

Unix en trois pages

Ce document décrit quelques commandes de base du système Unix. Pour d'autres possibilités, voir un manuel approprié.

Conventions

1. Les commandes (sauf les caractères de contrôle) sont terminées par Return.
2. Unix distingue les majuscules et les minuscules.
3. La majorité des commandes ont une certaine valeur mnémotique en anglais. Par exemple `cd` = *change directory*, `rm` = *remove*, etc.

Connexion et fin de session

À la demande `login`, répondez par votre nom d'utilisateur. À la demande `Password`, donnez votre mot de passe. Pour terminer une session, tapez la commande `logout`

Aide

Pour la description d'une commande Unix, utilisez `man`. Par exemple, `man cd` explique le fonctionnement de la commande `cd`, et ainsi de suite. La commande `man man` explique la commande `man` elle-même!

Fichiers et répertoires

Les fichiers sont organisés dans une hiérarchie de répertoires. En général, rien ne distingue le nom d'un fichier d'un nom de répertoire. L'hiérarchie de chaque usager commence à son répertoire de base. Voici quelques exemples.

<code>.</code>	Le répertoire courant
<code>..</code>	Le parent du répertoire courant
<code>toto</code>	Le fichier ou répertoire <code>toto</code> dans le répertoire courant
<code>toto/fifi</code>	Le fichier ou répertoire <code>fifi</code> dans le répertoire <code>toto</code> du répertoire courant
<code>~/toto</code>	Le fichier ou répertoire <code>toto</code> dans votre rép. de base
<code>~fred/toto</code>	Le fichier ou répertoire <code>toto</code> dans le répertoire de base de l'utilisateur <code>fred</code>
<code>/usr/local/toto</code>	Le fichier ou répertoire <code>toto</code> dans le répertoire système <code>/usr/local</code>

Opérations sur les répertoires

<code>cd</code>	Vous place dans votre répertoire de base
<code>cd toto</code>	Vous place dans le répertoire <code>toto</code> du répertoire courant
<code>cd ~/toto</code>	Vous place dans le répertoire <code>toto</code> de votre répertoire de base
<code>cd ~fred</code>	Vous place dans le répertoire de base de l'utilisateur <code>fred</code> (s'il le permet)
<code>pwd</code>	Où suis-je?
<code>mkdir fifi</code>	Crée un nouveau répertoire <code>fifi</code> dans le répertoire courant
<code>rmdir fifi</code>	Enlève le répertoire vide <code>fifi</code> du répertoire courant

Opérations sur les fichiers

<code>ls</code>	Liste les fichiers et répertoires dans le répertoire courant
<code>ls -al</code>	Comme <code>ls</code> , mais donne plus d'informations
<code>ls toto</code>	Si <code>toto</code> est un répertoire, liste son contenu
<code>cp toto fifi</code>	Copie le fichier <code>toto</code> ; la copie s'appellera <code>fifi</code>
<code>cp ~fred/toto .</code>	Copie le fichier <code>toto</code> de l'utilisateur <code>fred</code> dans le répertoire courant; la copie s'appellera également <code>toto</code>
<code>mv toto fifi</code>	Change le nom du fichier <code>toto</code> en <code>fifi</code>
<code>mv ~/toto .</code>	Déplace le fichier <code>toto</code> de votre répertoire de base au répertoire courant
<code>rm toto</code>	Enlève le fichier <code>toto</code> du répertoire courant
<code>cat toto</code>	Affiche le contenu du fichier <code>toto</code> du rép. courant
<code>more toto</code>	Affiche le fichier <code>toto</code> page par page; tapez un espace pour avancer à la page suivante
<code>lpr toto</code>	Imprime le fichier <code>toto</code>

Protection

Changez votre mot de passe souvent avec la commande `yppasswd` (sur certains systèmes, c'est `passwd`¹). Comme mot de passe, n'utilisez ni un mot simple ni un chiffre évident comme votre date de naissance. Inventez plutôt quelque chose de compliqué comme `t42at5pm`. Protégez vos fichiers avec `chmod`. Dans cette commande il y a

¹ Sur JSP la commande est `chmp` mais on peut aussi bien utiliser `passwd` et `yppasswd` dont l'appel est équivalent à celui de `chmp`.

1. trois sortes d'usager: u (l'usager lui-même), g (son groupe), o (les autres) et aussi a (tout le monde);
2. trois sortes de permission: r (permission de lire ou de copier), w (permission d'écrire ou de changer), x (pour un fichier, permission de l'exécuter; pour un répertoire, permission de passer à travers, mais pas de lister son contenu);
3. trois opérations: = (définit les permissions), + (ajoute des permissions),
– (enlève des permissions)

<code>chmod a=rw toto</code>	Tout le monde peut lire ou changer le fichier <code>toto</code> , mais personne ne peut l'exécuter
<code>chmod go= toto</code>	Personne sauf le propriétaire n'a accès à <code>toto</code>
<code>chmod go=r *</code>	Le groupe et les autres peuvent lire tous les fichiers du répertoire courant
<code>chmod o-x toto</code>	La permission d'exécuter <code>toto</code> est enlevée aux autres
<code>chmod u+w toto</code>	La permission de changer <code>toto</code> est ajouté pour l'usager lui-même

L'histoire

Unix se souvient des dernières commandes que vous avez faites. Dans la table ci-dessous `ch1` et `ch2` sont des chaînes de caractères quelconques.

<code>!!</code>	Répète la dernière commande
<code>!ch1</code>	Répète la dernière commande qui commence par <code>ch1</code> . Ainsi <code>!vi</code> va probablement répéter la dernière commande d'édition
<code>^ch1^ch2</code>	Répète la dernière commande en remplaçant la chaîne <code>ch1</code> par <code>ch2</code> . Utile pour corriger des fautes de frappe. Par exemple: <code>%cim frde.sim</code> <code>%^de^ed</code>

Les alias

Les alias permettent de définir des abréviations pour des commandes longues ou compliquées, ou de changer le comportement du système.

<code>alias lo logout</code>	<code>lo</code> est une abréviation de <code>logout</code>
<code>alias prot 'chmod go='</code>	<code>prot</code> est une abréviation de <code>chmod go=</code>
<code>alias ls 'ls -al'</code>	Vous préférez toujours utiliser la forme plus longue de <code>ls</code>

S'il y a des alias que vous voulez définir chaque fois que vous travaillez, ajoutez leurs définitions dans le fichier `.cshrc` dans votre répertoire de base. (Normalement vous avez déjà une version de ce fichier fournie par les SI.)

En cas de panique ...

La notation `ctl/c` veut dire "taper `c` tout en maintenant la touche Control enfoncée", et ainsi de suite pour `ctl/z`, etc. Pour arrêter un programme qui boucle, tapez `ctl/c`. En principe `ctl/z` suspend un programme sans l'arrêter définitivement. Si ni l'un ni l'autre ne marche, essayez `ctl/y` et `ctl/d`.

P.B., 93-08-26