

Carnet de Travaux Pratiques

Système et programmation système

Licence 2 Informatique

Julien BERNARD

Introduction

Ces travaux pratiques vous permettent de vous entraîner à manipuler toutes les notions vues en cours. Si vous ne parvenez pas à finir le TP dans le temps encadré imparti, il sera obligatoire de le finir pour la semaine suivante.

Tout le travail demandé ne nécessite qu'un éditeur de texte (comme **Kate**) et un terminal pour le shell (comme **Konsole**). En particulier, il est expressément interdit d'utiliser un gestionnaire de fichier graphique (comme **Konqueror**).

La langue de la programmation est l'anglais. Vous devrez donc écrire tout votre code (commentaires inclus) en utilisant l'anglais. Cela concerne aussi bien les noms (variables, fonctions, etc) dans votre code que l'affichage.

Table des matières

Travaux Pratiques de Système n°1	3
Exercice 1 : Prise en main de l'environnement	3
Exercice 2 : Archives	4

Travaux Pratiques de Système n°1

Exercice 1 : Prise en main de l'environnement

Cet exercice doit vous permettre de prendre en main votre environnement de travail. Pour cela, connectez-vous d'abord sur votre compte avec votre nom d'utilisateur. Puis, ouvrez une console (**Konsole** par exemple) qui vous donne accès à un interpréteur de commande.

→ Connexion à votre compte

Avant de vous connecter sur votre compte, tapez CTRL+ALT+F1. Entrez votre nom d'utilisateur et votre mot de passe.

Question 1.1 Dans quel répertoire vous trouvez-vous ?

Question 1.2 À l'aide de la commande `ls(1)`, afficher le contenu de votre répertoire. Combien de fichiers et répertoires cachés avez-vous ?

Question 1.3 Effacer l'écran à l'aide de la commande `clear` ou avec la combinaison de touches CTRL+L.

Question 1.4 Fermer la session à l'aide de la commande `exit` ou avec la combinaison de touche CTRL+D.

Pour revenir à l'écran de login graphique, tapez CTRL+ALT+F7.

→ Commandes de base

Question 1.5 Tester les commandes vues en cours : `whoami(1)`, `uname(1)`, `uptime(1)`, `date(1)`, `cal(1)`, `echo(1)`, `man(1)`, `whatis(1)`, `apropos(1)`. En particulier, en cas d'options multiples, vous pouvez utiliser deux écritures : `-a -b -c` ou `-abc`. Le vérifier à l'aide de la commande `uname` par exemple.

Question 1.6 Comment obtenir une commande équivalente à `whoami` avec la commande `id` ?

Question 1.7 Dans quels sections pouvez-vous trouver une manpage appelée `time` ?

→ Système de fichier

Pour créer un fichier, il existe la commande `touch(1)`. En fait, cette commande met à jour le `atime` et le `mtime` d'un fichier (en le touchant), mais si le fichier n'existe pas, il est créé. Vous utiliserez cette commande pour créer des fichiers vide. Pour rappel, la commande pour créer un répertoire est `mkdir`.

En outre, pour les commandes `cp(1)`, `rm(1)`, `mv(1)`, vous testerez l'option `-i` qui permet de demander une confirmation.

Question 1.8 Créer un répertoire `SYS` dans votre répertoire utilisateur. Entrer dans ce répertoire puis créer un répertoire `exemple`. Entrer dans le répertoire `exemple`.

Question 1.9 Dans le répertoire `exemple` que vous venez de créer, créer les répertoires `skywalker/luke` en une seule commande. Puis créer un répertoire `skywalker/anakin` et un fichier `skywalker/anakin/README`.

Question 1.10 Supprimer le répertoire `skywalker/luke`. Que se passe-t-il si vous essayez de supprimer `skywalker/anakin` ?

Question 1.11 Renommer (en fait, déplacer) le répertoire `skywalker/anakin` en `darth_vader`.

Question 1.12 Créer un répertoire `yoda` et un fichier `yoda/README`. Copier le fichier `yoda/README` dans le fichier `yoda/README.old`, puis supprimer le fichier `yoda/README`.

Question 1.13 Supprimer récursivement le répertoire `yoda` (c'est-à-dire le répertoire et tout ce qu'il contient) à l'aide de l'option `-r` de `rm`.

Continuez à créer des fichiers et des répertoires et à les déplacer, copier, supprimer.

Exercice 2 : Archives

Cet exercice consiste à manipuler des archives (compressées ou non) à l'aide de la commande `tar(1)` (*Tape ARchive*). À l'origine, la commande `tar(1)` servait à créer des sauvegardes sur des bandes magnétiques (*tape*). Les bandes magnétiques avaient une plus grosse capacité de stockage que les disques durs mais avait un accès linéaire (c'est-à-dire que le temps pour accéder à une donnée était fonction de sa position sur la bande). La commande `tar(1)` a évolué pour créer des archives dans des fichiers.

La commande `tar(1)` prend en option :

- Une option de compression parmi :
 - aucune option si on ne veut pas de compression
 - `z` pour compresser avec `gzip`
 - `j` pour compresser avec `bzip2`
- Une action parmi :
 - `c` pour créer une archive avec des fichiers
 - `x` pour extraire les fichiers d'une archive
 - `t` pour lister le contenu d'une archive
 - `r` pour ajouter des fichiers dans une archive
 - `u` pour mettre à jour des fichiers dans une archive
- L'option `f` qui indique qu'on utilise un fichier dont le nom est indiqué
- L'option `v` (non-obligatoire) si vous voulez afficher le déroulement des opérations

Par exemple :

```
tar cf archive.tar repertoire
tar cf archive.tar fichier1 fichier2
```

Question 2.1 Créez une archive `exemple.tar` avec le répertoire `exemple` de l'exercice de prise en main.

Question 2.2 Allez chercher le fichier des figures sur MOODLE. Quels sont les fichiers contenus dans cette archive ?

Question 2.3 Extrayez l'archive puis créez une archive `inodes.tar.bz2` avec les deux figures concernant les inodes.

Question 2.4 Les options `z` et `j` sont en fait équivalentes à appeler directement `gzip(1)` et `bzip2(1)` (pour la compression) ou `gunzip(1)` et `bunzip2(1)` (pour la décompression). Décompressez l'archive `inodes.tar.bz2` sans en extraire les fichiers. Vous obtenez l'archive `inodes.tar`.

Question 2.5 Ajoutez les figures concernant des liens (symboliques et durs) à l'archive `inodes.tar` obtenue à la question précédente.