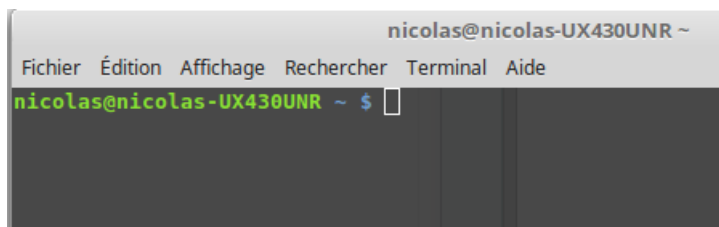


Les tous premiers systèmes d'exploitation, ces derniers étaient dépourvus d'interface graphique (système de fenêtres "pilotables" à la souris), toutes les interactions "système d'exploitation - utilisateur" se faisaient par l'intermédiaire de "lignes de commandes" (suites de caractères saisies par l'utilisateur).

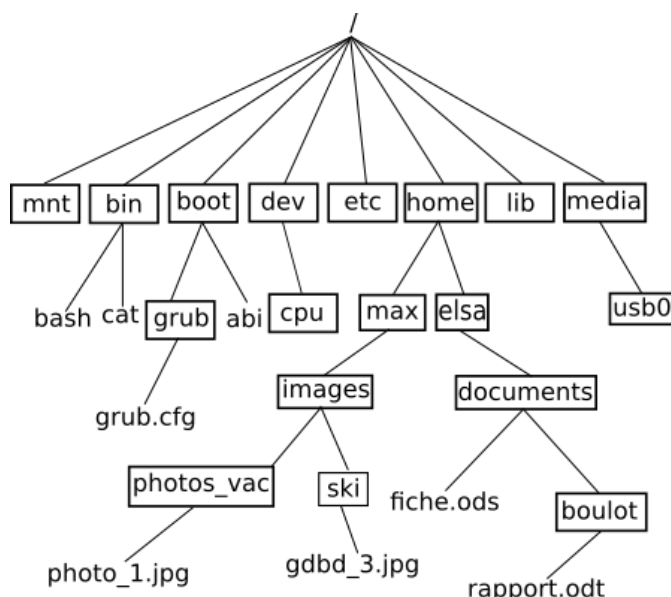
Aujourd'hui, même si les interfaces graphiques modernes permettent d'effectuer la plupart des opérations, il est important de connaître quelques-unes de ces lignes de commandes.

Pour saisir des lignes de commandes, nous allons utiliser une console. Ouvrez une console et vous devez obtenir quelque chose qui ressemble à cela :



Nous avons ci-dessus la console de l'utilisateur "nicolas" qui utilise un ordinateur qui se nomme "nicolas-UX430UNR" ("nicolas@nicolas-UX430UNR").

Principalement nous allons, grâce à la ligne de commande, travailler sur les fichiers et les répertoires. Dans les systèmes de type "UNIX" (par exemple GNU/Linux ou macOS), nous avons un système de fichier en arborescence :



Dans le schéma ci-dessus on trouve des répertoires (noms entourés d'un rectangle, exemple : "home") et des fichiers (uniquement des noms "grub.cfg").

À noter : les extensions des noms de fichiers, par exemple le "cfg" de "grub.cfg", **ne sont pas obligatoires** dans les systèmes de type "UNIX", par exemple, "bash" est bien un nom de fichier et il n'a pas d'extension.

On parle d'arborescence, car ce système de fichier ressemble à un arbre à l'envers.

Comme vous pouvez le constater, la base de l'arbre s'appelle la racine de l'arborescence et se représente par un "/"

Chemin absolu ou chemin relatif?

Pour indiquer la position d'un fichier (ou d'un répertoire) dans l'arborescence, il existe 2 méthodes : indiquer un chemin absolu ou indiquer un chemin relatif. Le chemin absolu doit indiquer "le chemin" depuis la racine. Par exemple le chemin absolu du fichier fiche.ods sera : /home/elsa/documents/fiche.ods

Remarquez que nous démarrons bien de la racine / (attention les symboles de séparation sont aussi des /)

Il est possible d'indiquer le chemin non pas depuis la racine, mais depuis un répertoire quelconque, nous parlerons alors de chemin relatif :

Le chemin relatif permettant d'accéder au fichier "photo_1.jpg" depuis le répertoire "max" est : "images/photo_vac/photo_1.jpg"

Remarquez l'absence du / au début du chemin (c'est cela qui nous permettra de distinguer un chemin relatif et un chemin absolu).

Imaginons maintenant que nous désirions indiquer le chemin relatif pour accéder au fichier "gdbd_3.jpg" depuis le répertoire "photos_vac".

Comment faire ?

Il faut "remonter" d'un "niveau" dans l'arborescence pour se retrouver dans le répertoire "images" et ainsi pouvoir repartir vers la bonne "branche". Pour ce faire il faut utiliser 2 points : ..

"../ski/gdbd_3.jpg"

Il est tout à fait possible de remonter de plusieurs "crans" : "../.." depuis le répertoire "photos_vac" permet de "remonter" dans le répertoire "max".

Exercice 1 : En vous basant sur l'arborescence ci-dessus, déterminez le chemin absolu permettant d'accéder au fichier :

- "cat"

- "rapport.odt"

Toujours en vous basant sur l'arborescence ci-dessus, déterminez le chemin relatif permettant d'accéder au fichier :

- "rapport.odt" depuis le répertoire "elsa"

- "fiche.ods" depuis le répertoire "boulot"

Comme déjà évoqué plus haut, les systèmes de type "UNIX" sont des systèmes "multi-utilisateurs" : chaque utilisateur possède son propre compte. Chaque utilisateur possède un répertoire à son nom, ces répertoires personnels se situent traditionnellement dans le répertoire "home". Dans l'arborescence ci-dessus, nous avons 2 utilisateurs : "max" et "elsa". Par défaut, quand un utilisateur ouvre une console, il se trouve dans son répertoire personnel. Dans l'image de la console ci-dessus, nous avons un "nicolas@nicolas-UX430UNR ~ \$" (au passage, on appelle cela "l'invite de commande"), le "~" (caractère "tilde") signifie que l'on se trouve actuellement dans le répertoire personnel de l'utilisateur courant, autrement dit dans le répertoire de chemin absolu "/home/nicolas" (puisque l'utilisateur courant est "nicolas"). Le répertoire "où l'on se trouve actuellement" est appelé "répertoire courant". L'invite de commande vous indique à tout moment le répertoire courant : "nicolas@nicolas-UX430UNR ~/Documents \$" vous indique que vous êtes dans le répertoire "Documents" qui se trouve dans le répertoire "nicolas" qui se trouve dans le répertoire "home" (chemin absolu : "/home/nicolas/Documents")

Attention : les systèmes de type "UNIX" sont "sensibles à la casse" (il faut différencier les caractères majuscules et les caractères minuscules) : le répertoire "Documents" et le répertoire "documents" sont 2 répertoires différents.

La commande cd

La commande "cd" permet de changer le répertoire courant. Il suffit d'indiquer le chemin (relatif ou absolu) qui permet d'atteindre le nouveau répertoire :

Par exemple (en utilisant l'arborescence ci-dessus) :

- si le répertoire courant est le répertoire "elsa" et que vous "voulez vous rendre" dans le répertoire "documents", il faudra saisir la commande : "cd documents" (relatif) ou "cd /home/elsa/documents" (absolu)
- si le répertoire courant est le répertoire "photos_vac" et que vous "voulez vous rendre" dans le répertoire "ski", il faudra saisir la commande : "cd ../ski" (relatif) ou "cd /home/max/images/ski" (absolu)
- si le répertoire courant est le répertoire "boulot" et que vous "voulez vous rendre" dans le répertoire "documents", il faudra saisir la commande : "cd .." (relatif) ou "cd /home/elsa/documents" (absolu)

Exercice 2 : Toujours en utilisant l'arborescence ci-dessus, quelle est la commande à saisir si le répertoire courant est le répertoire "home" et que vous "voulez vous rendre" dans le répertoire "boulot" (vous utiliserez d'abord un chemin absolu puis un chemin relatif)

La commande ls

La commande "ls" permet de lister le contenu du répertoire courant.

Dans l'exemple ci-dessus, depuis le répertoire personnel de l'utilisateur "nicolas", nous passons dans le répertoire "nsi" à l'aide d'un "cd nsi", puis nous affichons le contenu de ce répertoire "nsi" à l'aide de la commande "ls". Nous trouvons dans le répertoire "nsi" : 2 fichiers ("fiche1.odt" et "photo.jpg") et un répertoire ("test").

```
nicolas@nicolas-UX430UNR ~ $ cd nsi
nicolas@nicolas-UX430UNR ~/nsi $ ls
fiche1.odt photo.jpg test
nicolas@nicolas-UX430UNR ~/nsi $
```

Exercice 3 : Après avoir ouvert une console, utilisez la commande ls depuis votre répertoire personnel.

La commande "mkdir"

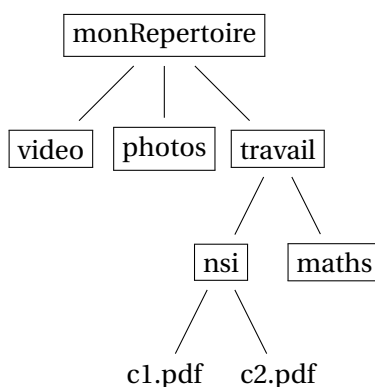
La commande "mkdir" permet de créer un répertoire dans le répertoire courant.

La commande est de la forme "mkdir nom_du_répertoire"

La commande "touch"

La commande "touch" permet de créer un fichier vide. La commande est de la forme "touch nom_du_fichier_à_créer"

Exercice 4 : A l'aide des deux commandes précédentes créer l'arborescence suivante où les dossiers sont encadrés et les fichiers non encadrés. On partira de votre répertoire courant (répertoire par défaut à l'ouverture de la console représenté par monRepertoire).



La commande "cp"

La commande "cp" permet de copier un fichier.

La commande est de la forme "cp /répertoire_source/nom_fichier_à_copier /répertoire_destination/nom_fichier"

À noter : le nom du fichier "destination" n'est pas obligatoirement le même que le nom du fichier "source" (on peut avoir "cp fic.txt info/fiche.txt")

Exercice 5 : Placez-vous dans le répertoire "travail" créé précédemment. Créez un fichier "test.txt". Donner la commande :

Créez un répertoire "doc". Copiez le fichier "test.txt" dans le répertoire "doc".

Donner la commande :

La commande "rm"

La commande "rm" permet de supprimer un fichier ou un répertoire.

La commande est de la forme "rm nom_du_répertoire_ou_nom_du_fichier"

La plupart des commandes UNIX peuvent être utilisées avec une ou des options. Par exemple, pour supprimer un répertoire non vide, il est nécessaire d'utiliser la commande "rm" avec l'option "-r" : "rm -r nom_du_répertoire"

Exercice 6 : Supprimer le répertoire nsi (et ce qu'il contient). Vérifier la suppression avec la commande ls.

Gestion des utilisateurs et des groupes

Les systèmes de type "UNIX" sont des systèmes multi-utilisateurs, plusieurs utilisateurs peuvent donc partager un même ordinateur, chaque utilisateur possédant un environnement de travail qui lui est propre.

Chaque utilisateur possède certains droits lui permettant d'effectuer certaines opérations et pas d'autres. Le système d'exploitation permet de gérer ces droits très finement. Un utilisateur un peu particulier est autorisé à modifier tous les droits : ce "super utilisateur" est appelé "administrateur" ou "root". L'administrateur pourra donc attribuer ou retirer des droits aux autres utilisateurs. Au lieu de gérer les utilisateurs un par un, il est possible de créer des groupes d'utilisateurs. L'administrateur attribue des droits à un groupe au lieu d'attribuer des droits particuliers à chaque utilisateur.

Comme nous venons de le voir, chaque utilisateur possède des droits qui lui ont été octroyés par le "super utilisateur". Nous nous intéresserons ici uniquement aux droits liés aux fichiers, mais vous devez savoir qu'il existe d'autres droits liés aux autres éléments du système d'exploitation ((imprimante, installation de logiciels...)).

Les fichiers et les répertoires possèdent 3 types de droits :

- les droits en lecture (symbolisés par la lettre r) : est-il possible de lire le contenu de ce fichier
- les droits en écriture (symbolisés par la lettre w) : est-il possible de modifier le contenu de ce fichier
- les droits en exécution (symbolisés par la lettre x) : est-il possible d'exécuter le contenu de ce fichier (quand le fichier du code exécutable)

Il existe 3 types d'utilisateurs pour un fichier ou un répertoire :

- le propriétaire du fichier (par défaut c'est la personne qui a créé le fichier), il est symbolisé par la lettre u
- un fichier est associé à un groupe, tous les utilisateurs appartenant à ce groupe possèdent des droits particuliers sur ce fichier. Le groupe est symbolisé par la lettre g
- tous les autres utilisateurs (ceux qui ne sont pas le propriétaire du fichier et qui n'appartiennent pas au groupe associé au fichier). Ces utilisateurs sont symbolisés la lettre "o"

Il est possible d'utiliser la commande "ls" avec l'option "-l" afin d'avoir des informations supplémentaires.

```
nicolas@nicolas-UX430UNR ~ $ cd nsi
nicolas@nicolas-UX430UNR ~/nsi $ ls -l
total 4
-rw-rw-r-- 1 nicolas nicolas  0 févr.  3 14:13 fichel.odt
-rw-rw-r-- 1 nicolas nicolas  0 févr.  3 14:13 photo.jpg
drwxrwxr-x 2 nicolas nicolas 4096 févr.  3 14:13 test
nicolas@nicolas-UX430UNR ~/nsi $
```

Prenons la première ligne :

Lisons cette ligne de gauche à droite :

- le premier symbole "-" signifie que l'on a affaire à un fichier, dans le cas d'un répertoire, nous aurions un "d" (voir la 3e ligne)
- les 3 symboles suivants "rw-" donnent les droits du propriétaire du fichier : lecture autorisée (r), écriture autorisée (w), exécution interdite (- à la place de x)
- les 3 symboles suivants "rw-" donnent les droits du groupe lié au fichier : lecture autorisée (r), écriture autorisée (w), exécution interdite (- à la place de x)
- les 3 symboles suivants "r - -" donnent les droits des autres utilisateurs : lecture autorisée (r), écriture interdite (- à la place de w), exécution interdite (- à la place de x)
- le caractère suivant "1" donne le nombre de liens (nous n'étudierons pas cette notion ici)
- le premier "nicolas" représente le nom du propriétaire du fichier
- le second "nicolas" représente le nom du groupe lié au fichier
- le "0" représente la taille du fichier en octet (ici notre fichier est vide)
- "févr. 3 14:13" donne la date et l'heure de la dernière modification du fichier
- "fichel.odt" est le nom du fichier

Exercice 7 : faites l'analyse de la troisième ligne en détaillant comme précédemment.

.....

.....

.....

.....

.....

Il est important de ne pas perdre de vue que l'utilisateur "root" a la possibilité de modifier les droits de tous les utilisateurs. Le propriétaire d'un fichier peut modifier les permissions d'un fichier ou d'un répertoire à l'aide de la commande "chmod". Pour utiliser cette commande, il est nécessaire de connaître certains symboles :

- les symboles liés aux utilisateurs : "u" correspond au propriétaire, "g" correspond au groupe lié au fichier (ou au répertoire), "o" correspond aux autres utilisateurs et "a" correspond à "tout le monde" (permet de modifier "u", "g" et "o" en même temps)
- les symboles liés à l'ajout ou la suppression des permissions : "+" on ajoute une permission, "-" on supprime une permission, "=" les permissions sont réinitialisées (permissions par défaut)
- les symboles liés aux permissions : "r" : lecture, "w" : écriture, "x" : exécution.

La commande "chmod" à cette forme :

```
chmod [u g o a] [+ - =] [r w x] nom_du_fichier
```

par exemple

```
chmod o+w pipo.txt
```

attribuera la permission "écriture" pour le fichier "pipo.txt" "aux autres utilisateurs"

Il est possible de combiner les symboles :

```
chmod g-wx pipo.txt
```

La commande "chmod" ci-dessus permet de supprimer la permission "écriture" et la permission "exécution" pour le fichier "pipo.txt" "au groupe lié au fichier"

Une fois de plus, "root" a tous les droits sur l'ensemble des fichiers et des répertoires, il peut donc utiliser la commande "chmod" sur tous les répertoires et tous les fichiers.

Exercice 8 : Créez un répertoire "testNSI" dans votre répertoire personnel. Placez-vous dans le répertoire "testNSI". Créez un fichier "blabla.txt", vérifiez les permissions associées à ce fichier.

Modifiez les permissions associées au fichier "blabla.txt" afin que les "autres utilisateurs" aient la permission "écriture"

Retour sur réseaux :

la commande "ifconfig"

Saisir la commande "ifconfig" est donner l'adresse IP de votre machine :

.....

La commande "ip"

On peut aussi obtenir l'adresse IP avec la commande "ip" suivie de "address". Faites l'essai.

La commande "ping"

Par défaut la commande rentre dans une boucle infinie, si l'on veut limiter le nombre de réponses on peut utiliser -c (count) avec le nombre de réponses attendues.

Saisir "ping -c 5 google.fr"