

TP Système d'exploitation N°2 - MMI 1

Ce TP a pour objectif la prise en main d'une distribution GNU/Linux, Linux Mint 17. L'interface graphique est relativement similaire à ce qui existe sous Windows ou MacOSX. Nous allons donc nous intéresser aux spécificités de cet OS. Et pour commencer : le shell.

Commandes de bases

Le shell est l'interpréteur de commande de Linux. Pour l'ouvrir, choisissez dans le menu l'application 'Konsole'.

Chaque commande s'écrit de la manière suivante :

Syntaxe :

```
commande [options] <arguments>
```

séparateur : caractère espace

Commande :

Action à accomplir ou application à démarrer

Arguments

Objets ou fichiers auxquels la commande s'applique

Options

Modification du comportement de la commande

Commencent généralement par un - (moins)

Exemple : tapez la commande `ls` et observez ce qui s'affiche. Quel est l'équivalent de la commande `ls` sur Windows ?

-

Commandes *man*, *ls*, *cd*

Placez vous dans le répertoire `/home/utilisateur`. Les commandes `cd` ou `cd ..` qui existent sous Windows fonctionnent de la même manière ici.

Affichez le contenu du répertoire grâce à la commande `ls`. La commande `ls -l` permet un affichage plus détaillé.

Placez vous à la racine du disque avec la commande `cd /`. Tapez maintenant la commande `ls ~`. Qu'en concluez vous à propos du caractère `~` ?

-

Grâce à la commande `man`, trouvez l'option de `ls` qui permet d'afficher le contenu caché (c'est à dire les fichiers ou répertoires commençant par un `.` (point)).

-

Affichez le contenu de votre répertoire courant de telle sorte que les fichiers modifiés récemment apparaissent en premier.

-

Commandes *cat*, *grep*

cat est l'équivalent de TYPE sous windows. Cela permet d'afficher un fichier sur la sortie standard.

Exemple :

```
cat /usr/share/applications/vlc.desktop
```

grep, quant à lui, permet de rechercher une chaîne de caractère dans un fichier.

```
grep vlc /usr/share/applications/vlc.desktop
```

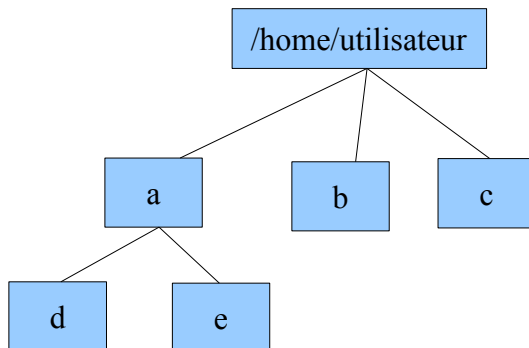
Commandes *mkdir*, *touch*, *rm*

Lancez la commande `touch foo`

Affichez le contenu du répertoire courant. Quelle est la taille de `foo` ? Qu'en déduisez-vous sur le rôle de la commande `touch`.

-

Grâce aux commandes que vous venez de voir, créez l'arborescence suivante :



Note : `a`, `b`, `c`, `d`, `e` sont tous des répertoires

Indiquez les commandes que vous avez utilisé :

-

Que fait la commande `cp foo ~/a` ?

-

Que se passe-t-il si vous essayez d'effacer le répertoire `b` ? Trouvez l'option de `rm` qui permet d'effacer les répertoires.

-

Quelle option ajouter à `cp` pour copier un répertoire ?

-

Indiquez la commande nécessaire pour copier le répertoire `a` dans `c`

-

Indiquez la commande nécessaire pour déplacer `~/a` dans `c/a/e`

-

Les droits

Tapez `ls -l`.

Rappel :

r – attribut 'read'

w - attribut 'write'

x - attribut d'exécution pour un fichier, ou attribut d'exploration pour un répertoire.

Essayez de créer un répertoire appelé `foo` dans `/usr/share`. Pourquoi cela ne fonctionne-t-il pas ?

-

Supposons qu'il existe un répertoire `toto/` qui a les droits suivants : `drwx--x--x`

Je ne suis pas propriétaire de ce répertoire; puis-je afficher le contenu du répertoire à l'aide de `ls` ?

-

On suppose que dans le répertoire `toto/` de la question précédente, il y a un fichier qui ne m'appartient pas :

`montexte.txt`. Ce fichier `montexte.txt` a les droits suivants : `-rw-r--r--`

Puis-je voir le contenu de ce fichier `montexte.txt` ? (justifiez votre réponse)

-

Commande `chmod`

Créez un fichier vide appelé `foo`. Tapez `ls -l foo`.

Tapez la commande `chmod u-w foo`. Tapez `ls -l foo` et indiquez ce qui a changé concernant `foo`.

-

Essayez d'effacer `foo`. Que se passe-t-il ?

-

Créez un répertoire `bar`. Donnez la commande permettant de réserver son exploration et sa lecture à votre utilisateur seulement.

-

Redirections

Comme sur windows, il est possible de rediriger l'entrée et la sortie standard.

Redirection de la sortie standard

```
ls -al > listerepertoire.txt
```

Redirection de l'entrée standard

Soit le script suivant dans le fichier appelé `foo`

```
#!/bin/bash
# commentaire
echo "Entrez un nombre:"
read nombre
echo "Le nombre entré est $nombre"
```

Un script ne peut être exécutable que s'il possède le droit d'exécution. Quelle est la commande à taper pour que `foo` puisse s'exécuter pour tous les utilisateurs ?

-

Exécutez le script une fois, en tapant ceci : `./foo`. Il vous demandera d'entrer un chiffre au clavier.

Créez un fichier appelé `entree` contenant le chiffre 54

Tapez la commande suivante :

```
./foo < entree
```

Qu'est-ce qui s'affiche à l'écran ?

-

Double redirection

Tapez maintenant :

```
./foo < entree > resultat
```

Que contient le fichier `resultat` ? Comment l'expliquez vous ?

-

Pipe

Reprenez les commandes `cat` et `grep` du chapitre précédent, et combinez les deux commandes en une seule, grâce au pipe.

-

Que fait la commande suivante ? :

```
cat /usr/share/applications/vlc.desktop | grep Name | grep f
```

-

Variables d'environnement

Tapez `echo PATH`, puis `echo $PATH`. Compte tenu de votre expérience sous windows (et du titre de cette rubrique...), interprétez ce qu'est `$PATH`

-

Annexe : Tableau de correspondance Linux/windows (non exhaustif...)

Linux	Windows	Usage
<code>cd</code>	CD	Changer de répertoire
<code>ls</code>	DIR	Afficher le contenu
<code>rm</code>	RMDIR	Effacer un répertoire ou un fichier (linux)
<code>mkdir</code>	MKDIR	Créer un répertoire
<code>cp</code>	COPY XCOPY	Copier un fichier ou un répertoire
<code>man</code>	HELP	Aide en ligne
<code>cat</code>	TYPE	Afficher un fichier
<code>more</code>	MORE	Afficher un fichier page par page
<code>rm</code>	DEL	Effacer un fichier
<code>rm -rf</code>	DELTREE	Effacer un répertoire
<code>mv</code>	REN MOVE	Déplacer ou renommer
<code>echo</code>	ECHO	Affiche une chaîne de caractère