

Notes sur les TP système et accueil système

Ecole polytechnique de l'université de Nantes

- INFO3 - 2021/2022

Enseignants :

1 Evaluation des TP à partir d'un cahier de TP

Vous devez tenir à jour un cahier de TP par binôme.

Sur ce cahier manuscrit, vous devrez noter lors des séances de TP :

- les compléments d'information indispensables apportés par les enseignants;
- les résultats obtenus en TP, votre démarche, vos explications et des exemples complémentaires;
- vos remarques personnelles et les difficultés rencontrées.

Les notes que vous prendrez sur ce cahier doivent être propres et claires.

Vous devez prendre vos notes "à chaud".

Chaque séance, un ou plusieurs cahiers de TP seront relevés et notés par une appréciation (TB, B, AB, P, I, TD).

Il ne sera pas demandé de rédiger de compte-rendu de TP, puisque votre cahier de TP doit permettre d'évaluer votre travail.

2 Correction des TP

Il ne sera pas distribué de correction des TP.

Les enseignants qui vous encadrent sont là pour répondre à vos questions.

TP 1

Unix, le système de fichier

Enseignants :

Objectif du TP : Initiation aux commandes principales d'unix.

A chaque nouvelle commande unix, vous devrez :

- faire systématiquement appel au manuel en ligne ;
- noter les principales options ;
- noter des exemples personnels pour chaque commande et les résultats obtenus.

Conseil : utilisez le support « guide unix utilisateur » et ses exemples

1 Documentation : manuel en ligne

Essayez les commandes `man`, `xman` et `whatis`, `apropos`. (ex: `man man`, `whatis man`, `apropos man`).

Quels sont les différents paragraphes du manuel décrivant une commande ?

Quels sont les différents niveaux du manuel ?

Recherchez (tapez `h` pour l'aide en ligne du manuel) puis essayez les possibilités de recherche automatique d'une séquence de caractères dans une page de manuel. (ex : `/option` puis `n N`).

2 Listage du système de fichiers : commande ls

Repérer et noter les différents champs produits par la commande `ls -l`. Que signifie le premier caractère ?

Quel est votre nom d'utilisateur et votre groupe (vérifier avec `id`, `whoami`) ?

Comparez `ls -a -l` et `ls -al`. Noter vos remarques.

Quels sont les répertoires du premier niveau sous la racine / ?

Repérez différents types de fichiers : répertoires, liens, fichiers de données, exécutable, sockets, tubes, pilotes (regardez dans les dossiers `/etc/dev` / `etc` / `usr` / `lib`).

Combien il y a-t-il d'arborescences de fichiers sous unix ?

Qu'est-ce qu'un fichier caché ? Il y en a-t-il sous votre répertoire de connexion ?

Que signifient les fichiers `.et . .` ? Combien il y a-t-il de lien (en dur) sur ces 2 dossiers ?

Quel est l'intérêt de l'option `-F` ? de l'option `--color` ?

Qu'est-ce qu'un lien symbolique (ou raccourci ou renvoi) ? En voyez-vous ?

Que signifie un caractère `.` (point) dans un nom de fichier ?

Comparez les commandes : `ls -l .` , et : `ls -ld . .`

3 Changement de répertoire

Toute commande unix s'exécute dans le contexte d'un répertoire courant (ou de travail, working directory). La commande `pwd` affiche ce répertoire.

Quel est votre répertoire de connexion (ou home directory) ?

Que signifie le dossier abrégé `~` ?

Quelle commande permet de changer de répertoire courant ?

Déplacez-vous à la racine du système de fichier.

Comment revenir simplement à votre répertoire de connexion ?

Quelle commande permet de se déplacer simplement vers le répertoire père du répertoire courant ? Vers son grand père ?

4 Caractères spéciaux pour la génération de noms de fichiers

Quels caractères spéciaux (ou métacaractères) peuvent être placés dans les noms de fichiers (en Bourne Shell)? Que signifient-ils? (voir le manuel `man bash` section `Pathname Expansion` ou `Pattern Matching`, utilisez la recherche par expression régulière dans le manuel `man` avec : `/^ *Pattern Matching puis n et N`)

Essayez des commandes utilisant ces caractères spéciaux. Notez-les, ainsi que les résultats obtenus.

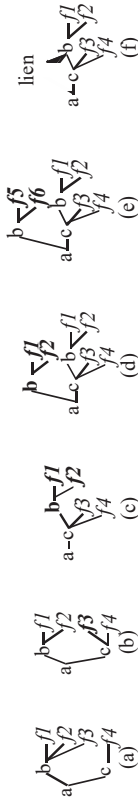
Est-il possible de créer un fichier nommé `*--?danger` ? Est-ce souhaitable ?

5 Création et suppression de fichiers et de répertoires

A quoi servent les commandes `touch` et `mkdir` ?

Que fait la commande `mkdir -p rep1/rep2/rep3` ?

Donnez les suites de commandes permettant de produire successivement :



C'est-à-dire : (a) créer la structure, (b) déplacer le fichier `f3`, (c) déplacer la branche `b`, (d) dupliquer la branche `b`, (e) renommer `f1` et `f2`, (f) supprimer la seconde branche `b` puis la remplacer par un lien symbolique vers la première branche `b`. (Chaque nouvelle commande doit être décrite avec ses options principales).

Quelles commandes acceptent les options `-i` `-f` et `-r` et `-v` ? Quel en est l'intérêt ?

Que se passe-t-il si, dans la situation (a), on lance la commande `mv a/b/£3 a/c/£4` ? Comment y remédier ?

Comparez les cas (d) et (f).

6 Nature d'un fichier

A quoi sert la commande `file` ? Donnez des exemples (`file *` , `file .` , ...)

NB : `bash` (mais pas seulement), on peut Utiliser l'historique, le rappel de commandes, la complétion de noms. Essayez `history`, `!!`, `!10`, et les touches `<tab>` ou `<esc>` après les premiers caractères d'un nom de fichier, ou encore les flèches haut et bas pour naviguer dans l'historique des commandes

Commandes utilisateur Unix - gestion des droits, redirections, recherche....

Enseignants :

Objectif du TP : Initiation aux commandes principales d'Unix.

A chaque nouvelle commande Unix, vous devrez : faire appel au manuel en ligne, noter les principales options, ainsi que des exemples personnels pour chaque commande et les résultats obtenus.

Il est conseillé de vous aider du *guide unix utilisateur*.

1 Gestion des droits des fichiers

Quelles sont les commandes permettant de voir et de modifier les droits (protection, permissions, autorisations) appliqués aux fichiers ?

Quels sont les droits accessibles par la commande `ls` ? Détailler les champs obtenus. Illustrer par des exemples trouvés dans le système.

Expliquez les modes symboliques et numériques.

Quels sont les droits du répertoire racine ? Pouvez-vous y créer un répertoire ?

Quels sont les droits de votre répertoire de connexion ?

Créez dans le dossier `/tmp` un lien symbolique vers le fichier `/etc/passwd`. Quels sont les droits de ce lien ? Pouvez-vous les changer ? Détruisez ce lien.

Pouvez-vous lire les fichiers contenus dans les répertoires de connexion d'autres utilisateurs ?

Créez un répertoire `Dossier` dans votre répertoire racine. Quels sont ses droits par défaut ? Donnez lui les permissions `rw-r--r--x`.

Créez un fichier `fic` dans `Dossier`. Quels sont ses droits par défaut ? Donnez-lui les permissions `rw-r--r--`.

Essayez de créer ou d'effacer le fichier `fic` après avoir placé successivement sur votre répertoire `Dossier` les droits : `rw-r--r--x`, `rw-r--x--x`, `rw-r--x--x`, `r--x--x--x`, `r--x--x--x`, `--x--x--x`. Expliquez ce qui se passe.

Quel est le résultat de la commande `umask` ? Expliquez.

Modifiez la valeur de `umask`, puis créez un fichier et un dossier. Visualisez leurs droits. Expliquez.

Pouvez-vous changer le propriétaire d'un fichier ? Qui peut le faire ? Expliquez.

2 Montages de systèmes de fichiers

Expliquez les commandes `df` et `mount` et les résultats obtenus, en particulier pour `df` les colonnes `Filesystem` et `Mounted on`. Expliquez la notion de greffe et son intérêt. Allez voir les fichiers `/etc/fstab` et `/etc/mtab`. Regardez les fichiers mentionnés localisés dans `/dev` (ex : `ls -l /dev/sda1`). Quelles sont les ressources locales, les ressources distantes ?

3 Listage du contenu d'un fichier de données

Essayez les commandes `cat`, `more` (ou `less`). Utilisez les fichiers `/etc/passwd` et `/etc/group`.

Évaluez les possibilités de recherche de la commande `more` (ex : `/regex` puis `n N`).

Essayez ces commandes avec plusieurs fichiers en argument.

4 Mécanismes de redirection

Expliquez les mécanismes de redirection enclenchés par les caractères (consulter la section REDIRECTION du manuel `bash`) :

```
Essayez :
ls -l > ls-fic
cat ls-fic
sort < ls-fic
ls -a >> ls-fic
ls -l /usr/bin | more
ls -ld $(pwd)
ls -ld $(pwd)
```

Donnez d'autres exemples permettant d'évaluer toutes les redirections.

Essayez de concaténer deux fichiers en un troisième (cf `cat`).

Que fait la commande `xargs` ? Donnez un exemple montrant l'intérêt de cette commande.

Proposez une suite de commandes produisant un résultat identique à « `ls -l /usr/bin | more` », sans Utilisez de `conduit`. Quel est l'intérêt du conduit (en temps d'exécution) ?

Que fait `ls -l > /dev/null` ? Expliquer. Quels sont les permissions et la nature de `/dev/null`.

Comparez `ls -l > /etc/passwd`

`et ls -l > /etc/passwd 2>/dev/null ou ls -l 2>/dev/null > /etc/passwd (en bash)`

5 Recherche d'un fichier dans une branche : find

- Options `-name` et `-print`

Recherchez dans `/usr` où se trouve la commande `find`.

Recherchez dans `/usr/bin` les fichiers commençant par `p`. Recherchez dans `/usr/bin` les fichiers d'une lettre puis de trois lettres. Recherchez dans `/usr/bin` les fichiers finissant par `h` ou `c`.

- Options `-time`

Cherchez dans votre répertoire de connexion, les fichiers modifiés dans les deux derniers jours.

- Options `-type`

Cherchez les dossiers dans votre répertoire de connexion. Cherchez dans le système local les fichiers sockets, les tubes.

- Options `-prune`

Cherchez les dossiers dans votre répertoire de connexion. Cherchez à partir de la racine / en évitant les dossiers inutiles (`/root /mnt/trsbetu/Exxx`) pour trouver les fichiers sockets, les tubes.

- Options `-user` avec droits symboliques ou octaux

Cherchez dans le système les fichiers dont vous êtes propriétaire.

- Options `-perm`

Cherchez dans le système les dossiers dans lesquels vous pouvez écrire. Recherchez, dans votre répertoire de connexion, les fichiers exécutables. Recherchez dans votre répertoire de connexion, les fichiers que tout le monde peut lire.

Cherchez dans le système les fichiers possédant un sticky bit (t). Cherchez dans le système les fichiers possédant un setuid ou un setgid bit (s).

- Options `-exec` `-ok`

Appliquez automatiquement la commande `ls -l` à tous vos fichiers de donnée non modifiés depuis au moins 15 jours.

Appliquez automatiquement, avec demande de confirmation, la commande `rm` à tous vos fichiers de données non modifiés depuis au moins 15 jours.

- Options `-regex` ...

Allez plus loin avec les critères booléens en consultant `man find` et/ou le « guide unix utilisateur » paragraphe 18.1 page 29

6 Recherche d'une expression dans un fichier : grep

Essayez `grep root /etc/passwd`.

Proposez une commande donnant le numéro (rien de plus) d'un utilisateur donné, par recherche dans `/etc/passwd`.

Idem pour le numéro de groupe d'un groupe donné.

Recherchez dans votre répertoire de connexion, les lignes de la commande `ls -l` donnant des fichiers du jour.

Les expressions régulières sont utilisées dans de nombreuses commandes unix (`grep`, `sed`, `ed`, `vi`, `emacs`, `find`, `expr`, `awk`, `perl`, etc...) et la plupart des langages de programmation.

Allez plus loin avec les expressions régulières en essayant les exemples du *guide unix utilisateur* paragraphe 18.2 page 31, et en consultant par exemple https://en.wikipedia.org/wiki/Regular_expression

Enfin tester vos expressions régulières (par exemple sur un site <https://regexr.com/> ou <https://www.regextester.com/>)

Quelques regex : '[a-z] +', '[A-Z][A-Za-Z]*', '[^] +', '[^0-9]\(3\)', ':[0-9]\(2,\)\:', '\.s', '(une(deux) fois)', '([a-z]+) \1'

Caractère	Signification
[...]	Plage de caractères permis.
[^...]	Plage de caractères interdits.
^	Début de ligne.
\$	Fin de ligne.
.	Tout caractère unique, y compris un espace.
*	Répétition du caractère ou expression placé devant
\N\	Répétition de N fois exactement.
\N{,}	Répétition de N fois au minimum.
\N1 N2\	Répétition de N1 à N2 fois

Expressions régulières « simples »

il existe des classes de caractères prédéfinies : [:alnum:], [:alpha:], [:cntrl:], [:digit:], [:graph:], [:lower:], [:print:], [:punct:], [:space:], [:upper:], [:xdigit:], \w, \W, etc...

Par exemple [:alnum:] correspond à [0-9A-Za-z]

Les positionnements ^ \$ peuvent être étendus par > < \b \B

Par défaut les expressions régulières sont simples. Utilisez de préférence les expressions étendues (ex : grep -E)

Dans les expressions régulières simples, les caractères spéciaux ?, +, *, {, |, (,) perdent leurs significations d'opérateur, il faut utiliser à la place leurs versions avec BackSlash \?, \+, *, \{, \|, \(, et \) .

Dans egrep le méta-caractère (perd sa signification spéciale, il faut Utiliser \ (à la place

Caractère	Signification
+	répétition. au minimum une fois.
?	répétition. Au plus 1 fois.
(a b)	L'une ou l'autre des expressions a et b
(...)	Sous expression régulière.
\N	Référence arrière à la Nième sous expression

Expressions régulières « étendus »