

# TP1 Système d'exploitation Unix (F.A.S.)

Commandes unix de base, gestion de fichiers, archives, l'éditeur *nano*

Vincent Berry - [vberry@lirmm.fr](mailto:vberry@lirmm.fr)

## Indications

Les temps indiqués à la fin des titres de paragraphes ne sont qu'indicatifs et approximatifs : votre expérience passée vous amènera à dépasser ou ne pas utiliser complètement le temps indiqué : ce n'est pas grave. Ceux qui ont plus d'expérience que les autres, sont invités à explorer à fond chaque aspect évoqué dans le TP, au besoin en effectuant des expérimentations complémentaires.

## Table des matières

<b>1</b>	<b>Gestion des répertoires (45mn)</b>	<b>1</b>
1.1	C'est parti pour le grand plongeon! . . . . .	1
1.2	Création d'une première arborescence (15mn) . . . . .	2
1.3	De l'image à la réalité (10mn) . . . . .	2
1.4	Archiviste (5mn) . . . . .	2
1.5	Une histoire de (pré)noms (10mn) . . . . .	3
<b>2</b>	<b>Sans l'explorateur de fichiers pour vérifier (25 mn)</b>	<b>3</b>
2.1	Les parties de l'arborescence du SGF situées en dehors de votre espace personnel (10mn) . .	3
2.2	Expérimentation dans votre espace de travail (15mn) . . . . .	4
<b>3</b>	<b>Apprivoisons l'éditeur de texte <i>nano</i> (20 mn)</b>	<b>5</b>

## 1 Gestion des répertoires (45mn)

Nous allons réaliser un peu de gymnastique avec les commandes de base. Certaines questions vous paraîtront répétitives, mais à travers la pratique intensive vous allez maîtriser certaines commandes très importantes sous Unix.

### 1.1 C'est parti pour le grand plongeon !

Trouvez l'application *Terminal* permettant de taper des commandes Unix. Affichez le nom du répertoire courant avec la commande `pwd` (*print working directory*).

Changez votre répertoire courant pour remonter dans le répertoire parent avec la commande `cd` (*change directory*) suivie du nom du répertoire. Indiquez trois moyens par lesquels vous pouvez vérifier que la commande précédente a fonctionné.

Changez votre répertoire courant pour revenir à votre répertoire initial à l'aide de la commande `cd` sans argument. Comment vérifier que vous êtes bien revenu à votre point de départ ? Essayez.

## 1.2 Création d'une première arborescence (15mn)

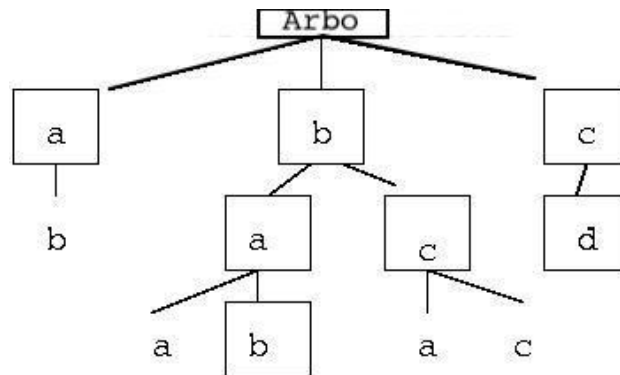
Placez vous dans votre répertoire d'accueil (utilisez la tilde ~ , sur Mac elle s'obtient par la combinaison alt + N). Dessinez sur la droite de la page l'arborescence obtenue par les commandes suivantes (tracez un carré autour des noms de dossier, voir ci-dessous pour un exemple) :

```
mkdir Test
cd Test
cp /etc/passwd a
mkdir b c
cd b
cp ../a d
mkdir ../e f
cd
cp a b/f/g
cd b/f
cp g ../../e
cd ..
rm ../a
rmdir ../c
mv ../e/g ../e/x
```

1. Dessinez l'arborescence résultante.
2. Quel est le répertoire courant à la fin de l'opération ?
3. Pouvez-vous expliquer pourquoi la commande `cd ~/a/f` ne fonctionne pas ?

## 1.3 De l'image à la réalité (10mn)

Quelle séquence de commandes vous permet de créer l'arborescence de la figure ci-dessous (les carrés sont des répertoires, les autres éléments des fichiers) ?



(note : la commande `touch nom-fichier` permet de créer un fichier du nom indiqué, de taille nulle mais existant quand même). Montrer le résultat à votre voisin ou votre enseignant pour voir s'il est convaincu que vous avez réussi à bien créer cette arborescence : il peut le vérifier par `ls -R` en se positionnant à la racine de cette arborescence.

À ce propos, dans cette arborescence, on a plusieurs fois un fichier de même nom, comment le système d'exploitation fait-il pour ne pas les confondre ?

## 1.4 Archiviste (5mn)

On veut maintenant récupérer toute l'arborescence **Arbo** pour la recopier sur un autre ordinateur, mais sans avoir à retaper toutes les commandes. Pour cela, on va créer une archive contenant une copie conforme d'**Arbo** et on pourra ensuite l'extraire sur l'autre ordinateur. L'idéal est de se placer dans le répertoire contenant **Arbo** : allez-y avec la commande `cd`.

Après avoir vérifié avec la commande `ls` que vous voyez bien **Arbo**, utilisez la commande `tar` pour créer une archive `arbo.tar.gz` contenant tout **Arbo**.

Vérifiez avec `ls` que l'archive a bien été créée. La commande `tar tf arbo.tar.gz` permet de voir le contenu de l'archive, vérifiez qu'il ne manque pas de répertoire ou fichier dedans.

Créez maintenant un répertoire `~/Autre_Ordi` (avec un *underscore*, pas un espace!) et déplacez-y l'archive `tar tf arbo.tar.gz` (pas le répertoire `Arbo`, mais l'image que vous en avez créé et mise dans l'archive). Déplacez-vous ensuite dans le répertoire `~/Autre_Ordi` et extrayez l'archive à cet endroit (commande `tar` avec options `xvf`). Vérifiez avec les commandes `ls` et `cd` que l'extraction vous a donné une copie conforme de l'arborescence que vous aviez au départ.

## 1.5 Une histoire de (pré)noms (10mn)

Revenez maintenant dans votre répertoire d'accueil et créez deux sous-répertoires, l'un ayant comme nom votre prénom, et l'autre comme nom votre nom de famille. Attention, évitez les accents et les espaces! Ci-dessous on suppose que ces dossiers s'appellent `Nom` et `Prenom`. Changez de répertoire pour vous positionner dans le répertoire `Prenom`, et créez plusieurs fichiers comme indiqué ci-dessous :

```
touch fichier1 > fichier2
ls -la > fichier3
cat fichier3 > fichier4
cat > fichier5 (tapez à ce moment du texte ensuite vous terminez par ctrl + d).
```

Copiez au choix deux de ces fichiers dans le répertoire `Nom` à partir du répertoire courant `Prenom` puis effacez une des copies se trouvant dans le répertoire `Nom`, toujours à partir du répertoire courant (`Prenom`), c'est-à-dire que vous agissez à distance : c'est possible en indiquant le chemin des fichiers concernés dans les commandes que vous tapez.

Changez le nom d'un des fichiers se trouvant dans le répertoire courant : commande `mv`.

Déplacez un de ces fichiers vers le répertoire `Nom`.

En restant dans le répertoire courant `Prenom`, copiez-y un des fichiers se trouvant dans le répertoire `Nom`.

Allez maintenant dans le répertoire `Nom` et affichez son contenu.

## 2 Sans l'explorateur de fichiers pour vérifier (25 mn)

Dans cette partie vous n'avez pas le droit d'utiliser l'explorateur graphique de fichiers (`Nautilus` sous Linux ou le `Finder` sous OSX) pour vérifier que vous ne vous êtes pas trompés (jouez le jeu!). Fermez donc toutes les fenêtres de cette application qui sont actuellement ouvertes dans votre session et mettez un terminal en plein écran<sup>1</sup>.

### 2.1 Les parties de l'arborescence du SGF situées en dehors de votre espace personnel (10mn)

Lancez un terminal de commande et utilisez les commandes de base pour répondre aux questions suivantes. Notez les réponses dans la colonne de droite

Initialement dans quel répertoire êtes-vous? Comment le savoir plus précisément que ce qu'indique l'invite de commande)?	
Déplacez-vous dans le répertoire <code>/tmp</code>	
Comment vérifier que vous êtes bien arrivé dans le bon répertoire?	
Fermez maintenant le terminal et ouvrez-le à nouveau. Dans quel répertoire vous situez-vous? qu'en concluez-vous?	

1. A l'extrême limite, s'il vous arrive d'être complètement perdu dans la suite, vous pourrez rouvrir une de ces fenêtres, mais uniquement pour connaître la forme de l'arborescence, pas pour agir dessus.

Comment savoir si vous avez les droits pour écrire dans le répertoire <code>/tmp</code> ? Par exemple, quel argument faut-il ajouter à la commande <code>ls</code> pour obtenir cette information ? Essayez de deux façons possibles : en désignant l'endroit intéressant de façon relative, puis en le désignant de façon absolue	
Dans la liste des répertoires et fichiers présents à cet endroit, comment pouvez-vous distinguer les répertoires des fichiers ?	
Revenez dans votre répertoire d'accueil en utilisant le raccourci <code>~</code> (oui, mais pas tout seul : indiquer d'abord le nom de la commande permettant de changer de dossier).	
Essayez de trouver le répertoire d'accueil de votre enseignant.	
Pouvez-vous vous déplacer dans le répertoire d'accueil de votre enseignant ? Pourquoi ?	
Pouvez-vous créer un fichier dans son répertoire d'accueil (par exemple par la commande <code>touch coucou.txt</code> une fois dans son répertoire) ? Pourquoi ? Comment prouver ce que vous dites ?	

## 2.2 Expérimentation dans votre espace de travail (15mn)

Revenez dans l'application Terminal, où nous allons faire quelques essais de commandes.

**Note :** dans le Terminal, il est possible de naviguer dans les commandes que vous avez tapées précédemment : on y accède par l'intermédiaire des touches flèches du clavier (haut, bas, gauche, droite). Ceci permet souvent de gagner du temps dans la saisie des commandes futures.

Remplacez-vous dans votre répertoire d'accueil	
En utilisant le manuel ci-besoin (commande <code>man</code> ), trouvez l'option de la commande <code>ls</code> qui vous permet de connaître tous les fichiers qui se trouvent dans ce répertoire, y compris les fichiers cachés (préfixés par un "point" dans les systèmes Unix) ?	
Pouvez-vous deviner les usages de certains fichiers cachés que vous observez ?	
Créez un répertoire nommé <code>UNIX</code> (commande <code>mkdir</code> ) dans votre répertoire d'accueil	
Déplacez-vous dans ce répertoire ( <code>cd</code> )	
Créez un sous-répertoire <code>TP2</code> dans <code>Unix</code> et déplacez-vous y	
Créez un fichier <code>titi</code> en utilisant <code>emacs</code> ( <code>xemacs</code> ). Vous pouvez par exemple y saisir : <i>Unix, j'ai même pas peur de toi, et d'abord je connais déjà plein de commandes</i>	
Quittez l'éditeur, revenez dans le Terminal et demandez à visualiser le contenu de ce fichier (commandes <code>cat</code> et <code>more</code> )	

Faites une copie de <code>titi</code> sous le nom <code>toto</code> (commande <code>cp</code> )	
Zut, il devait en fait s'appeler <code>tutu</code> . Renommez la copie du fichier en conséquence (commande <code>mv</code> )	
Double-zut, en fait, on voulait pas mettre le fichier dans le répertoire <code>TP2</code> , mais plutôt à la racine de votre arborescence personnelle. Déplacez le fichier (commande <code>mv</code> )	
En utilisant le mode absolu, changez les droits de <code>titi</code> de sorte que son propriétaire et le groupe aient les droits d'écriture et d'exécution, et les autres seulement le droit d'exécution ( <code>chmod</code> )	
Quelle option et quel argument de la commande <code>ls</code> vous permettent de vérifier l'attribution des droits sur ce fichier ?	
Ajoutez le droit de lecture pour les autres en utilisant le mode symbolique (c-a-d en utilisant un/des symbole/s <code>r,w,x</code> )	
Essayez de visualiser <code>titi</code> (commande <code>cat</code> ). Que faut-il faire pour y parvenir ?	
Remontez à votre répertoire d'accueil	
Tapez <code>cd U</code> puis appuyez sur la touche <code>TAB</code> afin d'expérimenter la complétion automatique. Validez	
Depuis le répertoire <code>Unix</code> créez un répertoire <code>Ursule</code> dans votre répertoire d'accueil (c-a-d que <code>Unix</code> et <code>Ursule</code> doivent être répertoires frères l'un de l'autre) en utilisant un <i>chemin relatif</i>	
Revenez dans votre répertoire d'accueil (commande <code>cd</code> ) et tapez à nouveau <code>ls U</code> suivi de la touche magique <code>TAB</code> Que se passe-t-il ? Pourquoi ?	
Essayez à nouveau en tapant maintenant deux fois sur la touche magique	

Avant de passer à la suite, remettez tous les fichiers texte manipulés ci-dessus dans le répertoire `TP2`

### 3 Apprivoisons l'éditeur de texte *nano* (20 mn)

Il s'agit d'un éditeur minimaliste fonctionnant à l'intérieur d'un terminal. Cet avantage permet de l'utiliser à distance, quand on est connecté sur une machine distante sans avoir besoin de voir l'écran de la machine distante (trop fort !). Avant tout, trouvez vos notes de cours sur l'éditeur `nano` et serrez les très fort, on va en avoir besoin (raccourcis, ...).

Dans un terminal, lancez l'éditeur `nano` en tapant simplement son nom. Pouvez-vous taper de nouvelles commandes dès lors ? S'exécutent-elles ? Pourquoi (expliquez ce qui se passe) ?

Bon, assurez-vous que vous êtes toujours dans `nano` (sinon, revenez-y), tapez une ligne de texte puis sauvegardez dans un fichier nommé `essai-nano.txt`. Quittez ensuite l'éditeur et par la commande `ls -l` assurez-vous que vous avez bien créé ce fichier et qu'il n'est pas vide.

Comment relancer nano en lui demandant de charger le fichier <code>essai-nano.txt</code> ?	
--	--

### Couper - Coller - Copier

Dans le navigateur web ouvrez la page <http://www.nano-editor.org> prenez dans le presse-papier le contenu de la page Documentation. Revenez dans le terminal (sortez de nano si vous y êtes), et tapez la commande suivante : `cat >> essai-nano.txt` puis videz le contenu du presse-papier, et tapez `CTRL + C`.

Normalement votre fichier <code>essai-nano.txt</code> a grossi de quelques lignes. Indiquer deux façons de le vérifier.	
---	--

Lancez à nouveau nano pour éditer ce fichier. Quels raccourcis vous permettent de localiser les occurrences du mot "new" dans le fichier ? Essayez.	
---	--

Sélectionnez maintenant la 1ère phrase du 2ème paragraphe de votre texte puis déplacez le curseur avec les flèches du clavier) et coupez cette phrase. Quel est le raccourci pour faire cette dernière action ? Placez ensuite la phrase que vous avez coupée au début du 3ème paragraphe.	
--	--

### Bien comprendre la circulation de l'information

Ouvrez une autre fenêtre de type terminal (dans le fenêtre en cours restez dans nano). Dans ce nouveau terminal, utilisez la commande `rm` pour supprimer le fichier `essai-nano.txt`. Utilisez la commande `ls` pour vérifier que le fichier n'existe plus sur disque.

Revenez dans le terminal où nano s'exécute, vous devez constater que le contenu du fichier <code>essai-nano.txt</code> existe toujours, <b>Question</b> mais comment est-ce possible puisque vous venez de le supprimer ? Est-ce vraiment le cas ? Pouvez-vous expliquer la situation au moment où vous venez de détruire le fichier par la commande <code>rm</code> ?	
--	--

Revenez maintenant dans l'éditeur de textes et sauvegardez votre texte à nouveau. Dans l'autre terminal inspectez le contenu du répertoire : que constatez-vous ? Expliquez pourquoi	
--	--

Ok, lancez maintenant `nano` pour éditer un fichier `mystere.txt` où vous écrivez deux trois mots sur deux lignes et sauvegardez ensuite le fichier. Sans quitter `nano`, dans un 2ème terminal (ou un autre onglet du même terminal), lancez l'édition du même fichier avec `nano`. Normalement le contenu des deux terminaux est à ce moment identique.

Maintenant, choisissez l'un de ces terminaux et modifiez le contenu du fichier édité, sans sauver pour l'instant. Le contenu de l'autre terminal (autrement dit du même fichier change-t-il ? Expliquez pourquoi	
--	--

Dans le terminal où vous avez changé le contenu du fichier édité sauvegardez maintenant le fichier édité.

Pensez-vous que le contenu de l'autre terminal va changer ? ... .. Allez voir maintenant dans l'autre terminal, est-ce que le contenu de ce deuxième terminal a changé ? Expliquez pourquoi	
---	--