

6 Le système de fichiers

Pour explorer l'organisation du système de fichiers sous Linux, nous allons travailler avec des machines virtuelles sur lesquelles vous disposerez des droits d'administration (sur la machine réelle vous êtes connectés dans un mode utilisateur pour lesquels les droits sont restreints).

Comme on pourra faire beaucoup de choses avec les machines virtuelles, et donc aussi des erreurs, on prendra soin de prendre des 'instantanés' (= snapshots en anglais) de l'état de la machine virtuelle. Ceci nous permettra de remettre la machine virtuelle dans un état antérieur.

En particulier, en fin de séance, on remettra la machine dans son état initial

La prise d'instantanés est décrite pas à pas à l'adresse :

<https://laboiteaphysique.fr/site2/index.php/numerique-et-sciences-informatiques/systeme-dexploitation/obtenir-un-systeme-dexploitation-linux/machine-virtuelle>

ou en faisant une recherche avec l'expression '*machine virtuelle*' sur l'entrée du site laboiteaphysique.

Un OS Linux doit avoir un administrateur (le '**root**') et peut avoir **un ou plusieurs utilisateurs** (on dit que c'est un *OS multi-utilisateurs*).

Le mot de passe de l'administrateur est défini au moment de l'installation de la distribution par la personne qui la réalise.

Lors de cette installation un 'utilisateur' est également défini par :

- un nom d'utilisateur (*user name*)
- et un mot de passe (*password*)

Remarque : il est possible d'attribuer à un utilisateur des droits supérieurs. Il peut devenir momentanément un 'superutilisateur' pour faire des actions d'administration (par exemple installer ou mettre à jour des paquets logiciels). Pour cela l'utilisateur doit pouvoir passer en mode 'sudo'. Cette possibilité ne peut lui être attribuée que par le 'root'.

Voici les mots de passe de nos deux machines virtuelles :

Debian 10		Ubuntu_server	
mdp administrateur	sudebian	server's name	nsiserv
nom utilisateur 1	nsi1	username	admins
mdp utilisateur 1	nsi1	password	admins
nom utilisateur 2			
mdp utilisateur 2			

... des mots de passe à ne pas prendre en exemple !!

6.1 Exploration du système de fichiers avec un frontal graphique

Remarque : la machine virtuelle peut accéder à Internet, mais ceci se fait à travers la machine réelle. Si celle-ci n'est pas connectée/autorisée sur le réseau, la machine virtuelle en sera privée. Avant de lancer la machine virtuelle, il faut donc lancer Firefox sur la machine réelle et fournir ses identifiants de connexion. Ne pas refermer Firefox.

Lancer Virtualbox puis sélectionner (et non pas *Lancer*) la machine virtuelle Debian.

Réaliser un premier instantané nommé 'Post installation' si celui-ci n'existe pas encore

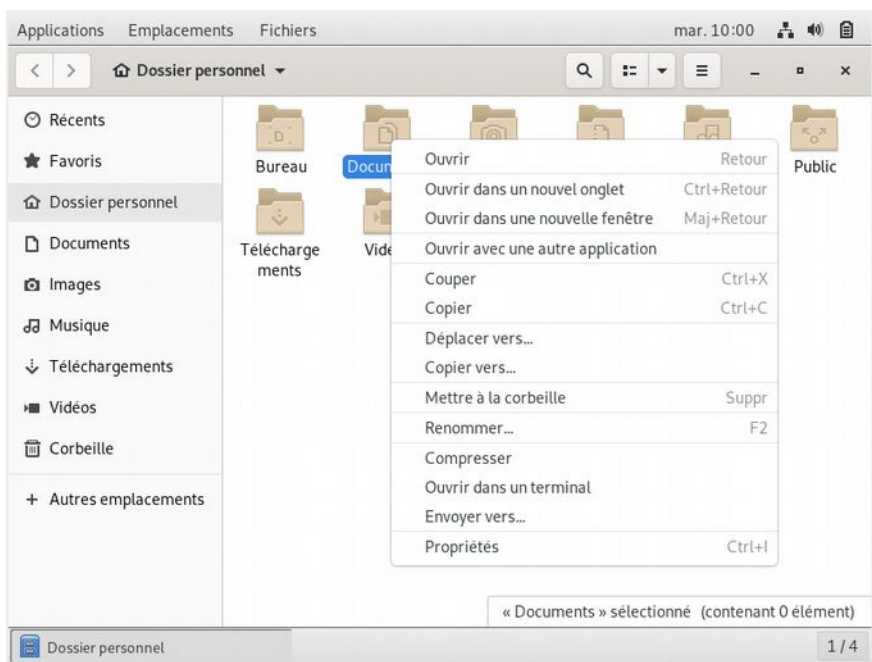
Démarrer maintenant la machine virtuelle Debian.

Le seul utilisateur déclaré est 'nsi1' se logger avec le mot de passe donné ('nsi1'). On démarre alors sur une version Debian en mode graphique que vous connaissez déjà. Pour travailler sur les fichiers, on pourra :

- soit lancer le programme 'Fichiers' dans le menu Applications → Accessoires (ou peut-être dans Applications → Favoris)
- soit à partir du menu Emplacements

L'explorateur de fichiers s'ouvre par défaut dans un répertoire (ou dossier) appelé « Dossier personnel » (on verra que cette appellation ne correspond à aucune appellation « physique » sur le disque dur. A la création de l'utilisateur nsi1, plusieurs sous-répertoires ont été créés (Bureau, Documents ...). Un clic droit sur un dossier permet d'afficher un *menu contextuel* qu'il sera intéressant d'explorer pour améliorer sa *productivité*. Essayer par exemple *Ouvrir dans un nouvel onglet*.

6.1.1. Présentation de l'explorateur de fichiers



Cliquer sur *Propriétés* :

Onglet 'Général'	Onglet 'Permissions'
<p>The 'Général' tab shows the following information: Nom: Documents, Type: Dossier (inode/directory), Contenus: aucun, Dossier parent: /home/nsi1 (circled in red), Espace libre: 6,2 Go.</p>	<p>The 'Permissions' tab shows: Propriétaire: Moi, Accès: Création et suppression des fichiers, Groupe: nsi1, Accès: Accès aux fichiers, Autres: Accès aux fichiers, Contexte de sécurité: inconnu. A button at the bottom says 'Changer les permissions des fichiers inclus...'.</p>

Remarquer ici le *dossier parent* c'est à dire le dossier dans lequel se trouve le dossier Documents.
Documents se trouve dans un dossier *nsi1*, lui même situé dans un dossier *home*

L'onglet permissions permet de faire apparaître le caractère *multi-utilisateurs* d'un OS Linux. Un fichier ou un dossier pourra être manipulé de façon plus ou moins exhaustive selon que l'on en est le propriétaire ou pas. Un utilisateur peut :

- en être le propriétaire et disposer des pleins droits (comme ici la possibilité de créer ou de supprimer des fichiers à l'intérieur des fichiers à l'intérieur de ce dossier *Documents*)
- faire partie d'un même groupe que le propriétaire, ici le groupe '*nsi1*', ce qui peut donner lieu à des droits restreints, (comme ici ne pouvoir qu'*Accéder aux fichiers*')
- appartenir à un autre groupe ('*Autres*') ce qui peut donner lieu à des droits éventuellement encore plus restreints

Propriétaire :	Moi
Accès :	Création et suppression des fichiers ▼
Groupe :	nsi1 ▼
Accès :	Aucun
Autres :	Lister seulement les fichiers
Accès :	Accès aux fichiers
Contexte de sécurité :	Création et suppression des fichiers

Le propriétaire d'un dossier ou d'un fichier peut allouer les droits aux utilisateurs (voir la copie d'écran ci-dessus).

Remarque : tout ceci peut paraître anecdotique pour un poste de travail (surtout si on l'utilise en mono-utilisateur...) mais revêt une importance considérable dans le cadre d'un serveur !!!

6.1.2. Quelques exemples de manipulations :

Créer dans Documents un dossier '*Textes*'.

Dans le menu Applications → Accessoires, lancer le logiciel *Editeur de texte* et créer un texte que l'on enregistrera par exemple sous l'appellation *essai1.txt* dans le dossier '*Textes*' créé précédemment.

Avec le menu contextuel, modifier les droits que vous vous autorisez sur

- le dossier *Documents*
- ou sur le dossier *Textes*
- ou sur le fichier *essai1.txt*

Essayer toutes les combinaisons possibles et regarder à chaque fois s'il est possible de :

- lire le fichier *essai1.txt*
- le modifier (et enregistrer ces modifications)
- le déplacer dans un autre dossier
- le supprimer

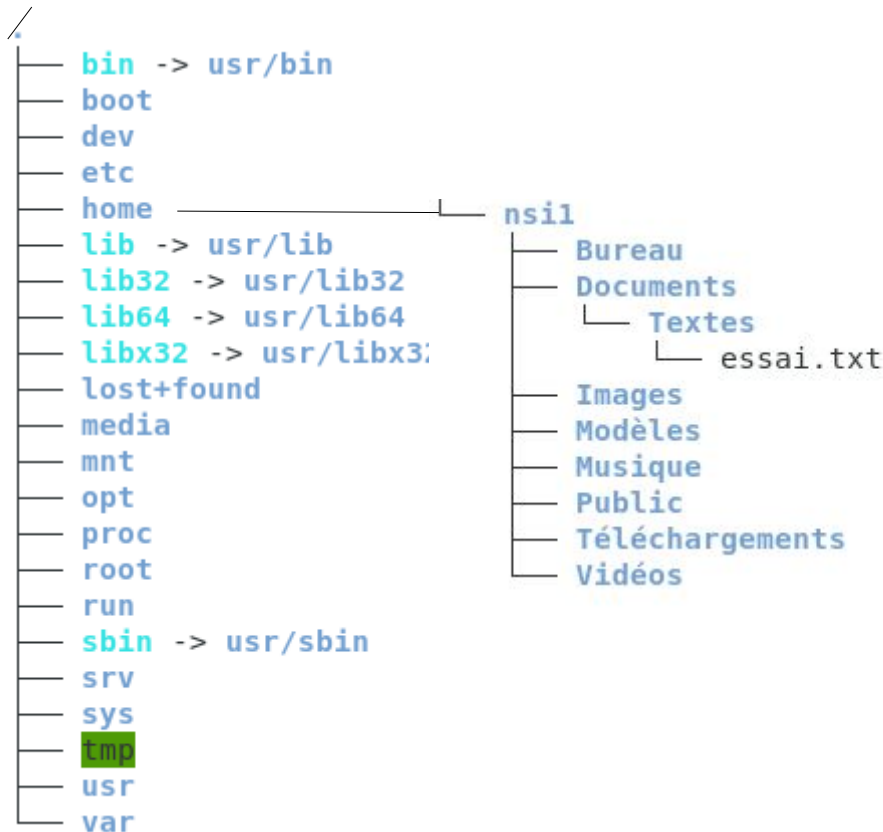
6.1.3. Exploration de l'arborescence

Dans l'explorateur de fichiers, sélectionner '*Autres emplacements*' puis '*Ordinateur*'. On se trouve ici à la racine de l'arborescence du système de fichiers.

Observer les icônes des dossiers :

- peut-on accéder au dossier '*root*' ?
- quel est l'emplacement réel du dossier '*bin*' ?
- où se trouve notre '*Dossier personnel*' ?

Voici une représentation de l'arborescence de notre machine virtuelle :



6.2 Utilisation de la ligne de commandes

6.2.1. L'interpréteur de commandes

Lancer Virtualbox puis sélectionner (et non pas *Lancer*) la machine virtuelle Ubuntu_Server.

Réaliser un premier instantané nommé 'Post installation' si celui-ci n'existe pas encore

Démarrer maintenant la machine virtuelle Ubuntu_Server. Se logger avec le compte et mot de passe donné...

```
admins@nsiserver:~$ ... et maintenant, que fait-on ???
```

Sur ce serveur, le système d'exploitation est réduit au minimum : après avoir démarré ses différents services, le système d'exploitation offre une interface homme-machine des plus réduites : le *shell* . Celui disponible sous Debian et distributions dérivées (comme Ubuntu) se nomme *bash* . Ce logiciel est un *interpréteur de commandes* . Une fois démarré, il affiche le *prompt* :

```
admins@nsiserver:~$ (admins@nsiserver = l'utilisateur admins connecté sur la machine nsiserver)
```

...qui attend indéfiniment que l'utilisateur lui passe une commande valide pour pouvoir l'exécuter.

Exemple : `pwd` est une commande qui provient de l'expression anglaise '*print working directory*' (= affiche le répertoire de travail = le répertoire courant)

```
admins@nsiserver:~$ pwd
```

renvoie : `/home/admins` et réaffiche le prompt pour se mettre en attente d'une autre commande :

```
admins@nsiserver:~$
```

6.2.2. Quelques commandes de base.

Le tableau ci-dessous liste quelques commandes de base pour se localiser / déplacer dans le système de fichier, afficher de l'information, copier / déplacer / effacer des dossiers ou des fichiers

commande	signification	traduction
man <i>nom_de_commande</i>	manual	aide sur une commande
id	identity	informations sur l'utilisateur
pwd	print working directory	afficher le répertoire de travail
tree		affiche l'arborescence
cd <i>chemin absolu ou relatif</i>	change directory	changer de répertoire
ls	list	lister les fichiers
mkdir <i>nom_nouveau_dossier</i>	make directory	créer un répertoire
cp <i>source destination</i>	copy	copier
mv <i>source destination</i>	move	déplacer
rm <i>nom_du_fichier</i>	remove	supprimer un fichier
rmdir <i>nom_du_dossier</i>	remove directory	supprimer un dossier vide
cat <i>nom_du_fichier</i>	concatenate	affiche le contenu du fichier
shutdown shutdown now		arrêter le système arrêter le système maintenant

Essayer de se déplacer dans l'arborescence de ce serveur et de lister le contenu de répertoires. Puis l'arrêter.

Pour un apprentissage plus aisé, nous allons travailler 'en ligne de commande' dans le 'Terminal' de la machine virtuelle Debian.

6.2.3. Le Terminal

C'est un programme qui tourne en mode fenêtré et qui donne accès à l'*interpréteur de commande* (donc bash également dans cette machine Debian)

Lancer le Terminal à partir du menu Application → Utilitaires

Qu'affiche le prompt : _____

Comment savoir à quel niveau de l'arborescence on se trouve : _____

Pour afficher l'arborescence, on va installer un paquet nécessaire : tree, ce qui va obliger à passer en mode super-utilisateur :

```
nsi1@VMDebian:~$ su -
```

entrer le mot de passe administrateur, et remarquer le changement du prompt :

```
root@VMDebian:~#
```

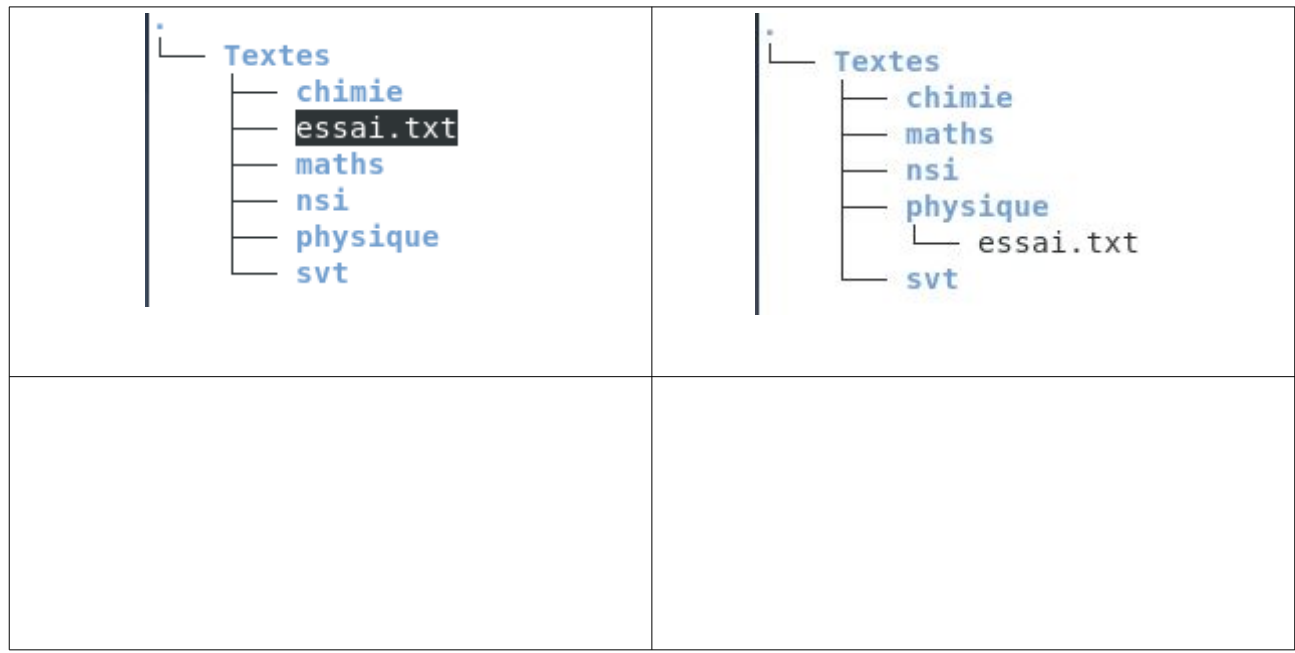
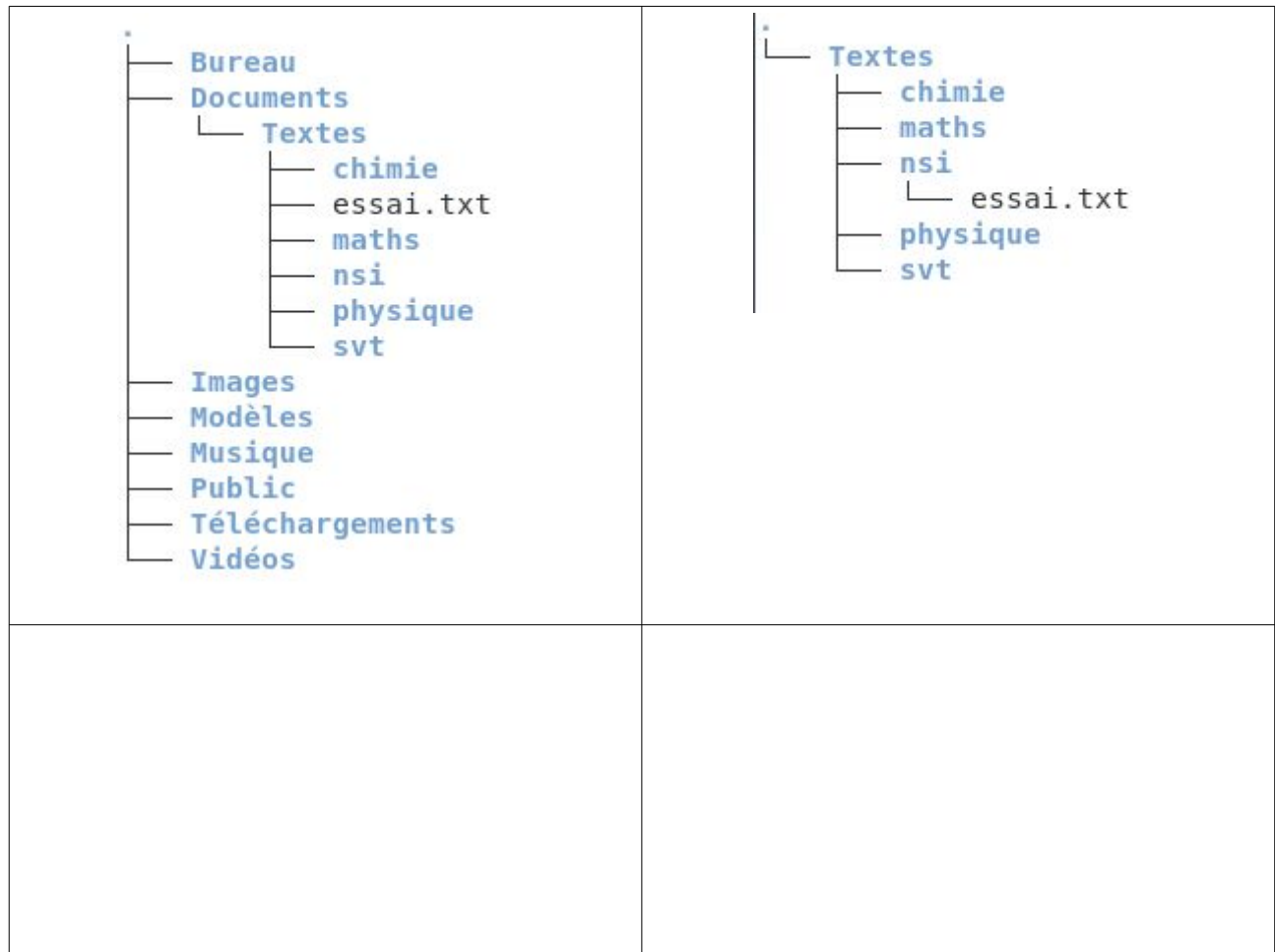
```
root@VMDebian:~# apt install tree
```

Pour quitter le mode root et revenir à l'utilisateur nsi1, entrer la commande exit ou Ctrl + d. On doit retrouver le prompt initial :

```
nsi1@VMDebian:~$
```

Tester la commande tree

Exercice :
 Réaliser **en ligne de commande** les opérations nécessaires pour obtenir **successivement** ces résultats :



6.2.4. Gestion des utilisateurs

Le tableau ci-dessous liste quelques commandes relatives à la gestion des utilisateurs (identification, création / suppression de compte, modification de compte, gestion des groupes...)

commande	signification / options	traduction
man <i>nom_de_commande</i>	manual	aide sur une commande
id	identity	informations sur l'utilisateur
groups		donne les noms de groupes auxquels l'utilisateur est affilié
su -	super user	passer en root
exit ou Ctrl + d		quitter le mode root
adduser		ajoute un utilisateur au système
addgroup		ajoute un groupe au système
usermod	-g -G -aG	affilier un utilisateur à un groupe
su <i>nom_utilisateur</i>		se connecter sur le compte d'un autre utilisateur

1. Récupérer de l'information :

commande id : cette commande fournit des informations sur l'utilisateur connecté :

```
nsi1@VMDebian:~$ id
```

```
uid=1000(nsi1)gid=1000(nsi1)groupes=1000(nsi1),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),109(netdev),112(blueetooth),117(lpadmin),118(scanner)
```

uid (user identifiant) : le nombre 1000 pour le premier utilisateur

gid (group identifiant) : le nombre 1000 pour le groupe par défaut 'nsi1' qui porte le même nom que l'utilisateur

groupes : l'ensemble des groupes auxquels est affilié l'utilisateur connecté (ici *nsi1*) :

commande groups : elle donne uniquement le nom des groupes auxquels est affilié l'utilisateur connecté :

```
nsi1@VMDebian:~$ groups
```

```
nsi1 cdrom floppy audio dip video plugdev netdev blueetooth lpadmin scanner
```

2. Créer un utilisateur :

Seul un utilisateur avec des privilèges peut créer un nouvel utilisateur. (On décide de le nommer *nsi2*)

Passer en *root* et utiliser la commande `adduser` ; compléter ci-dessous ce que renvoie la console :

```
root@VMDebian:~# adduser nsi2
```

```
Ajout de l'utilisateur «_____» ...
```

```
Ajout du nouveau groupe «_____» (____) ...# remarquer la valeur du gid de nsi2
```

```
Ajout du nouvel utilisateur « _____ » (____) avec le groupe «_____» ...
```

```
Création du répertoire personnel «_____»...
```

```

Copie des fichiers depuis « /etc/skel »...
Nouveau mot de passe :                # ici on doit attribuer un mot de passe à nsi2
Retapez le nouveau mot de passe :
passwd: _____ updated successfully
Changing the user information for _____
Enter the new value, or press ENTER for the default
    Full Name []: Nsi2 # on peut mettre des majuscules au nom apparent
    Room Number []:    # inutile de renseigner ces 4 rubriques ici...
    Work Phone []:
    Home Phone []:
    Other []:
Cette information est-elle correcte ? [0/n] # 0 = la lettre o majuscule (=Oui)
root@VMDebian:~#

```

→ Recommencer cette procédure pour créer un troisième utilisateur nommé *nsi3*.

Se déconnecter du compte *root* pour se reconnecter successivement sur le compte *nsi2* puis *nsi3* pour en récupérer les identifiants. Que remarque-t-on pour les uid et gid :

→ compte nsi2 : _____

→ compte nsi3 : _____

3. Créer un groupe :

On souhaite créer un groupe que l'on veut nommer *ma_classe*. Là aussi c'est en tant qu'utilisateur privilégié que l'on peut le faire. On utilisera le commande *addgroup* :

4. Adhérer à un groupe :

Toujours en tant qu'administrateur privilégié, on utilise la commande *usermod*. Mais attention, celle-ci s'utilise avec différentes options possibles. La syntaxe sera de la forme :

```
root@VMDebian:~# usermod [options] le_groupe l'utilisateur
```

Dans l'interpréteur de commande, obtenir la page d'informations relative à cette commande. Il faut taper : _____

On vous propose de regarder les options : -g ; -G et -aG :

usermod -g :

usermod -G :

usermod -aG :

On souhaite faire appartenir les utilisateurs nsi1 et nsi2 au même groupe ma_classe ; nsi3 sera laissé en dehors de ce groupe. Pour affilier nsi1 au groupe ma_classe, sans modifier son appartenance à d'autres groupes :

```
root@VMDebian:~# usermod -aG ma_classe nsi1
```

Est-il possible de vérifier à partir de ce compte root (ou de tout autre compte) que l'utilisateur nsi1 est bien affilié au groupe ma_classe ? De quelle instruction faut-il consulter le manuel ?

Affilier de même l'utilisateur nsi2 au groupe ma_classe. Le vérifier.

Nsi2 appartient-il encore à son groupe primaire nsi2 : _____

On veut maintenant changer le groupe primaire de l'utilisateur Nsi3 pour le faire appartenir à un groupe nommé *pasteur* qui n'existe pas encore. Réaliser les opérations nécessaires.

A la fin de ces petits exercices vous devriez avoir :

Utilisateur (nom apparent)	Utilisateur (nom interne)	Groupes
Nsi1	nsi1	nsi1 cdrom floppy audio ... scanner ma_classe
Nsi2	nsi2	nsi2 ma_classe
Nsi3	nsi3	pasteur

Remarque importante : nous avons travaillé en mode *root* pour obtenir les privilèges d'administration, ce qui oblige à donner à l'utilisateur nsi1 par exemple le mot de passe root (sudebian pour nous)...

Il y a des situations où il n'est pas souhaitable de donner ce mot de passe administrateur à un utilisateur, mais de pouvoir lui donner néanmoins des droits d'administration. Il faut alors affilier cet utilisateur au groupe *sudo*. Cet utilisateur pourra réaliser une tâche d'administration en la faisant précéder de l'instruction *sudo*. Avant l'exécution de cette tâche, l'interpréteur de commande demandera à l'utilisateur d'entrer, non pas le mot de passe *root*, mais son propre mot de passe.

Dans cette opération, c'est toujours l'utilisateur courant qui est connecté à la machine :

on ne passe pas sur le compte root et on ne reste 'superutilisateur' que de façon momentanée

Exemple : affilier l'utilisateur Nsi2 au groupe sudo, puis se connecter en tant qu'utilisateur Nsi2. Lui faire créer un nouvel utilisateur (Nsi4)

6.2.5. Gestion des droits en mode graphique :

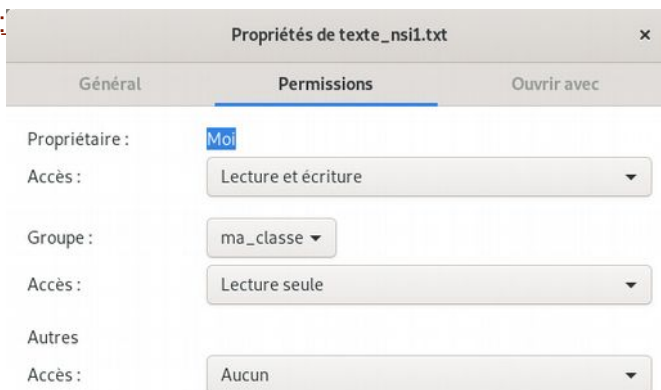
Pour que nos utilisateurs soient effectivement reconnus dans leurs groupes respectifs, il est nécessaire de fermer la / les sessions actives.

Dans une session Nsi1, créer un fichier texte (appelé ici *texte_nsi1.txt*) puis modifier les droits comme ci-contre :

Moi (nsi1) a tous les droits dessus

Le groupe *ma_classe*, uniquement les droits de lecture

Tous les autres utilisateurs : aucun droit



1. Cas d'un utilisateur appartenant au même groupe *ma_classe* :

L'utilisateur Nsi2 peut-il trouver le fichier *texte_nsi1.txt* ? _____

Peut-il l'ouvrir ? _____

Peut-il le modifier ? _____

Peut-il l'enregistrer ? _____

Peut-il l'enregistrer sous un autre nom dans le dossier Documents de Nsi1 ? _____ ; dans son propre dossier Documents ? _____

→ Revenir dans le compte Nsi1.

L'utilisateur Nsi1 peut-il trouver la version enregistrée par Nsi2 du fichier *texte_nsi1.txt* ? _____

Quels droits sont appliqués à ce fichier ? _____

Les droits initiaux du fichier *texte_nsi1.txt* ont-ils été conservés lors de l'enregistrement de la version faite par Nsi2 ? _____

2. Cas d'un utilisateur n'appartenant pas au groupe *ma_classe* :

→ Changer d'utilisateur pour se connecter en Nsi3.

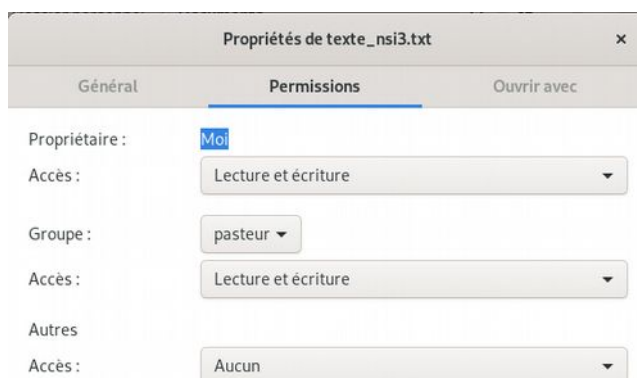
L'utilisateur Nsi3 peut-il trouver le fichier *texte_nsi1.txt* ? _____

Peut-il l'ouvrir ? _____

Créer dans les Documents de Nsi3 un texte (ici *texte_nsi3.txt*) et lui attribuer les droits comme ci-contre.

On souhaite que les utilisateurs Nsi1 et Nsi2 puissent accéder en Lecture et en Ecriture à ce fichier, **mais pas l'utilisateur Nsi4**.

Réaliser en ligne de commande, l'ensemble des opérations nécessaires. Les décrire ci-dessous :



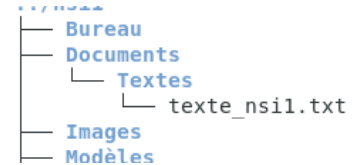
6.2.6. Gestion des droits en ligne de commande :

Le tableau ci-dessous liste quelques commandes relatives à la gestion des droits (listage, modification)

commande	signification / options	traduction
man <i>nom_de_commande</i>	manual	aide sur une commande
ls	list	liste les fichiers
chmod	change file mode bits	modifier les droits sur un fichier / dossier

Reconnecter la session graphique sur le compte Nsi1.

On suppose pour la suite que l'on a l'arborescence suivante pour le compte nsi1, sinon faire les modifications nécessaires (... de préférence en ligne de commande)



Dans le terminal, rechercher l'information sur la commande *ls*. A quoi sert l'option *-l* ? (L minuscule)

L'essayer à partir de */home/nsi1* :

```
nsi1@VMDebian:~$ pwd
/home/nsi1
nsi1@VMDebian:~$ ls -l
total 32
drwxr-xr-x 2 nsi1 nsi1 4096 sept.  1  2019 Bureau
drwxr-xr-x 3 nsi1 nsi1 4096 mars  11 17:56 Documents
drwxr-xr-x 2 nsi1 nsi1 4096 sept.  1  2019 Images
```

Se placer dans *home/nsi1/Documents* et refaire un listage du contenu :

```
nsi1@VMDebian:~/Documents$ ls -l
total 4
drwx--x--x 2 nsi1 nsi1 4096 mars  11 17:56 Textes
```

Puis rentrer dans le répertoire *Textes* et lister à nouveau le contenu :

```
nsi1@VMDebian:~/Documents/Textes$ ls -l
total 4
-rw-r--r-- 1 nsi1 ma_classe 30 mars  10 09:42 texte_nsi1.txt
```

Remarque : on n'est pas obligé de se mettre dans le répertoire dont on veut lister le contenu ! Cela peut se faire 'à distance', par exemple à partir du répertoire *home/nsi1* :

il suffit de faire suivre la commande *ls -l* du chemin qu'il faut suivre pour accéder au répertoire voulu :

```
/home/nsi1
nsi1@VMDebian:~$ ls -l Documents/Textes
total 4
-rw-r--r-- 1 nsi1 ma_classe 30 mars  10 09:42 texte_nsi1.txt
```

Exemple : à partir de *home/nsi1*, lister la racine du système de fichiers :

```
lrwxrwxrwx  1 root root    7 sept.  1  2019 bin -> usr/bin
drwxr-xr-x  3 root root 4096 sept.  1  2019 boot
drwxr-xr-x 17 root root 3160 mars  11 17:48 dev
drwxr-xr-x 119 root root 4096 mars  11 17:48 etc
drwxr-xr-x  6 root root 4096 mars  10 09:28 home
```

Décryptage : chaque fichier ou dossier listé est décrit sur une ligne que l'on peut décomposer en 8 colonnes :

Type / Droits	Liens	Propriétaire	Groupe	Taille	Date	(Date)Heure	Nom
drwx--x--	2	nsi1	nsi1	4096	mars 11	17:56	Textes
-rw-r--r--	1	nsi1	ma_classe	30	mars 10	09:42	texte_nsi1.txt
lrwxr-xr-x	1	root	root	7	sept 1	2019	bin → usr/bin
drwxr-xr-x	6	root	root	4096	mars 10	09:28	home

On va regarder de plus près la première colonne : elle est constituée de ____ caractères :

		Droits								
Ex.	Type	propriétaire (u)			groupe (g)			autres (o)		
		lect. (r)	écrit.(w)	exéc.(x)	lect. (r)	écrit.(w)	exéc.(x)	lect. (r)	écrit.(w)	exéc.(x)
1	-	r	w	-	r	-	-	r	-	-
		4	2	0	4	0	0	4	0	0
		4 + 2 + 0 = 6			4 + 0 + 0 = 4			4 + 0 + 0 = 4		
	valeur en octal :	644								
2	d	r	w	x	-	-	x	-	-	x
	valeur en octal									

Dans le tableau ci-dessus, l'exemple 1 correspond aux droits du fichier `texte_nsi1.txt` et l'exemple 2 au dossier `Textes`.

En s'aidant des tableaux de décryptage ci-dessous, indiquer les droits qu'ont les différents utilisateurs sur ce fichier et ce dossier :

fichier `texte_nsi1.txt` :

dossier `Textes` :

Compléter le tableau pour déterminer la valeur en octal correspondant aux droits du dossier `Textes`

Droits			
symbole :	pour un fichier :	pour un dossier =:	poids =
r	droit en lecture	on peut lister le contenu	4
w	droit en écriture	ajout, effacement, renommage des fichiers	2
x	droit en exécution *	on peut 'traverser' le dossier	1
-	le droit n'est pas donné	le droit n'est pas donné	0

* pour un fichier exécutable (= un programme)

Type :
- : pour un fichier ordinaire
d : pour un répertoire (' <i>directory</i> ')
l : pour un lien symbolique (' <i>link</i> ')

Modifier les permissions sur un fichier ou sur un dossier consiste à modifier de façon individuelle ou globale les bits rwx pour chaque type d'utilisateur.

```
nsi1@VMDebian:~/Documents$ ls -l
```

Dans l'interface graphique modifier les permissions du fichier *texte_nsi.txt* de façon à ce que tous les utilisateurs puissent avoir les droits en lecture et en écriture. Refaire le listage complet du dossier Documents.

quels changements observe-t-on ? _____

Refaire la manipulation en n'accordant que le droit de lecture à tous les utilisateurs :