



Outils de développement logiciel

TP n° 1 : prise en main du terminal

Alain Lebreton

2023-2024

Objectif

Se familiariser avec le terminal et son interpréteur de commande, et apprivoiser quelques commandes Unix de base.

Pré-requis : chapitres 1 et 5 du polycopié de cours

Durée estimée : 1 séance

Introduction

Depuis le bureau, cliquez sur le bouton "Activités" situé à gauche de la barre supérieure. Dans le champ de recherche qui apparaît, entrez les premières lettres du mot "terminal", puis sélectionnez l'icône de l'application "Terminal" dès qu'elle paraît. Vous avez intérêt à redimensionner la fenêtre de votre terminal de manière à ce qu'elle occupe la moitié de votre écran, la seconde moitié étant réservée pour le navigateur qui vous permettra l'accès à ce document en ligne. Ça y est, sauf à de rares exceptions, vous ne manipulerez quasiment plus la souris de toute la séance, hormis pour lire ce document et éteindre l'ordinateur !

Lorsque vous avez ouvert votre terminal, celui-ci vous a placé par défaut à la racine de votre dossier personnel. Le dossier dans lequel vous vous trouverez à un instant donné, c.-à-d. le dossier courant, sera appelé "dossier de travail".

Voici quelques règles à retenir lorsqu'on utilise un terminal :

- dans le monde Unix, majuscules et minuscules sont différenciées (les commandes sont généralement en minuscules) ;
- dans le terminal, une commande n'est pas soumise tant que l'on n'a pas appuyé sur la touche **Enter** ;
- l'interpréteur de commandes dispose d'un mécanisme de complétion des commandes à l'aide de la touche de tabulation **↹** ;
- afin de corriger un des éléments de la commande en déplaçant le curseur, on utilisera les touches **←** et **→** ; de la même façon, les raccourcis **Ctrl**+**a** et **Ctrl**+**e** permettent de placer le curseur respectivement en début et en fin de ligne, **Ctrl**+**u** efface l'ensemble de la ligne et **Ctrl**+**c** abandonne la commande en cours ;
- un historique des commandes préalablement exécutées est accessible à l'aide de la touche **↑** (parcours inverse avec **↓**) – ou encore à l'aide de la commande **history** comme nous le verrons plus loin ;
- le raccourci **Ctrl** + **I** efface le contenu affiché sur le terminal – la commande **clear** réalise la même chose.

Une quasi “vidéo” comme rapport

Afin d’éviter la rédaction d’un rapport alors que nous n’avons pas encore pratiqué l’éditeur textuel **Vim**, nous vous proposons d’utiliser l’application **asciinema** qui est pré-installée sur Ubuntu (si vous souhaitez l’installer sur votre machine, rendez-vous sur le site <https://asciinema.org>). Cette application va vous permettre d’enregistrer les commandes que vous entrez dans le terminal et les résultats de ces commandes pour un exercice donné, dans un fichier d’animation. Voici un exemple d’exécution permettant d’enregistrer les manipulations de l’exercice n° 5 dans le fichier “tp01-nom1-nom2-ex5.cast” (l’option “-i=3” permet de ne pas tenir compte d’inactivités de plus de 3 secondes).

```
asciinema rec tp01-nom1-nom2-ex5.cast -i=3
```

Dans le cas où vous auriez stoppé l’enregistrement avant terme, relancez la commande précédente en ajoutant l’option “--append” de manière à continuer l’enregistrement du fichier.

Pour stopper l’enregistrement d’**asciinema**, cela se fait à l’aide de la commande **exit** ou de la séquence `Ctrl`+`d`.

Se documenter

La commande **man** affiche la page de manuel des commandes, des primitives du système et des fonctions de la bibliothèque standard du C.

Exercice n° 1

1. Affichez la page de manuel de la commande **man** elle-même (touche `q` pour quitter).
2. En langage C, une des premières fonctions que vous allez utiliser est `printf()` qui permet d’afficher un message sur le terminal. Affichez sa page de manuel (présente dans la section 3) à l’aide de la commande `man 3 printf`. Que se passe-t-il si l’on entre simplement la commande `man printf` ?
3. Vérifiez ce que fait la commande :
`printf "\t%s\n" "Bienvenue à l'ENSICAEN !"`
Le même résultat peut être obtenu avec la commande `echo` :
`echo -e "\tBienvenue à l'ENSICAEN !"`

Opérer sur les dossiers et fichiers

Vous allez apprendre à créer et supprimer des dossiers, à vous déplacer dans l'arborescence, à lister le contenu d'un dossier, à copier, renommer et supprimer des fichiers, et aussi à modifier les droits d'accès.

Les commandes qui vont être mises en oeuvre sont, par ordre alphabétique : **cd** (***change directory***), **chmod** (***change mode***), **cp** (***copy***), **du** (***disk usage***), **file**, **ls** (***list***), **mkdir** (***make directory***), **mv** (***move***), **pwd** (***print working directory***), **rm** (***remove***), **rmdir** (***remove directory***) et **tar**.

Exercice n° 2

1. Afin de bien comprendre la différence entre **chemin absolu** ("depuis la racine /") et **relatif** ("par rapport au dossier courant"), déplacez-vous de manière relative dans le dossier "Documents" qui se trouve à la racine de votre dossier personnel. Revenez dans le dossier parent (`cd ..`, ou plus spécifiquement dans le cas du dossier personnel avec `cd ~` ou encore `cd` sans argument). Enfin, retournez dans le dossier "Documents" de manière absolue cette fois.
2. Afin d'éviter de placer tous vos documents au même endroit, vous allez structurer le dossier "Documents" de votre dossier personnel en créant un sous-dossier que vous nommerez "odl" dans lequel vous vous placerez. Téléchargez le fichier "ressources-odl-fisa.tgz" en entrant la commande suivante :

```
wget http://web.ecole.ensicaen.fr/~alebret/ressources-odl-fisa.tgz
```

Puis décompressez-le à l'aide de la commande `tar -xvf ressources-odl-fisa.tgz`. Un dossier "ressources-odl-fisa" devrait apparaître dans "odl". Indiquez la taille de ce dossier à l'aide de la commande `du -h`, puis listez de manière détaillée son contenu. Vous pouvez supprimer le fichier "ressources-odl-fisa.tgz".

3. Créez à présent le sous-dossier "tp01" (dans "odl" :-), puis placez-vous dans ce dernier. "tp01" sera votre dossier de travail pour cette séance et vous y effectuerez toutes les actions.

Exercice n° 3

1. Créez le sous-dossier "ressources-bioinfo-2019" dans votre dossier de travail. Copiez dans ce sous-dossier le fichier "tumor-protein-p53.txt" que vous trouverez dans le dossier

"ressources-odl-fisa/tp01/", puis sans vous déplacer, listez de manière détaillée son contenu.

2. Copiez-y aussi le fichier "cell_stem.tex" que vous trouverez au même endroit. Renommez-le convenablement, car en anglais en effet, on emploie plutôt le terme "stem cell".
3. Autorisez l'écriture du fichier "tumor-protein-p53.txt" aux utilisateurs du même groupe que vous ainsi qu'aux autres, et vérifiez que les droits sont modifiés.
4. Empêchez à tout utilisateur l'accès du dossier "ressources-bioinfo-2019" (nouveaux droits : `rw-r--r--`) et vérifiez que vous n'y accédez plus.
5. Supprimez le dossier "ressources-bioinfo-2019". Auriez-vous pu utiliser une commande unique afin de supprimer le dossier et son contenu ?

Visualiser des fichiers texte

Afin de visualiser des fichiers sans forcément les éditer, vous utiliserez les commandes **cat** (dont ce n'est pas le rôle premier), **less** et **more**.

Exercice n° 4

1. Copiez dans votre dossier de travail ("tp01") le fichier "rouge_et_noir.txt" que vous trouverez dans le dossier "ressources-odl-fisa/tp01/", puis visualisez son contenu à l'aide de la commande `cat`. Ce n'est pas très pratique n'est-ce pas ? C'est normal, car ce n'est pas le rôle initial de `cat`.
2. Recommencez avec les commandes `more`, puis `less` (`Espace` pour avancer de page et `q` pour quitter).

Utiliser des jokers

Il peut devenir laborieux d'avoir à entrer une longue liste de fichiers (ou dossiers) afin de les passer en arguments à une commande. Les jokers (**wildcards** en anglais) permettent de définir des noms de fichiers ou de dossiers sans les indiquer en totalité. Ce sont des caractères spéciaux qui sont interprétés :

- "." désigne le dossier courant ;
- ".." désigne le dossier parent ;
- "*" désigne n'importe quelle séquence de caractères ;
- "?" désigne un unique caractère ;

- "[]" correspond à une liste de caractères précisés entre les crochets ;
- "-" désigne un intervalle de caractères.

Ces caractères peuvent toutefois apparaître dans les noms de fichiers ou de dossiers, à condition d'être précédés du caractère "\".

Exercice n° 5

1. Quel joker représente presque tous les noms de fichiers d'un dossier ? Lesquels n'apparaissent pas ?
2. Affichez à l'aide de ce joker les fichiers du dossier "ressources-odl-fisa/tp01/" dont le nom commence par "knock_".
3. Listez uniquement ceux du dossier "ressources-odl-fisa/tp01/" dont le nom commence par "knock_" et est suivi par trois chiffres, et dont l'extension est "txt".
4. Copiez ces derniers dans votre dossier de travail puis affichez de manière détaillée le contenu du dossier. Indiquez dans quel ordre les fichiers sont listés ? Affichez-les à nouveau, mais par taille décroissante (la taille sera indiquée avec des multiples d'octets).

Rediriger vers des fichiers

Toute commande interagit avec l'extérieur au moyen de trois fichiers spéciaux :

- l'entrée standard (stdin ou 0) ;
- la sortie standard (stdout ou 1) ;
- la sortie standard d'erreur (stderr ou 2).

Pour l'interpréteur de commande, ce sont par défaut : le clavier (0) et l'écran (1 et 2).

Il est possible de rediriger les entrées et sorties à l'aide des caractères spéciaux < et >. Ainsi, l'entrée standard est redirigée par 0< ou plus simplement <, la sortie par 1> ou >, et la sortie erreur par 2>.

Exercice n° 6

1. Lancez la commande **date**.
2. Réitérez en lançant la commande `date "+%d-%m-%Y"`. En consultant la page de manuel de la commande, comment parvenir à afficher la date et l'heure courante sous la forme : "jj-mm-aaaa (hh:mm:ss)" ?

3. Exécutez la commande précédente en ajoutant en fin de commande la chaîne "> date-du-jour.txt", puis visualisez le fichier "date-du-jour.txt" nouvellement créé.
4. Que devient le contenu de "date-du-jour.txt" si vous relancez la commande ?
5. Comment évolue-t-il si vous renouvelez l'opération, mais en doublant le chevron (>>) ? Et si vous répétez cette opération un certain nombre de fois ?
6. Le rôle initial de la commande cat est de concaténer des fichiers texte et d'afficher le résultat sur la sortie standard. À l'aide de cette commande puis d'une redirection, concaténez les fichiers "knock_001.txt" à "knock_004.txt" du dossier "ressources-odl-fisa/tp01/" dans un fichier du dossier de travail que vous nommerez "knock.txt". Vérifiez son contenu.



Comme vous venez de le constater, lorsqu'on utilise la redirection de sortie vers un fichier (">"), ce dernier est créé s'il n'existait pas et écrasé s'il existait déjà. Il est aussi possible de rediriger en "ajoutant" en fin de fichier (c.-à-d. sans écrasement), cela à l'aide de la redirection avec deux chevrons ">>".

Gérer des alias

Certaines commandes peuvent être longues à taper ce qui peut entraîner des erreurs de saisie, surtout si elles sont amenées à être fréquemment utilisées. La commande `alias` permet de créer un raccourci vers une commande.

Exercice n° 7

1. Créez un alias pour la commande `date` "+%d-%m-%Y (%T)" rencontrée précédemment. Testez-le.
2. Lancez un nouvel interpréteur de commande **bash** en entrant simplement la commande `bash`. Votre alias est-il toujours utilisable ? Et vos anciennes commandes ? Entrez la commande `exit` ou appuyez sur `Ctrl` + `d` afin de revenir dans l'état précédent.
3. Comme vous pouvez le constater, votre alias disparaît si vous ouvrez un nouvel interpréteur de commande ou encore si vous fermez votre terminal. Pour réutiliser un alias à chaque démarrage, il faudra le placer dans un fichier `.profile` disposé à la racine de votre dossier personnel. Vous apprendrez dans le TP suivant à éditer ce fichier avec **Vim**. Toutefois, si vous êtes vraiment pressé de découvrir le résultat, utilisez une commande présentée dans l'exercice n° 1 et le mécanisme de redirection que vous avez mis en oeuvre dans l'exercice n° 6.

Créer des liens

La commande **ln** permet de réaliser un lien avec un autre fichier ou dossier. Ce lien peut soit être normal (**hard link**), soit être symbolique (**soft link** ou **symbolic link**), un lien symbolique se comportant comme un alias du fichier ou du dossier.

Exercice n° 8

1. Dans votre dossier de travail, créez un sous-dossier "livres" et copiez-y tous les fichiers du dossier "ressources-odl-fisa/tp01/" qui concernent des textes littéraires.
2. À l'aide de la commande **ln**, créez un lien physique nommé "stendhal-r-et-n.txt", à partir du fichier "livres/rouge_et_noir.txt", puis listez de manière détaillée le contenu du dossier courant au moyen de la commande **ls -li**. Quels sont les points communs et les différences entre le lien physique et sa cible ? Affichez le contenu de "stendhal-r-et-n.txt".
3. Créez cette fois-ci un lien symbolique, appelé "livres_copie" sur le sous-dossier "livres". Que contient "livres_copie" ?
En listant derechef le dossier courant, indiquez quels sont les points communs et les différences entre le lien symbolique et sa cible ? Comment distingue-t-on qu'un lien symbolique vient d'être créé ?
Supprimez les deux liens précédemment créés. Les fichiers et dossiers cibles sont-ils toujours présents ?

Rechercher des fichiers

La commande **find** recherche des fichiers dans un ou plusieurs dossiers. **find** est appelée de la manière suivante :

```
find (dossier de départ) (matching criteria and actions)
```

Par exemple, pour rechercher tous les fichiers du dossier courant qui ont pour extension "txt", on entrera la commande :

```
find . -name "*.txt" -print
```


Exercice n° 9

1. À l'aide de la commande `find`, trouvez et affichez tous les fichiers `"hough.txt"` qui sont présents dans le dossier `"ressources-odl-fisa/tp01/rep/"`, ainsi que ses sous-dossiers.
2. Même chose pour le fichier `"loremipsum.txt"`.
3. À présent, trouvez et affichez uniquement tous les fichiers `"loremipsum.txt"` dont la taille dépasse 3 ko.

Chercher et extraire du texte, compter

Vous allez mettre en oeuvre les commandes suivantes :

- **grep** qui permet de filtrer l'ensemble des lignes d'un ou plusieurs fichiers qui contiennent un motif particulier ;
- **tail** et **head** qui permettent d'afficher respectivement les dernières et premières lignes d'un fichier.
- **wc** qui réalise un comptage des lignes, mots et octets d'un texte.

Exercice n° 10

1. Copiez le fichier `"chateaubriand-memoires-outr-tombe.txt"` dans votre dossier de travail (vous le trouverez dans le dossier `"ressources-odl-fisa/tp01/"`).
2. Affichez les lignes du texte de Chateaubriand où apparaît le nom `"Saint-Malo"`.
3. Est-il fait mention de la ville de `"Combourg"` dans le texte de Chateaubriand ?
4. Affichez les 10 premières lignes du texte de Chateaubriand.
5. Affichez les 15 dernières lignes du texte.
6. Combien de lignes et de mots sont contenus dans le texte ?

Enchaînement de commandes et tubes

Si nous souhaitons visualiser le nombre de fichiers et de dossiers contenus dans notre dossier personnel, nous pouvons utiliser des redirections vers un fichier temporaire, et exécuter l'enchaînement de commandes suivant :

```
ls ~ > temp ; wc -l temp ; rm temp
```

L'exemple précédent peut être simplifié à l'aide du mécanisme appelé **tube**¹ (*pipe* en anglais). Un tube permet de connecter la sortie standard d'une commande à l'entrée standard d'une autre sans passer par un fichier intermédiaire. Le caractère `|` représente un tube. L'exemple précédent devient alors :

```
ls ~ | wc -l
```

Exercice n° 11

1. Que retourne la commande `ls | wc -l` ?
2. En utilisant les commandes `tail`, `head` et un tube, affichez la 27e ligne du texte de Stendhal en partant de sa fin.
3. Comptez le nombre de mots de la ligne récupérée à la question précédente.
4. À l'aide des commandes `grep`, `tail` et d'un tube, affichez la dernière ligne du texte de Stendhal à contenir le prénom "Julien".

Exercice n° 12

Dans cet exercice, vous allez mettre en oeuvre les commandes **cut** (couper du texte), **sort** (trier des lignes de texte par ordre alphabétique ou numérique), **tr** (transposer ou éliminer des caractères) et **uniq** (supprimer les occurrences multiples).

1. Copiez dans le dossier de travail le fichier "sales2009.csv" qui est disponible dans le dossier "ressources-odl-fisa/tp01/". Ce fichier contient la liste des locations saisonnières réalisées en janvier 2009 par une agence immobilière de Floride aux USA.
2. Extrayez le nom des individus ayant traité les transactions (champ "Name") ainsi que leur montant (champ "Price"), puis triez le résultat suivant les montants croissants, et enfin redirigez le tout vers le fichier "results.txt".
3. Quelle personne a réalisé le plus de transactions ? Combien ?
4. Quelle carte bancaire a été la plus utilisée et combien de fois ?

Comparer des fichiers

La commande **diff** compare les fichiers texte (nous allons la retrouver à plusieurs reprises dans les différents TP).

¹Nous verrons plus en détail ce mécanisme dans le cours de systèmes d'exploitation en 2e année.

Exercice n° 13

1. Copiez dans votre dossier de travail les fichiers "zweig-marie-stuart-1.txt" et "zweig-marie-stuart-2.txt" que vous trouverez dans "ressources_tp01".
2. Sans visualiser le contenu des fichiers, affichez les différences entre les deux fichiers à l'aide de la commande `diff`. Que devez-vous modifier dans "zweig-marie-stuart-2.txt" pour corriger les différences ? Cherchez l'option de `diff` qui place les fichiers côte à côte pour la comparaison. Quelle option choisir de manière à éviter que `diff` ne prenne en compte des différences liées aux espaces ?
3. Vérifiez votre analyse en visualisant les fichiers.

Personnaliser son invite de commande primaire

La variable d'environnement `PS1` spécifie le format de l'invite de commande primaire du **shell**. Pour en visualiser le contenu, vous pouvez entrer la commande : `echo $PS1`. Sur vos systèmes, vous devriez avoir quelque chose du genre `\u@\h:\w\$`, avec :

- `\u` : identifiant de l'utilisateur
- `\h` : nom de la machine
- `\w` : chemin absolu depuis le dossier personnel
- `\$` : caractère \$ (sauf pour le super-utilisateur qui verra un #)

Exercice n° 14

Comme vous pouvez le remarquer, si votre arborescence est profonde alors l'invite de commande primaire par défaut de votre **shell** peut devenir très longue et gêner la lisibilité lorsque vous entrez vos commandes.

1. Modifiez `PS1` de telle manière à ce que l'invite de commande primaire soit composée d'un chevron supérieur suivi d'un espace : `>`.
2. Transformez `PS1` afin d'avoir une invite de commande sur deux lignes, la première indiquant le chemin absolu entre crochets (par exemple `[~/Documents/odl]`) et celle du dessous spécifiant l'utilisateur suivi des caractères `"->"` (par exemple : `jsaigne->`).
3. Pour les initiés, améliorez l'invite de commande en ajoutant éventuellement des couleurs.



Afin que votre invite devienne permanente, vous penserez au début du TP n°2 à l'ajouter dans votre fichier ".profile". N'oubliez pas, comme toute variable d'environnement, de l'exporter : `export PS1="..."`.

Mettre en oeuvre **tmux**

Le multiplexeur de terminaux **tmux** permet de lancer plusieurs terminaux virtuels dans une fenêtre de terminal.

Exercice n° 15

1. Lancez le programme **tmux**.
2. Renommez la fenêtre active de la session par défaut en "tp01".
3. Décomposez la fenêtre en deux panneaux horizontaux, puis réduisez le panneau du bas de 8 cellules (`Ctrl + b` `:`), puis à l'invite, entrez "`resize-pane -D 8`". Dans ce panneau du bas, lancez la commande `top` qui permet de surveiller les processus actifs et gourmands².
4. À présent, décomposez le panneau du haut en deux panneaux verticaux de même taille. Le panneau de gauche sera utilisé pour le client SSH³ de l'exercice suivant, et celui de droite pour les commandes locales.

Surtout, ne fermez aucun panneau de **tmux**, nous allons nous en servir dans l'exercice suivant.

Accéder à un serveur distant

Dans les deux exercices qui suivent, vous allez mettre en oeuvre les commandes **ssh** et **curl** pour accéder à des serveurs distants, ainsi que **whoami** et **who** afin de relever les utilisateurs connectés aux machines.

Exercice n° 16

La commande **ssh** permet de vous connecter à un serveur distant, et ce, de manière sécurisée. Le serveur Cybèle par exemple gère la connexion à votre compte ENSICAEN.

²À l'avenir, vous préférerez le programme `htop` pour surveiller les processus et leur utilisation par les coeurs du processeur.

³Protocole de connexion sécurisée (**Secured SHell**).

1. Dans le panneau en haut et à gauche créé précédemment sous **tmux**, connectez-vous à votre compte "distant" :
ssh <votre identifiant>@cybele.ecole.ensicaen.fr
Une fois connecté, assurez-vous que vous êtes bien dans votre dossier personnel, puis créez le sous-dossier "doc" dans votre dossier "tp01". Vérifiez dans le panneau de droite de **tmux** (connexion locale) que vous voyez bien apparaître ce dossier "doc".
2. Dans chacun des deux panneaux, établissez qui est l'utilisateur courant et qui sont éventuellement les autres utilisateurs connectés.
3. Dans le panneau de gauche, clôturez la connexion avec le serveur Cybèle (commande `exit` ou `Ctrl+d`), puis fermez le panneau gauche de **tmux**.

Exercice n° 17

curl est une commande très puissante qui permet d'envoyer des requêtes et de recevoir des données sur le réseau. Elle supporte la plupart des protocoles Internet tels que HTTP, SMTP, POP3, etc. Par exemple, la commande suivante récupère le code de la page d'accueil du serveur Web de l'ENSICAEN : `curl http://www.ecole.ensicaen.fr`.

1. Vérifier avec votre navigateur que le site <http://wttr.in> vous retourne les prévisions météorologiques locales sur trois jours.
2. Réalisez à l'aide de la commande `curl` la même opération depuis le panneau haut du terminal (éventuellement en élargissant la fenêtre). Plutôt sympathique, n'est-ce pas ?

Allez ! Je vous donne encore un aperçu de ce que vous pourriez faire avec `curl` :

```
# Un accès à des mémos en ligne :
$ curl https://cht.sh/<commande Unix>
# La dernière vidéo mise en ligne sur une chaîne YT que vous suivez,
# ici la chaîne "The coding train" sur la programmation et l'IA :
$ curl -s https://decapi.me/youtube/latest_video?user\=shiffman
# Le dernier "tweet" du youtubeur précédent :
$ curl -s https://decapi.me/twitter/latest?name\=thecodingtrain
# etc.
```

Où et laquelle ?

Si vous êtes arrivé jusqu'ici, vous devez avoir à présent une idée un peu plus claire sur les commandes (commandes du système Unix en particulier), ainsi que la manière dont est structurée l'arborescence du système.

Les commandes Unix sont pour la plupart stockées dans le dossier `/bin` et pour certaines dans le dossier `/sbin`. Par contre, les programmes seront placés à d'autres endroits (par exemple `/usr/bin`, etc.).

La commande **whereis** retourne le(s) chemin(s) absolu(s) des fichiers (exécutables, codes sources, pages de manuel) qui concernent une commande Unix ou un programme référencé. La commande **which** quant à elle affiche le chemin absolu de la commande ou du programme référencé.

Exercice n° 18

1. À l'aide des commandes `whereis` et `which`, afficher les fichiers et le chemin absolu de la commande `printf`.
2. Faites de même avec la commande `mkdir`, le programme `vim`, ainsi qu'avec la commande qui permet de lancer votre navigateur Firefox : `firefox`.

Historique des commandes

Les interpréteurs de commande conservent un historique des commandes que vous tapez, en général dans un fichier caché à la racine de votre dossier personnel. L'interpréteur **bash** utilise le fichier `.bash_history`.

Bien que cet historique vous soit propre, l'administrateur **root** peut tout à fait accéder à un historique global des différentes commandes entrées sur une machine. Il est donc utile de savoir effacer tout ou partie du contenu de ce fichier, par exemple les commandes où l'on aurait malencontreusement tapé un mot de passe en clair.

Exercice n° 19

La commande **history** affiche l'ensemble des commandes entrées dans le terminal. Elle propose aussi des options qui permettent de manipuler le fichier `.bash_history`.

1. Enregistrez une copie de votre fichier `.bash_history`.
2. Effacez la 12e commande que vous avez tapée et vérifiez qu'elle n'est plus disponible en parcourant l'historique avec les flèches du clavier.
3. Restaurez l'ancien historique et utilisez un des moyens vus précédemment afin de générer à partir du fichier `.bash_history` le fichier `tp01-nom1-nom2.txt` que vous placerez dans

votre dossier de travail.

4. Fermez successivement tous les panneaux de **tmux** à l'aide de la commande `exit`.

Un dernier effort

La configuration par défaut de **tmux** est plus que minimale. Afin de modifier cette configuration, il faut créer dans votre dossier personnel un fichier `~/.tmux.conf` et éventuellement un fichier `~/.tmux.local.conf` qui viendra surcharger le précédent sans le modifier.

Les fichiers que nous vous proposons d'utiliser sont ceux de Gregory Pakosz. Puisque ces derniers sont stockés sur son dépôt **Github** et qu'à présent vous connaissez la commande `curl`, vous allez entrer les commandes suivantes :

```
$ curl -o ~/.tmux.conf -LJO  
  ↪ https://raw.githubusercontent.com/gpakosz/.tmux/master/.tmux.conf  
$ curl -o ~/.tmux.conf.local -LJO  
  ↪ https://raw.githubusercontent.com/gpakosz/.tmux/master/.tmux.conf.local
```

Relancez **tmux**. Outre le nouveau visuel, cette fois-ci c'est un appui sur les touches `Ctrl` + `a` qui va vous permettre de gérer **tmux** (elles ne nécessitent qu'une main en principe). Intéressant non ?

Livable

Vérifiez que les fichiers d'animation ("`tp01-nom1-nom2-ex*.cast`") sont valides en initiant leur lecture à l'aide de la commande `asciinema play` (pas la peine de les lire en entier !)

Déplacez alors l'ensemble des fichiers `*.cast` vers le sous-dossier `anim` que vous aurez préalablement créé, puis compressez ce dossier à l'aide de la commande **zip**. Déposez l'archive compressée sur Moodle sur vos comptes.

Résumé

Dans ce TP vous avez découvert quelques commandes Unix, en particulier : `alias`, `cat`, `cd`, `chmod`, `clear`, `cp`, `curl`, `date`, `du`, `diff`, `file`, `find`, `grep`, `head`, `history`, `less`, `ln`, `ls`, `man`,

`mkdir, more, mv, pwd, rmdir, rm, sort, ssh, tail, tar, uniq, wc, whereis, which, who, whoami` et `zip`.

De plus, vous avez mis en oeuvre les mécanismes de redirection, d'enchaînement des commandes et utilisé des tubes. Vous avez aussi personnalisé votre invite de commande primaire.

Enfin, vous avez mis en oeuvre un multiplexeur de terminaux, ***tmux*** en l'occurrence, qui vous permettra à l'avenir de transformer votre terminal en véritable tableau de bord.