

Utiliser les commandes Shell de base

Objectifs

En effectuant cet atelier, les étudiants apprendront à naviguer et à gérer des fichiers et des répertoires.

Contexte

Les commandes de base sont souvent nommées par de courtes abréviations de leurs fonctions. Vous allez utiliser entre autres la commande `ls` pour répertorier les fichiers, la commande `cp` pour copier des fichiers, la commande `mv` pour déplacer ou renommer des fichiers, la commande `rm` pour supprimer des fichiers, la commande `mkdir` pour créer des répertoires, la commande `rmdir` pour supprimer des répertoires (s'ils sont vides) et la commande `rm` pour supprimer récursivement des répertoires contenant des fichiers.

Ressources requises

Ordinateur équipé d'Ubuntu 14.04 Desktop LTS dans une machine virtuelle VirtualBox.

Ouvrez une fenêtre de terminal dans Ubuntu.

Connectez-vous à Ubuntu à l'aide des informations d'identification suivantes :

Utilisateur : admin

Mot de passe : admin



Cliquez sur l'icône du terminal pour ouvrir une fenêtre de terminal.



À l'invite de commande, vous allez manipuler les commandes de base GNU/Linux .

La plupart des commandes Linux suivent un modèle de syntaxe simple:

```
commande [options...] [arguments...]
```

Nous commencerons par des commandes simples et continuerons ensuite à exécuter des commandes avec des options et des arguments.

À l'invite de commande shell, tapez la commande `pwd` et appuyez sur la touche Entrée. La commande `pwd` affiche le répertoire où se trouve actuellement l'utilisateur dans le système de fichiers.

```
admin@localhost:~$ pwd

/home/admin

admin@localhost:~$
```

Le répertoire de travail est le répertoire dans lequel se trouve actuellement votre fenêtre de terminal Shell. Il s'agit également du répertoire courant ou homedir.

La sortie de la commande `pwd` (/home/admin dans cet exemple) est appelée un chemin. La première barre oblique représente le répertoire **racine**, le niveau supérieur de la structure du répertoire; **home** est un répertoire sous le répertoire **racine** et **admin** est un répertoire sous le répertoire **home**.

Exécutez la commande `ls` à l'invite et appuyez sur Entrée; cette commande listera les fichiers et répertoires qui se trouvent dans le répertoire courant:

```
admin@localhost:~$ ls

Desktop    Downloads  Pictures   Templates  Documents  Music
Public    Videos

admin@localhost:~$
```

Il est généralement possible de combiner plusieurs options dans n'importe quel ordre. Exécutez la commande `ls` suivante avec trois options :

```
ls -alr
```

Cette sortie de la commande `ls` a l'effet combiné de toutes les options spécifiées:

L'option `-a` inclut les fichiers commençant par un point.

L'option `-l` fournit le format de liste longue.

L'option `-r` trie le résultat dans l'ordre inverse.

Certaines commandes utilisent des arguments qui ne nécessitent aucune option. Par exemple, exécutez la commande `touch` avec un nom de fichier. Cela va créer un fichier vide. Exécutez ensuite la commande `ls` pour confirmer qu'il a créé le fichier:

```
admin@localhost:~$ touch lpicfile.txt
```

```
admin@localhost:~$ ls
Desktop    Downloads  Picture s  Templates  lpicfile.txt
Documents  Music      Public     Videos
admin@localhost:~$
```

De même, vous pouvez exécuter la commande `rm` avec le même nom de fichier comme argument pour supprimer ce fichier :

```
admin@localhost:~$ rm lpicfile.txt
admin@localhost:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
admin@localhost:~$
```

Certaines commandes, comme la commande `cp`, nécessitent au moins deux arguments. La commande `cp` nécessite la copie du fichier source et le fichier de destination (où copier le fichier). Par exemple, exécutez ce qui suit pour copier le fichier `hosts` du répertoire `/etc` dans le répertoire actuel avec un nouveau nom de `myhosts` :

```
admin@localhost:~$ cp /etc/hosts    myhosts
admin@localhost:~$ ls
Desktop    Downloads  Pictures  Templates  myhosts  Documents
Music  Public     Videos
admin@localhost:~$
```

La commande `mkdir` vous permet de créer (créer) un répertoire.

Pour créer un répertoire nommé `testdir` dans le répertoire actuel et vérifier que le nouveau répertoire existe, exécutez les commandes suivantes :

```
admin@localhost:~$ mkdir testdir
```

Pour créer plusieurs répertoires dans le répertoire actuel et vérifier qu'ils sont créés, exécutez les commandes suivantes :

```
admin@localhost:~$mkdir dir1 dir2 dir3 dir4 dir5

admin@localhost:~$ls -d dir?
```

La commande `rmdir` est utilisée pour supprimer les répertoires vides:

```
admin@localhost:~$rmdir testdir

admin@localhost:~$ls -d testdir/
```

Le shell Bash fournit une fonctionnalité conçue pour éviter les erreurs de frappe et économiser l'effort de frappe; cette fonctionnalité est appelée complétion de commande. L'achèvement de la commande vous permet de taper quelques caractères d'une commande (ou son argument de nom de fichier), puis d'appuyer sur la touche Tab. Le shell Bash terminera automatiquement la commande (ou son argument de nom de fichier) pour vous.

Par exemple, tapez d'abord `ec`:

```
admin@localhost:~$ ec
```

Ensuite, appuyez sur la touche Tab, et le shell terminera automatiquement l'écho de la commande pour vous (il place même un espace après la commande afin que vous puissiez immédiatement commencer à taper l'option ou l'argument):

```
admin@localhost:~$ echo
```

Lorsque les caractères saisis sont insuffisants pour correspondre à une seule commande, tous les choix possibles s'affichent si vous appuyez deux fois sur la touche Tab:

```
admin@localhost:~$ ca
cal                capsh                cat cautious-
lauchercalendar    captainfo                catchsegv caller
case               catman

admin@localhost:~$ cal

April 2020
```

```
Su Mo Tu We Th Fr Sa
1  2  3  4
5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
    26 27 28 29 30
admin@localhost:~$
```

Pour accéder aux pages de manuel de la commande `ls`, exécutez la commande suivante :

```
admin@localhost:~$ man ls
```

Voici un tableau de quelques options qui vont vous permettre de manipulation les pages du manuel :

Commande	Fonction
Return (ou Enter)	Descendre d'une ligne ;
espace	Descendre d'une page.
/terme	Recherche de mot terme.
n	trouver l'élément de recherche suivant.
1G	Aller au début.
G	Allez jusqu'au bout.
h	Afficher l'aide.
q	Quitter la page man

Lorsque vous voulez quitter la page de manuel de la commande `ls`, appuyez sur la touche `q`

A VOUS DE JOUER ...

Testez les commandes élémentaires relatives aux fichiers et répertoires :

1. Créez le répertoire **cours** sous votre répertoire d'accueil, puis un répertoire **tp1** sous le répertoire **cours**.
2. Effacez le répertoire **cours** . Que constatez-vous ?
3. Après avoir effacé les répertoires **tp1** et **cours**, créez à l'aide d'une seule commande les répertoires **cours**, **cours/tp1**, **cours/tp2**
4. Renommez le répertoire **cours** en **test**
5. Copiez un fichier de votre choix du répertoire **/bin** dans le répertoire **test/tp1** :
 - a) Depuis le répertoire **/bin**.
 - b) Depuis le répertoire **test/tp1**.
 - c) Depuis votre homedir, en utilisant le chemin absolu.
 - d) Depuis votre homedir, en utilisant des chemins relatifs.
6. Effacez à l'aide d'une seule commande les répertoires **test** **test/tp1** et **test/tp2**
7. Créez le dossier **gnu** dans votre répertoire d'accueil.
8. Créez un fichier vide **fic** dans le répertoire **gnu** et indiquez ensuite ses droits par défaut.
9. Modifiez les permissions du répertoire **gnu** pour supprimer tous les accès à tout le monde.
10. Ajoutez l'accès en lecture, écriture et traversée pour vous seul .
11. Ajoutez l'accès en lecture et traversée pour le groupe.
12. Créez le répertoire **gnu/tmp** , puis le fichier **gnu/tmp/perm**, et positionnez successivement les permissions adéquates pour que vous puissiez :
 - a) Lire, modifier et supprimer perm.
 - b) Lire, modifier mais pas supprimer perm.
 - c) Lire, supprimer mais pas modifier perm.

Voici la liste des commandes GNU/Linux à utiliser

<code>ls</code>	Liste les fichiers d'un répertoire, affiche les attributs d'un fichier.
<code>cp</code>	Copie de fichier.
<code>rm</code>	Supprime un fichier.
<code>mv</code>	Déplace ou renomme un fichier.
<code>cd</code>	Change de répertoire.
<code>touch</code>	Crée un fichier vide ou mettre à jour la date de modification
<code>mkdir</code>	Crée un répertoire.
<code>rmdir</code>	Supprime un répertoire.
<code>chmod</code>	Modifie les droits d'un fichier.
<code>pwd</code>	Dans quel répertoire suis-je ?
<code>adduser</code>	Créer un compte SHELL utilisateur
<code>sudo</code>	Pour Substitute User DO Héríte des droits d'administration (root)
<code>su</code>	Change d'utilisateur shell