:
 - `tar -zxvf nomarchive`