

Initiation à Linux/Unix

AliENS

Table des matières

Utilisation de Linux / Unix	2
S'habituer à l'environnement graphique	2
Préparer une future connexion depuis l'extérieur	2
Commençons à utiliser le terminal.	2
Lancer des programmes	2
Place aux couleurs	2
Gestion de fichiers en ligne de commande	3
Manipulation	3
Fichiers et dossiers cachés	3
Contenu des fichiers	3
Connexion à distance : SSH	3
Utilisation de programmes distants	4
Se documenter	4
Gestion des permissions	4
Archives	4
Recherche de fichiers	4
Fin du TD	5
Utilisation avancée du shell	5
Gestion des processus	5
Pipes	5
Utilisation avancée de find	5
Utilisation avancée des pipes	5
Programmation shell	6

Cette feuille de TD est destinée à l'apprentissage, par la pratique, des commandes de bases d'Unix, qui sont utilisées sous GNU/Linux.

Elle est conçue pour être suivie sur les machines des salles libre service, qui fonctionnent sous Ubuntu. Cependant, n'importe quelle distribution GNU/Linux fera l'affaire.

Il est fortement recommandé de conserver la référence à portée de clavier.

<http://vrac.aliens-lyon.fr/linux/page-support.pdf>

En cas de doute sur une question, la réponse s'y trouve probablement.

Utilisation de Linux / Unix

S'habituer à l'environnement graphique

- Lancez (sur 4 espaces de travail distincts) Firefox, un terminal, LibreOffice Writer, et affichez votre dossier "Documents".

Pour aller plus loin : N'utilisez que le clavier pour faire cet exercice.

Préparer une future connexion depuis l'extérieur

- Activez votre accès SSH aux machines du parc informatique de l'ENS de Lyon depuis l'extérieur.

Commençons à utiliser le terminal...

Rappel : en cas de blocage, consultez la page suivante. N'hésitez pas à poser des questions si vous êtes perdu.

<http://vrac.aliens-lyon.fr/linux/page-support.html>

- Lancez un terminal.

Pour aller plus loin : Trouvez 3 façons de lancer un terminal.

Lancer des programmes

- Lancez la commande `whoami`. Observez la réponse.
- Lancez la commande `xeyes`. Que se passe-t-il dans le terminal tant que le programme est ouvert ?
- Lancez la commande
`zenity --info --text='Avec Linux, je peux tout faire' puis notify-send "C'est vraiment puissant"`
- Lancez la commande `sleep 15`. Observez. Perdez patience. Interrompez le programme `sleep` sans fermer le terminal.
- Lancez la commande `ls`, puis `ls -l`. Observez que l'**option** rajoutée modifie le comportement du programme `ls`.

Pour aller plus loin : Lancez la commande `sleep 120`, stoppez-la en arrière plan, reprenez-la en arrière-plan, puis au premier plan et enfin terminez-la.

Place aux couleurs

Pour avoir des couleurs et des alias sympathiques, nous vous conseillons d'utiliser notre fichier de configuration pour `bash`.

- Lancez la commande suivante :
`wget http://vrac.aliens-lyon.fr/linux/.bashrc`

- Relancez votre terminal pour que les modifications soient prises en compte.

Gestion de fichiers en ligne de commande

Pour travailler, commencez par créer un fichier texte, contenant n'importe quoi (par exemple, avec un éditeur de texte graphique).

Manipulation

- Affichez, dans le terminal, les fichiers de votre répertoire personnel.
- Créez le dossier `~/Initiation/test/`, puis déplacez-y vous.
- Copiez un fichier quelconque dans ce dossier.
- Supprimez le fichier que vous venez de copier.
- Y a-t-il des fichiers et dossiers en dehors de votre répertoire personnel ? Si oui, essayez de créer un fichier en dehors de votre répertoire personnel, par exemple dans `/usr/`. Vous pouvez utiliser `touch` pour créer un fichier vide.

Pour aller plus loin : Créez en plus le dossier `~/Initiation/test2/` contenant 2 fichiers texte et copiez ce dossier dans votre dossier personnel.

Fichiers et dossiers cachés

- Dans votre répertoire personnel, affichez la liste des fichiers et dossiers, et cherchez-y un dossier nommé `.mozilla`. Le trouvez-vous ? Si non, pourquoi, et comment faire pour le voir ?

Pour aller plus loin : Cachez votre fichier quelconque créé auparavant.

Contenu des fichiers

- Déterminer si le fichier `/etc/os-release` est un fichier texte. Si oui, afficher son contenu.
- Déterminer le type du fichier `/usr/share/man/man1/mv.1.gz`. Est-ce une bonne idée d'afficher son contenu avec la commande `cat` ?

Connexion à distance : SSH

- Connectez-vous en SSH à `servs1s.ens-lyon.fr`, en utilisant la commande `ssh`.

Lors de la première connexion, `ssh` affiche un message d'avertissement, puisqu'il ne connaît pas le serveur. À moins que quelqu'un ne s'amuse à faire du **man-in-the-middle**, vous pouvez accepter l'avertissement.

- Une fois connecté en SSH, affichez la liste des utilisateurs connectés, et informez-vous sur l'utilisation du CPU, de la RAM et de l'espace disque sur cette machine.
- Déconnectez-vous pour revenir sur votre propre machine.
- Copiez un de vos fichiers depuis votre machine vers le dossier `/tmp/` de votre voisin. Et pour copier un dossier récursivement ?

Utilisation de programmes distants

- Notez votre adresse IP sur <http://ipof.me/>
- Fermez `firefox`.
- Lancez `firefox` sur la machine de votre voisin, grâce à `ssh`.
- Quelle est l'IP avec ce nouveau firefox ?

Note : Si vous essayez de lancer firefox sur une machine distante alors que firefox est déjà lancé en local, ça ne fonctionnera pas, Ubuntu se contentera de rajouter un onglet sur le firefox local.

Se documenter

Pour connaître l'usage d'une commande particulière, le **manuel** est la source de documentation privilégiée. On y accède en utilisant la commande `man`, en passant en argument la commande souhaitée.

- Copiez l'intégralité du dossier `/usr/include/X11` dans `/tmp/`. Cela fonctionne-t-il avec `cp` ? Quelle option de `cp` est-il nécessaire d'utiliser pour copier le dossier récursivement (avec tout son contenu et ses sous-dossiers) ? Vous n'avez pas le droit d'utiliser Internet pour répondre à cette question, utilisez le manuel.
- Dans votre répertoire personnel, supprimez le dossier `Initiation` précédemment créé.
- Affichez la liste des fichiers de votre répertoire personnel, en format long, avec des unités de taille compréhensibles (KB, MB, GB, etc), et triés par date de modification croissante. (de même ici, utilisez la page de manuel)

Gestion des permissions

- En vous connectant en SSH, créez un dossier nommé `/tmp/partage-login` sur `servs1s.ens-lyon.fr`, en remplaçant "login" par votre login. Affichez-en les permissions. Permettent-elles à un autre utilisateur de lister son contenu ? Et de créer des fichiers à l'intérieur ?
- Créez un sous-dossier `secret` dans ce dossier, et copiez-y un fichier. Faites en sorte qu'un autre utilisateur ne puisse pas lister le contenu du dossier `secret`. Demandez à quelqu'un de se connecter sur `servs1s.ens-lyon.fr` pour vérifier.

Pour aller plus loin : Utilisez 2 instructions différentes pour modifier les permissions de `secret`.

Archives

- Créez un dossier `archive`, copiez quelques fichiers dedans, puis compressez-le en une archive `archive.tar.gz`.
- Déplacez vous dans `/tmp`, et extrayez l'archive précédemment créée.

Recherche de fichiers

- Listez tous les fichiers se terminant par `.cfg` dans `/etc`.

Fin du TD

Bravo, vous avez fini le TD! Pas de panique, il y a de quoi vous occuper.

Si d'autres personnes ont fini le TD, vous pouvez faire une partie de `hunt`, un jeu en mode texte qui se joue en réseau. Les touches de déplacement sont inspirées de vim (`hjkl`), et on utilise `f` pour tirer. Le manuel est suffisamment détaillé pour s'en sortir.

Note : sur les machines de l'ENS, `hunt` semble rentrer dans une boucle infinie, et ne fonctionne pas correctement. Exercice complémentaire : trouver d'où vient ce bug et le corriger.

Utilisation avancée du shell

Gestion des processus

- Lancer firefox, et le tuer à l'aide de `ps` et `kill`. Quels problèmes cela pose-t-il?
- Utiliser `killall` ou `pkill` pour tuer firefox.
- Utiliser `htop` en mode arbre (`tree`) pour tuer firefox.

Pipes

- Afficher toutes les lignes de `dmesg` contenant la chaîne de caractères "usb".
- Discuter de la pertinence des commandes suivantes :
 - `cat monfichier.txt | grep toto`
 - `find . | cat > /tmp/sortie.txt`

Utilisation avancée de find

- En utilisant `find`, trouver tous les fichiers de votre répertoire personnel vérifiant les critères suivants :
 - date de dernière modification : il y a moins de deux semaines
 - taille : entre 10 KB et 10 MB

Utilisation avancée des pipes

Télécharger le fichier <http://vrac.aliens-lyon.fr/linux/OSMTowns.txt.bz2> contenant les coordonnées GPS des villes de France.

Bonus : faire les questions suivantes en manipulant directement le `.bz2`, sans le décompresser au préalable.

- Compter le nombre de villes.
- Trouver la ville la plus à l'Ouest de France.
- Télécharger le fichier <http://vrac.aliens-lyon.fr/linux/OSMTowns-original.txt.bz2> et remarquer que le format n'est pas très pratique à manipuler. Le transformer en TSV (par exemple en utilisant `sed`). Comparer la sortie au fichier `OSMTowns.txt.bz2` téléchargé précédemment.

La question suivante est indépendante des questions ci-dessus.

- Chercher, dans votre répertoire personnel, tous les fichiers ou dossiers dont le nom contient un espace. Pour chaque nom de fichier ainsi obtenu, ne garder que la partie du nom qui suit le premier espace, puis remplacer tous les espaces restants par des underscores (« _ »). La solution doit tenir en une ligne. On pourra utiliser `find`, `cut` et `tr`.

Programmation shell

- Créer 1000 fichiers dont le nom suit la convention « toto/042ff1337/.TXT », où « 42ff1337 » est, pour chaque fichier, une chaîne aléatoire de longueur 8 composée de caractères hexadécimaux.
- Les extensions en majuscule, c'est moche. Renommer tous ces fichiers pour remplacer « .TXT » par « .txt », sans utiliser `rename`.
- Refaire la question précédente avec `rename`.
- Créer un programme permettant de savoir qui est connecté sur chaque machine de la salle informatique dans laquelle vous vous trouvez.
- Écrire un programme bash qui prend en argument une liste de noms de fichier, et qui affiche le nombre total de lignes dans les fichiers spécifiés. Vérifier que votre programme fonctionne bien même si on lui donne des noms de fichiers contenant des espaces.
- Étendre le programme pour qu'il puisse également prendre en argument des noms de dossier. Dans ce cas, tous les fichiers non-cachés de ce dossier (ainsi que les sous-dossiers, de façon récursive) sont pris en compte dans le calcul du nombre de lignes. Tester ce nouveau programme, en donnant en argument un mélange de noms de fichier et de dossier.