Atelier Système 1 V 2 Année 2023	
Atelier 03	Classes : ***Classes L1TI***

Partie 02 : Permission des fichiers

Objectifs

- Devenir familier avec les fonctions, la syntaxe et l'utilisation de plusieurs commandes essentielles de modification des permissions des fichiers.
- Combiner ces commandes de manière utile pour accomplir des tâches d'utilisateur communes.

Configuration de départ

Un système Linux Ubuntu, installé et fonctionnant avec des comptes utilisateurs sans privilèges.

Préambule

A - Ce TP s'effectue en mode console en premier lieu puis avec l'interface graphique à la fin de la dernière séance.

B - Chaque utilisateur appartient à un ou plusieurs groupes. Pour cela, il doit pouvoir gère les droits d'utiliser ses fichiers par les autres utilisateurs, qui se sont soit membres de ses groupes ou non et de les interdire ou les autoriser à lire, modifier ou exécuter ses fichiers.

C - Notions

1. **Utilisateur propriétaire** d'un fichier **(u) :** Il s'agit du créateur du fichier, à noter qu'un fichier créé par une commande exécutée à l'aide de **sudo** appartiendra à l'utilisateur ***root***.
2. **Membre** de **groupe propriétaire (g)** : Est tout utilisateur membre d'un groupe de **propriétaire** de fichier, ce membre aura aussi certaines permissions particulières sur ce fichier.
3. **Other**, le reste des utilisateurs (**o**) : tout autre utilisateur n'étant ni propriétaire du fichier, ni membre du groupe propriétaire du fichier.
4. Les permissions se définissent sur trois niveaux **:**
   - **Lecture** d'un fichier : permission nécessaire pour pouvoir accéder au contenu d'un fichier sans le modifier. Cette permission est notée **r** (***read***).
   - **Ecriture** dans un fichier : permission nécessaire pour pouvoir apporter des modifications à un fichier (pour un dossier : ajouter, modifier, renommer ou supprimer un fichier). Cette permission est notée **w** (***write***).
   - **Exécution** d'un fichier : permission nécessaire particulièrement pour les logiciels, afin qu'ils puissent être exécutés. Cette permission est notée **x** (***execute***).
5. **Types de fichiers,** à chaque type est associé un caractère
   - Fichier ordinaire **(-)**
   - Fichiers répertoires (**d**)
   - **Fichier spécial** : périphérique accède en mode caractère (**c**)
   - **Fichier spécial** : périphérique accède en mode bloc (**b**)
   - Tubes (**p**)
   - Lien symbolique (**I**)

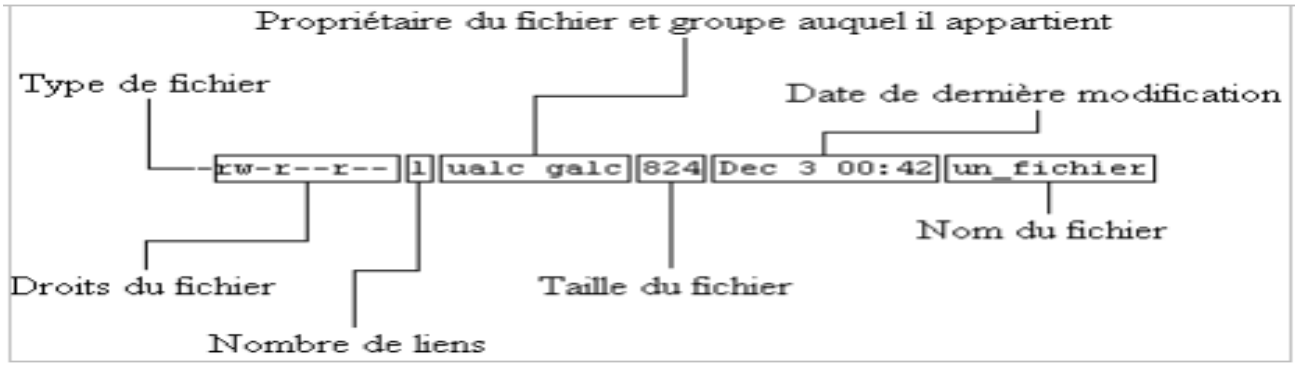
I – Passage entre les terminaux (consoles)

I – 1 - Allumer l'ordinateur**.**

I - 2 - Connexion au système : Passer à la console **tty4** puis Connectez-vous à votre compte en suivant les indications suivantes : **Login** : …….. et **passwd** : …….

I - 3 - Affichage de différents droit des fichiers

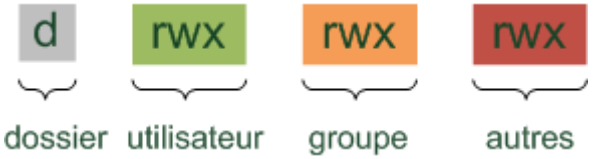
Pour afficher les différents droits des fichiers, exécutons la commande **ls -l** dans un terminal



La première colonne (la plus à gauche) contient les informations concernant les différentes permissions appliquées sur les fichiers ainsi que leurs types.

Cette colonne est composée de dix (10) lettres dont :

- La première désigne le type de fichier :
  - - : fichier "classique"
  - d : répertoire (directory)
  - l : lien symbolique (link)
- Les neuf suivantes désignent les différentes permissions sur les fichiers :
  - **r** : read (droit de lecture)
  - **w** : write (droit d'écriture)
  - **x** : execute (droit d'exécuter un fichier ou d'ouvrir un répertoire)
- Ces droits sont affichés de la sorte :
  - Les 3 premiers sont les droits du propriétaire du fichier.
  - Les trois suivants du groupe.
  - Les trois derniers des autres.



Utilisez cette commande (**ls - l**) et vérifiez s'il y a des fichiers dans votre répertoire et remarquez leurs permissions et leurs types (complétez le tableau suivant décrivez 3 fichiers)

Nom de fichier	Type	Droits propriétaire	Droits groupe	Droits autres

II - Permissions par défaut

La commande **umask** permet **d'afficher** ou de **modifier** les permissions par défaut d'un utilisateur

1. Utilisez la commande **man** pour voir le fonctionnement d'**'umask',** expliquez le fonctionnement de cette commande

Déposer ici une capture d'écran de résultat

……..……

2. Utilisez cette commande pour afficher vos permissions par défaut

$ ----------

Déposer ici une capture d'écran de résultat

3. Créez deux fichiers et un répertoire. Affichez leurs permissions.

$ touch untest1 untest2

$ mkdir untestdir1

$ ls -ld un*

Permission **untestfich1 :** ----------

Permission **untestfich2 :** ----------

Permission **untestrep1 :** ----------

4. Modifiez votre **umask** pour des paramètres plus sécurisés. Créer ensuite deux nouveaux fichiers et un répertoire. Comparez ensuite leurs permissions.

$ umask 027

$ touch untestfich3 untestfich4

$ mkdir untestrep2

$ ls -ld un*

Permission untestfich1 : ----------

Permission untestfich2 : ----------

Permission untestrep1 : ----------

Permission untestfich3 : ----------

Permission untestfich4 : ----------

Permission untestrep2 : ----------

Expliquez la différence entre les permissions des fichiers créés en 3^{ieme} et 4^{ieme} questions :

……..……...…
……..……...……
……..……...…
……..……...……

III -: Définition de permissions de fichier

Explication

En réalité, pour modifier les droits, on utilise **chmod** suivent deux façons différentes, avec des lettres ou avec des chiffres.

chmod [-R] droits fichier

Utiliser **chmod** avec **-R** sur un dossier, permet d'affecter récursivement les droits à tous les fichiers et sous-dossiers de ce dossier.

- mode lettre (forme symbolique)

Dans ce mode, on utilise des lettres et des opérateurs pour ajouter ou supprimer des droits.

u	**u**ser, utilisateur	r	Read
g	**g**roup, groupe	w	write
o	**o**thers, autres	x	execute
a	**a**ll, tous représente l'ensemble des trois catégories.		

Les opérateurs disponibles sont

+	Ajoute un droit
-	Supprime un droit
=	Ajoute un droit et supprime les autres

- mode chiffre (forme numérique)

Dans ce mode, on utilise des **valeurs octales** pour ajouter ou supprimer des droits

chiffre	signification	En additionnant ces chiffres, on obtient toutes les combinaisons possibles	
1	execute '--x'	1+2 =3 ==>'-wx'	1+2+3=7 ==>' wrx'
2	write '-w-'	1+4 =5 ==>'r-x'	0 '---' aucun droit à n'importe qui.
4	read 'r--'	2+4 =6 ==>'rw-'	

En concatènent ces chiffres, on arrive à un nombre à trois chiffres. Centaine pour l'utilisateur, dizaine pour le groupe et unité pour les autres. **Exemples**

chmod o-w fichier3 enlèvera le droit d'écriture pour les autres.

chmod a+x fichier4 ajoutera le droit d'exécution à tout le monde

chmod u+rwx, g+rx-w, o+r-wx fichier3

- ajouter la permission de lecture, d'écriture et d'exécution sur le fichier fichier3 pour le propriétaire ;
- ajouter la permission de lecture et d'exécution au groupe propriétaire, on retire la permission d'écriture ;
- ajouter la permission de lecture aux autres, on retire la permission d'écriture et d'exécution.

Pour Changer le groupe d'un fichier (seuls **root** ou le propriétaire actuel d'un fichier peut utiliser **chgrp**)

chgrp [-R] newgroup filename/directoryname

sudo chgrp mesPotes fichier2 Le fichier fichier2 appartient maintenant au groupe mesPotes.

Pour Changer le propriétaire et le groupe d'un fichier (seuls **root** ou le propriétaire actuel d'un fichier peut utiliser **chown**)

chown [-R] user [:group] fichier

chown utilisateur2:groupe2 foo.txt ==> Le propriétaire de foo.txt devient *utilisateur2* et le groupe de ce fichier devient *groupe2*

1. En utilisant la commande **sudo**, passer en mode **super-utilisateur (root)**

Déposer ici une capture d'écran de résultat

2. Créez un utilisateur **L1TI1 avec** mot de passe **L1TI1** et **L1TI2** avec mot de passe **L1TI2.**

Déposer ici une capture d'écran de résultat

3. Créez les utilisateurs **L2SEM** avec mot de passe **L2SEM**. Et **L2RSI** avec mot de passe **L2RSI**.

Déposer ici une capture d'écran de résultat

4. quittez le mode **super-utilisateur (root**) et revenez à votre mode.

……..……...…

5. Quel est votre groupe principal

……..……...…

6. ajoutez **L1TI1** et **L1TI2** à votre **groupe**

……..……...…

7. Quels sont les permissions de Votre répertoire personnel ?

……..……...…

8. L'utilisateur **L2RSI** qui ne faisait partie de votre groupe peut-il modifier ou accéder à votre répertoire ou lister vous fichiers ?

……..……..…

9. et **L1TI2**, le pourrait-il (il faisait partie de votre groupe) ?

……..……..……

10. La commande **chmod** (**ch***ange* **mod***e*, changer les permissions) permet de modifier les droits d'un fichier. En utilisant la commande **man** expliquez le fonctionnement de **chmod**, **man chmod**

Déposer ici une capture d'écran de résultat

11. La commande **chgrp** (**ch**ange **gr**ou**p**) permet de changer le groupe auquel appartient le fichier. En utilisant la commande **man** expliquez le fonctionnement de **chgrp**, **man chgrp**

Déposer ici une capture d'écran de résultat

12. La commande **chown** (**ch**ange **gr**ou**p**) permet de changer à la fois le propriétaire et le groupe propriétaire d'un fichier. En utilisant la commande **man** expliquez le fonctionnement de **chown**, **man chown**

Déposer ici une capture d'écran de résultat

13. Quelles commandes devriez-vous écrire pour accorder le droit de visite de votre répertoire personnel seulement à **L1TI2** (et pas à **L2RSI**) ?

……..……..…

14. Quelles commandes devriez-vous écrire pour accorder le droit de visite de votre répertoire personnel seulement à **L2RSI** qui ne faisait partie du groupe ?

……..……..……

IV - Changements des droits

1. Comparer les permissions de **/etc/passwd** et **/etc/shadow**. Pourquoi a-t-on nommé ainsi ce dernier fichier ? pourriez-vous le lire ? et voir sa présence ? L'examiner pour deviner son rôle.

……..……..…
……..……..……
……..……..…
……..……..……

2. Par précaution, faire une copie de **shadow** sous le nom **shadow.bak** dans **/home/votre répertoire/temp**! vérifiez les droits de **/home/votre répertoire/temp/shadow.bak**

……..……..…

3. Pensez-vous tout de même pouvoir supprimer le fichier précédent ? Concluez !

……..……..…
……..……..……

4. En utilisant la commande **sudo** copier maintenant le fichier **shadow** chez vous, dans votre répertoire d'accueil, sous le nom **shadow.bak** et accordez-vous la propriété de la copie.

a) Comment faites-vous ?

……..……

b) Vérifiez le résultat

……..……..…

5. Vous éditez ce fichier**,** le modifiez, par exemple en supprimant des lignes, et vous faites une mise à jour. La mise à jour sera-t-elle réalisée ? pourquoi ?

……..……..…
……..……..……

6. Pensez-vous que vous pouvez supprimer ce fichier ? Essayez et expliquez !

……..……..…
……..……..……

7. pouvez-vous créer le répertoire temporaire **/home/temp01** ? pourquoi ?

……..……..…
……..……..……

8. Effectuez cette création comme **root** (pensez à la commande **su**).

……..……..…

9. Accorder vous les permissions maximales sur **/home/temp01**; vérifiez.

……..……..…

10. **L2RSI**, toujours lui, tout content d'avoir enfin un droit d'écriture, dans **/home/temp** essaie de copier les 2 fichiers système **/etc/hosts** et **/etc/passwd** dans **/home/temp** ?

y parviendra-t-il ? Pourquoi ? Que donne la commande **ll /home/temp** ?

……..……..…
……..……..……

11. **L1TI2**, essaie maintenant de supprimer ces 2 fichiers de **/etc**. Réussit-il ?

……..……..…
……..……..……

12. Effrayé à l'idée de se faire pincer par **root**, **L2RSI** veut masquer sa faute tout en vous faisant à sa place ! Pour cela, il veut que vous devienne propriétaire du fichier copié **passwd**.

Comment s'y prend-il ? Réussit-il ? Et vous comment auriez-vous fait ?

……..……..…
……..……..……

13. éliminez **L2RSI** de votre groupe

……..……..…
……..……..……

V - sécuriser le partage d'un répertoire

Il s'agit de créer un répertoire partagé par tous les membres de votre groupe

1. Créez dans **/home** un répertoire appelé **rep-stagiaire01**. Rappelez pourquoi cette tâche relève des prérogatives de **root**

……..……
……..……

2. Faites-le appartenir à votre groupe

……..……..…

3. Modifier les permissions sur le répertoire pour que tous les membres de votre groupe puissent y écrire et s'y déplacer.

……..……..…

4. créez un fichier texte et vous le déposez dans **/home/rep-stagiaire01**.

……..……..……

Si vous êtes paresseux, vous y faites une copie d'un fichier qcq, par exemple /**etc/hosts,** mais en attribuant des droits 660

……..……..……

5. Vérifiez le bon accès en lecture **seulement** pour les membres du groupe. Ainsi **L2RSI** qui a fini par être **exclu** de votre groupe ne doit pas pouvoir le lire. A vérifier.

……..……..…

6. Votre collègue (membres de votre groupe) le perfide **L1TI2**, tente de supprimer ce fichier ou de le renommer Y parvient-il ? Essayez !

Pourtant, vérifiez que ce fichier appartient à votre groupe ? N'est-ce pas inquiétant ? Expliquez comment cela est possible.

……..……..…
……..……..……

Résultat

Changement d'un certain nombre de permissions, le propriétaire et le groupe d'un fichier.

Fin du cet atelier

Tableau récapitulatif de liste des "fichiers système" et commandes utiliser dans cet atelier

---

Commandes

Commandes	Signification
ls -l	Lister le contenu d'un répertoire.
touch	permet de changer la date du fichier et aussi de créer un fichier de taille 0
mkdir	Permet de créer un répertoire

useradd	Ajouter un utilisateur.
userdel	Supprimer un utilisateur.
groups	Permet de connaître les groupes auxquels appartient l'utilisateur courant.
sudo	Permet à des utilisateurs de lancer des commandes de **root**.
exit	**exit** après **sudo** ou **su**; Permet de reprendre l'identité de l'utilisateur qui a lancé ces commandes.

umask	permet d'afficher ou de modifier les permissions par défaut d'un utilisateur
chmod	permet de modifier les droits d'un fichier.
chgrp	permet de modifier le groupe d'un fichier.
chown	permet de modifier le propriétaire d'un fichier.