

Utiliser un système d'exploitation libre

Notion : Système d'exploitation

Operating System en anglais (OS),
c'est un ou

dont

le **but** est de
matérielles et logicielles d'un ordinateur. Il est
l' entre les programmes et le
matériel.

Quelques exemples d'OS actuels :

Pour ordinateurs	Pour smartphones
<input type="text"/>	<input type="text"/>

Un système d'exploitation est constitué :

d'un **noyau** (*kernel*) : fourni les
fonctionnalités basiques aux pro-
grammes. Consiste en du **code exécuté**
 et
dont le rôle est d'initialiser les périphériques
correctement.

→ le noyau ne contient ni programmes utilisateurs, ni interface (graphique ou texte)

→ un **ordinateur avec uniquement le noyau n'est pas utilisable en l'état**

Utilisateur

↕

Programmes

↕

Système d'exploitation

Pilotes	Ordonnanceur
Gestionnaire de mémoire	...
Systèmes de fichiers	Pile réseau

↕

Matériel

Parmi les différents composants logiciels d'un OS moderne on retrouve :

L'ordonnanceur

Le gestionnaire de mémoire

Les systèmes de fichier

La pile réseau

Les pilotes de périphérique

Notion : L'interface utilisateur du système d'exploitation

Abrévié **interface système** (ou *shell*), c'est un programme permettant à l'utilisateur d'interagir avec le système d'exploitation. Il peut-être de 2 formes :

Interface en ligne de commandes interface historique composée uniquement d'une **invite de commandes** et permettant d'interagir **uniquement en mode texte**

Interface graphique interface actuelle, avec un bureau virtuel ou un écran d'accueil

Ici on utilise un **logiciel graphique** appelé « émulateur de terminal » (ou juste « terminal » ou « console ») pour recréer l'interface en lignes de commandes.

Activité 1 Invités à commander l'OS



Utiliser l'invite de commande
Identifier les informations fournies dans l'invite de commande



- Ouvrez un émulateur de terminal et écrivez ci-dessous ce qui est écrit dans l'invite de commande. Identifiez votre nom d'utilisateur, le nom de votre machine (si indiqué) et votre localisation.

```
akobsch@Felidae ~ %
```

nom nom de la dossier = position
utilisateur machine courant = actuelle

Pour travailler tous avec le même ordinateur (les même dossiers et fichiers de départ), vous allez télécharger une copie d'une image disque d'un ordinateur fonctionnant sous le système d'exploitation Ubuntu (Linux). Pour cela exécutez chaque commande suivante l'une après l'autre :

`docker pull astranais/ubuntu-escapegame`

On récupère l'image stockée en ligne

`docker run -it --user eleve astranais/ubuntu-escapegame`

On lance ce mini ordinateur virtuel



Ces deux lignes (qui ne sont pas à savoir!) indiquent à l'OS que l'on souhaite utiliser le logiciel appelé docker pour exécuter d'abord la commande `pull`, puis la commande `run`. Ce qui suit consiste en les paramètres et/ou les options passées aux commandes.

- Recopiez ci-dessous l'invite de commande que vous obtenez. Quel est maintenant votre nom d'utilisateur ainsi que le nom de cette machine virtuelle ?
- Quel élément vous indique que vous ne vous trouvez **pas** dans votre dossier personnel ? (home)

Répertoire racine

Répertoire (dossier) contenant **tous** les autres (soit directement soit imbriqués dans d'autres répertoires). Il porte comme nom le symbole

La commande `pwd` (pour *print working directory*) affiche le **chemin absolu** du **répertoire courant** (celui où l'on se trouve).

Chemin absolu

Succession des dossiers à traverser en partant de la racine du disque dur de l'ordinateur pour arriver à la position actuelle



On peut **lister le contenu d'un répertoire** avec la commande `ls` (provient du mot *list*).

Pour se déplacer dans notre dossier personnel on peut utiliser la commande `cd` (pour *change directory*) toute seule, ou bien suivie d'un espace et du symbole `~` : `cd ~`

- Rendez-vous dans votre répertoire personnel puis écrivez ci-dessous le chemin absolu de ce répertoire. Complétez alors le schéma ci-dessus représentant **l'arborescence** des répertoires (l'organisation des dossiers dans votre machine).

Au fur et à mesure du TP, complétez le document « Commandes de base des systèmes POSIX ».

Activité 2 **Jeu de piste éducatif**

Vous voilà maintenant prêts à commencer ce jeu de piste qui vous guidera de salle en salle et vous fera découvrir les commandes utiles, ainsi que plein d'informations sur les systèmes d'exploitation !

1. Affichez le contenu de votre espace personnel. Un message de bienvenue a été laissé à votre intention, affichez-le avec la commande `cat` suivie d'un espace et du nom complet du fichier.

Partie I — Salle 1



Utiliser les commandes `ls` et `cd`
 Tracer (à la main) l'arborescence des fichiers et dossiers
 Identifier les composants principaux d'un OS



Vous entrez dans la salle et tout de suite une note manuscrite attire votre attention sur la table.

1. Rendez-vous dans le dossier **Table** et utilisez ce qui est écrit dessus pour dessiner ci-dessous l'arborescence de la **Salle_1** (ne représentez pas tous les fichiers de la bibliothèque).
 N'oubliez pas de consulter les fichiers que vous rencontrerez au passage !

2. Utilisez les informations du *tableau_ls* pour légender (au moins en partie) le résultat ci-dessous de la commande `ls -l` (vous aurez le reste des explications plus tard)

```
total 20
drwxr-xr-x  1 adminlocal visitor 4096 Apr 27 11:55 Bibliotheque
drwxr-xr-x  1 adminlocal admin  4096 Apr 27 11:35 Etagere
drwxr-xr-x  1 adminlocal admin  4096 Apr 27 10:45 Table
-rw-r--r--  1 adminlocal admin   109 Apr 27 11:19 horloge
-rw-r--r--  1 adminlocal admin   217 Apr 27 11:59 tableau_ls
```

3. Avec ce que vous trouverez dans la boîte posée sur l'étagère, complétez l'arborescence précédente.

Partie II — Salle 2

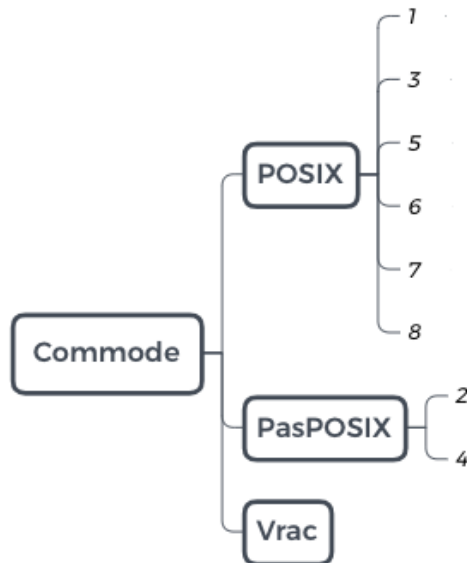


Créer des dossiers avec la commande `mkdir`
 Manipuler des fichiers avec les commandes `cp` et `mv`
 Identifier les différences entre les OS libres et propriétaires



Continuez à suivre les indices du jeu de piste et complétez les questions suivantes lorsque vous obtenez les informations dans le jeu.

1. Complétez l'arborescence ci-dessous que vous devez obtenir dans le jeu en mettant entre parenthèses à côté de chaque fichier, son contenu et la/les dates de développement du système d'exploitation concerné.



La différence la plus notable à notre niveau entre un système d'exploitation compatible avec la norme POSIX et un non compatible, est l'utilisation des instructions en lignes de commande. Elles seront sensiblement équivalente sur les systèmes POSIX mais pourront être très différentes d'un système non POSIX. Notons également que dans un système POSIX, l'arborescence des fichiers commencera toujours par le répertoire racine `/`.

2. Complétez le tableau suivant avec les 4 caractéristiques que doivent respecter un logiciel libre. Donnez au moins un exemple de système d'exploitation libre et un propriétaire. Est-ce qu'un logiciel libre est forcément gratuit ?

Les libertés logicielles

Document : Quitter son conteneur docker et le redémarrer plus tard

Pour sortir de son conteneur docker il suffit d'utiliser la commande `exit`.

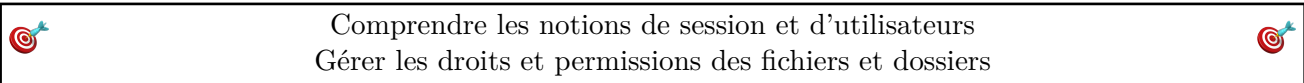
Pour retourner dans le même conteneur que précédemment quitté, afin de continuer le jeu exactement à l'endroit où vous vous êtes arrêtés vous devez taper les deux commandes suivantes :

<code>docker restart num_conteneur</code>	on redémarre le conteneur qui a pour numéro num_conteneur
-------------------------------------------	-----------------------------------------------------------

```
docker exec -it num_conteneur /bin/bash
```

Pour vérifier que son conteneur est bien encore enregistré sur l'ordinateur, affichez la liste des conteneurs avec la commande `docker ps -a`

Partie III — Salle 3



Notion : Utilisateurs et groupes

Les systèmes POSIX sont **multi-utilisateurs** : chaque utilisateur possède un identifiant de connexion (*login* en anglais) auquel est généralement associé un mot de passe afin d'authentifier l'utilisateur.

L'ensemble des données et fichiers de l'utilisateur constitue son **compte**. Lorsque l'utilisateur se connecte à son compte, le système démarre une interface utilisateur personnalisée.

L'ensemble des interactions de l'utilisateur authentifié avec le système est appelé une session. Lorsque l'utilisateur se déconnecte, il ferme sa session.

1. Quelle est la commande qui vous a été donnée à la fin de la Salle 2 et qui permet d'obtenir l'affichage ci-dessous ?

```
uid=1000(eleve) gid=1991(visitor) groups=1991(visitor)
```

2. En vous aidant de l'*ecriteau-porte*, légendez le résultat ci-dessus sachant que pour l'utilisateur adminlocal, la même commande donne le résultat ci-dessous :

```
uid=1001(adminlocal) gid=1000(admin) groups=1000(admin),27(sudo)
```

3. Tracez autant que possible l'arborescence de cette Salle 3.

4. Trouvez l'indice qui vous permettra de terminer de légender la capture d'écran de la question 2 Partie I.
5. Complétez le tableau ci-dessous avec le nom du propriétaire et du groupe propriétaire pour chaque dossier de la **Salle_3**. Cochez ensuite les cases pour chaque permission accordée dans chaque cas.

Dossier		Bureau	Coffre	Corbeille-papier	Vitrine
Propriétaire					
Groupe propriétaire					
Permissions du propriétaire	Lecture				
	Écriture				
	Exécution				
Permissions du groupe propriétaire	Lecture				
	Écriture				
	Exécution				
Permissions des autres utilisateurs	Lecture				
	Écriture				
	Exécution				

6. Tirez parti des obstacles rencontrés dans la question 3 et du tableau précédent pour en déduire la signification des permissions "Lecture" et "Exécution" pour un dossier.
7. À partir des indications que vous trouvez dans la Corbeille à papier, faites quelques tests dans la Corbeille (et en dehors) pour en déduire la signification des permissions "Lecture" et "Écriture" pour un fichier, ainsi que la permission "Écriture" pour un dossier.
8. La commande `chmod` ne fonctionne que sur deux éléments (fichier ou dossier). Lesquels et pourquoi ?
9. Donnez l'identifiant de connexion ainsi que le mot de passe associé du compte qui vous permet d'accéder aux dossiers et fichiers inaccessibles.
10. Où vous trouvez-vous lorsque vous venez de vous connecter à ce compte ? Indiquez ci-dessous le chemin absolu et complétez l'arborescence tracée dans la partie I.
11. Quel est le login du super-utilisateur et son uid et gid ?

Partie IV — Salle 4



Utiliser l'option `-r` des commandes `mv`, `cp` `rm` ;
 Comprendre et utiliser le mode octal des permissions
 Concaténer des fichiers avec `cat` et utiliser les motifs glob



1. D'après ce qui est écrit sur le tableau, à quoi sert la commande `rm *` ?
2. Quelle option faut-il ajouter pour pouvoir utiliser les commandes `rm`, `mv` ou `cp` sur des dossiers ?
3. Quelle est la commande si dangereuse citée dans l'affiche et quel a été son effet à votre avis sachant que l'option `f` signifie "forcer" (il n'y a aucune demande de confirmation) ?
4. Complétez le tableau ci-dessous avec le code octal correspondant aux permissions littérales ou inversement (n'hésitez pas à mettre des couleurs!).

Permission	<code>rw-rw-r--</code>		<code>rwX-W----</code>	
Mode octal		751		304

5. Complétez le texte suivant :

L'opérateur permet de sauvegarder dans un fichier le résultat d'une commande au lieu de l'afficher dans la console.

Ex. : `ls -l > resultat` sauvegarde dans le fichier le résultat de l'instruction . Si le fichier `test` existe déjà, il est écrasé.

L'opérateur permet d'ajouter le résultat à la suite du contenu du fichier de destination s'il existe déjà au lieu de l'écraser.

On peut rediriger le résultat d'une instruction vers l'entrée d'une autre instruction avec l'opérateur (*pipe* en anglais).

Ex. : `ls -l | sort` permet de passer en entrée de la commande le résultat de la commande , ce qui a pour effet de trier les lignes (selon la taille pour une liste de fichiers).

6. Complétez le tableau ci-dessous avec les symboles ou caractères utilisés pour constituer les motifs de base des expressions régulières :

Symboles	Signification
	n'importe quelle séquence de caractères
	un et un seul caractère (quel qu'il soit)
	représente un et un seul des caractères de la liste
	représente un et un seul caractère différent de ceux dans la liste
	représente un et un seul caractère compris dans l'intervalle entre crochets

Exercice 1 Comprendre des instructions

On suppose que le répertoire personnel de l'utilisateur courant est vide. Décrire l'effet de chacune des commandes suivantes, en supposant qu'elles sont exécutées les unes à la suite des autres.

1. `cd ~`
2. `mkdir NSI`
3. `mkdir NSI/TP_SHELL`
4. `cd NSI/TP_SHELL`
5. `cd ..`
6. `ls`
7. `chmod u+rwx,g-rwx,o-rwx TP_SHELL`

Exercice 2 Dessiner l'arborescence

On suppose que l'on se trouve dans un répertoire TEST, que ce dernier est vide et que l'on exécute les sept commandes suivantes. Dessiner l'arborescence finale des fichiers et répertoires (on utilisera TEST comme racine de l'arborescence).

1. `mkdir a b c d`
2. `touch a/t.txt d/foo.txt`
3. `cd c`
4. `mkdir ../b/e f g`
5. `cd ..`
6. `cp */*.txt c/g`
7. `rm -rf d`

*Indications : le symbole * seul signifie "tous les fichiers ou dossiers qui ont pour nom n'importe quelle chaîne de caractères"*

L'instruction `rm -rf` permet de supprimer un dossier ainsi que tout son contenu (fichiers et dossiers) sans demander la permission

Exercice 3 Permissions

On suppose pour cet exercice que le répertoire courant est le répertoire personnel. De plus on suppose que les répertoires **NSI** et **NSI/TP_SHELL** existent. Donner les commandes permettant de mettre les permissions demandées, quelles que soient les permissions initiales sur le fichier ou répertoire.

1. Le répertoire personnel possède tous les droits pour l'utilisateur et uniquement le droit d'exécution pour le groupe et les autres.
2. Les répertoires **NSI** et **NSI/TP_SHELL** possèdent tous les droits pour l'utilisateur et les droits de lecture et d'exécution pour le groupe et les autres (donner une commande par répertoire).
3. Le fichier *lisible.txt* du répertoire **NSI/TP_SHELL** possède les droits de lecture/écriture pour l'utilisateur et uniquement les droits de lecture pour le groupe et les autres.
4. Le fichier *secret.txt* du répertoire **NSI/TP_SHELL** possède les droits de lecture/écriture pour l'utilisateur et aucun droit pour le groupe et pour les autres.

Exercice 4 Les motifs glob

Partie I

Que permettent les lignes de commandes suivantes ?

1. `ls mesphotos/*.jpg`
2. `ls /bin/*.???`
3. `ls *[aei]??[0-9]`
4. `ls *[^.].???`
5. `ls -l /bin/*sh`
6. `ls *[^A-Z]`
7. `ls *[a-h]*[^4-7]`
8. `ls [0-9]*[a-z]`

Partie II

1. Quelle commande permet d'afficher tous les fichiers ou dossiers du répertoire **/usr/bin** commençant par s ?
2. Quelle commande permet d'afficher la liste des fichiers ou dossiers du répertoire **/etc** dont le nom commence par toute lettre minuscule comprise entre d et f ?
3. Quelle commande permet d'afficher la liste des fichiers ou dossiers du répertoire **/bin** dont le nom comporte exactement 6 caractères ?
4. Quelle commande permet d'afficher la liste des fichiers ou dossiers du répertoire **/var/log** qui ont une extension .log ?
5. Quelle commande permet d'afficher la liste des fichiers ou dossiers du répertoire **/var/log** dont le nom commence par la lettre a et qui contiennent des chiffres ?

Notion : Commandes de base des systèmes POSIX

Com- mande	Description	Exemple d'utilisation
pwd		
cd		