

Droits d'accès

D. Bogdaniuk – P. Portejoie

GICQUEL Antoine

23/09/2016

1A2

RAPPELS

A chaque fichier est associé un ensemble d'indicateurs précisant qui peut y accéder et selon quel mode. Pour chaque fichier, il existe 3 types d'utilisateurs :

- le propriétaire du fichier (**u** pour user),
- les membres du groupe du propriétaire du fichier (**g** pour group),
- les autres utilisateurs du système (**o** pour others).
- l'ensemble des types (**a** pour all)

Pour chaque fichier et par type d'utilisateur, il existe 3 modes d'accès :

- autorisation de lecture (**r** pour read),
- autorisation d'écriture (**w** pour write),
- autorisation d'exécution (**x** pour execute).

En outre, il existe plusieurs types de fichiers :

- - pour fichier ordinaire,
- **d** pour répertoire (directory)
- **c** pour un fichier de périphérique en mode caractère,
- **b** pour un fichier de périphérique en mode bloc,
- **p** pour un tube nommé (named pipe),
- **l** pour un lien symbolique,
- **s** pour un socket (prise).

A noter que pour un répertoire, les modes d'accès ont un sens légèrement différent :

- **r** autorise à lister le contenu du répertoire par la commande `ls`,
- **w** autorise à créer des fichiers dans le répertoire,
- **x** autorise à accéder au répertoire par la commande `cd`.

Les droits sont visualisables par la commande `ls`, avec le paramètre `-l` (pour format long) :

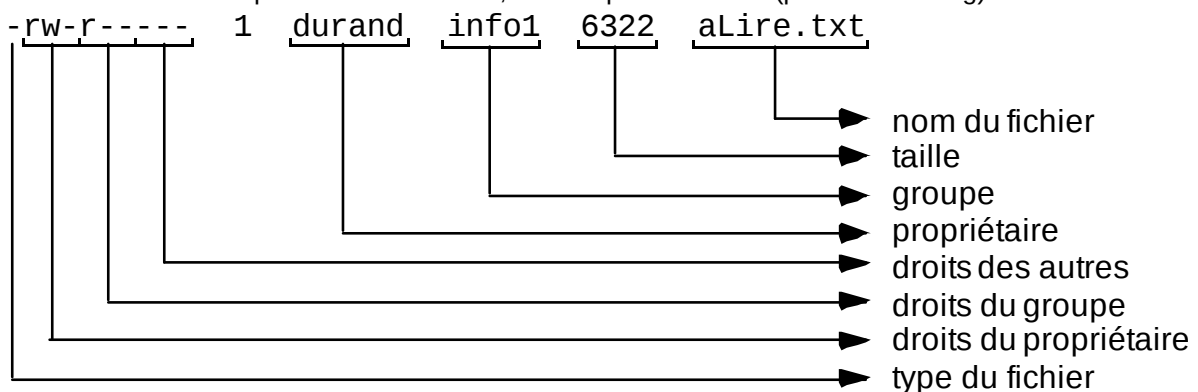


Fig 0 - Exemple de droits

Les droits d'accès à un fichier ne peuvent être modifiés que par le propriétaire du fichier à l'aide de la commande **chmod**. Il existe deux manières de spécifier des nouveaux droits : soit sous forme d'un nombre en octal (méthode numérique), soit sous forme d'une description des droits d'accès (méthode symbolique).

Description des droits d'accès

Le droit est spécifié de la manière suivante :

<qui> <opération> <permission>

qui	est une combinaison de u pour le propriétaire, g pour le groupe, o pour les autres, ou a (all) pour tous,
opération	vaut + pour ajouter des droits, - pour en supprimer, = pour en affecter,
permission	est une combinaison de r pour la lecture, w pour l'écriture, x pour l'exécution.

L'exemple précédent d'attribution de droits au fichier `resume.txt` peut se présenter de la façon suivante :

```
chmod u+rw resume.txt
chmod g=rx resume.txt
chmod o=r resume.txt
```

ou, en une seule instruction :

```
chmod u+rw,g=rx,o=r resume.txt
```

Cas particulier

L'utilisateur a toujours le droit de supprimer un fichier dans son espace personnel, même s'il ne dispose pas explicitement des droits nécessaires.

Représentation des droits d'accès par un nombre octal (cf cours pour transcodage binaire-octal)

La présence d'un droit est désignée par un 1, alors que son absence l'est par un 0. Par exemple, `rwxr-xr--` est désigné par la suite de bits `111101100` qui se traduit en octal par `754`. Donc pour donner au fichier `resume.txt` les droits précédents il suffit de lancer la commande `chmod 754 resume.txt`.

Exercice 1

Pour chaque entrée du tableau, déterminez les types et les droits d'accès des utilisateurs :

<code>-rwx----- essai</code>	Fichier : lire, écrire et exécuter pour le propriétaire. Aucun droit pour le groupe et les autres.
<code>-rw-r--r-- resume.txt</code>	Fichier : lire et écrire pour le propriétaire. Lire pour le groupe et les autres.
<code>drwxr-x--- achats</code>	Répertoire: lister, lire, écrire et exécuter pour le propriétaire. Exécuter pour le groupe.
<code>drwxr-x--x bilan</code>	Répertoire: lister, lire, écrire et exécuter pour le propriétaire. Exécuter pour le groupe et les autres.
<code>lrwxrw-rw- raccourci</code>	Lien symbolique : lire, écrire et exécuter pour le propriétaire. Lire et écrire pour le groupe et les autres.
<code>prw-rw-rw- communication</code>	Pipe : lire et écrire pour tout les utilisateurs.

Exercice 2

Observez et commentez les résultats des 2 commandes lancées ci-après :

```
>whoami
durand@univ-ubs.fr
>ls -l
-rwx----- dupont codage
-r-xr--r-- durand pgcd
dr----- durand reserve
-rwxr--r-- durand resume.txt
----- durand secret.txt
d----- durand vide
drwx--x--x dupont bilan
```

→ L'utilisateur courant est durand et dans le répertoire courant il y a des fichiers appartenant à durand mais aussi d'autres fichiers appartenant à un autre utilisateur nommé dupont.

Pour chacune des commandes suivantes, indiquez son résultat dans le cas général, puis dans le cas particulier des conditions indiquées ci-avant, en justifiant à chaque fois votre réponse.

```
>cat resume.txt
```

```
>more secret.txt
```

```
>cat pgcd
```

```
>./pgcd
```

```
>./codage
```

```
>chmod 777 codage  
>ls reserve  
>cp resume.txt reserve  
>ls vide  
>ls -ld vide  
>cp resume.txt pgcd  
>rm pgcd  
>rm codage  
>ls bilan  
>cd bilan  
>cd .. ; rmdir bilan
```

Exercice 3

Quels sont les droits donnés au fichier `essai` par les commandes suivantes :

<code>chmod 700 essai</code>	u : rwx g : rien o : rien
<code>chmod 601 essai</code>	u : rw g : rien o : x
<code>chmod 754 essai</code>	u : rwx g : rx o : r
<code>chmod 234 essai</code>	u : w g : wx o : r

NB : la table de codage binaire-octale est donnée dans le cours

Fin de la première séance de TD

En TP (TP2, 1^{ère} séance du TD-TP Droits) :

Pour réaliser les exercices en TP, vous devez recopier dans votre espace personnel certains fichiers nécessaires que vous trouverez dans

[/ubs/forum/prof/1tin01/ASR/M1101/tpsUNIX](#)

Créez ceux qui n'y seraient pas par la commande touch.

Exercice préliminaire

Allez dans le forum qui vous est dédié (1tin01), puis tapez ls -l. Repérez les fichiers éventuels dont vous êtes propriétaire et supprimez-les.

```
[e1600718@ens-iutva-0027 ~]$ cd /ubs/forum/etud/1tin01
[e1600718@ens-iutva-0027 1tin01]$ ls -l
total 1516
lrwxrwxrwx. 1 e1600174 1tin01      28 14 sept. 11:05 1tin01 -> /ubs/home/etud/2016/e1600174
lrwxrwxrwx. 1 e1600118 1tin01      28 12 sept. 17:07 e1600118 -> /ubs/home/etud/2016/e1600118
lrwxrwxrwx. 1 e1600122 1tin01      28 13 sept. 12:55 e1600122 -> /ubs/home/etud/2016/e1600122
lrwxrwxrwx. 1 e1600939 1tin01       9 13 sept. 15:07 e1600939 -> /e1600939
drwxr-sr-x. 2 e1601354 1tin01    4096 13 sept. 14:49 e1601354
lrwxrwxrwx. 1 e1601783 1tin01      28 13 sept. 12:56 e1601783 -> /ubs/home/etud/2016/e1601783
lrwxrwxrwx. 1 e1602535 1tin01      28 13 sept. 15:30 e1602535 -> /ubs/home/etud/2016/e1602535
lrwxrwxrwx. 1 e1602661 1tin01      28 16 sept. 15:31 e1602661 -> /ubs/home/etud/2016/e1602661
lrwxrwxrwx. 1 e1604207 1tin01      28 16 sept. 08:29 e1604207 -> /ubs/home/etud/2016/e1604207
drwxr-sr-x. 2 e1604677 1tin01    4096 23 sept. 08:39 e1604677
lrwxrwxrwx. 1 e1600019 1tin01       5 12 sept. 17:06 essai -> essai
lrwxrwxrwx. 1 e1600019 1tin01       9 12 sept. 17:05 essai.txt -> essai.txt
lrwxrwxrwx. 1 e1600063 1tin01       5 16 sept. 16:35 forum -> forum
lrwxrwxrwx. 1 e1600202 1tin01      39 15 sept. 15:08 liens1 -> ../../../../../../ubs/forum/etud/1tin01
drwxr-sr-x. 2 e1502566 1tin01    4096 13 sept. 12:06 m3301
drwxr-sr-x. 3 e1602408 1tin01    4096 14 sept. 14:27 niv1
-rwxrwxr-t+ 1 e1404399 1tin01 1524279 6 sept. 11:32 Polycopie_01_Denombrement.pdf
-rw-r--r--. 1 e1602477 1tin01       0 16 sept. 09:56 rm
-rw-r--r--. 1 e1602485 1tin01       0 13 sept. 14:54 tmpHamon.txt
drwxr-sr-x. 2 e1604677 1tin01    4096 16 sept. 08:18 tmpLeDrevo.txt
drwxr-sr-x. 2 e1604677 1tin01    4096 16 sept. 09:17 tmpLeDrevo.txt
-rw-r--r--. 1 e1600314 1tin01       0 13 sept. 14:24 tmps.txt
lrwxrwxrwx. 1 e1600019 1tin01       9 12 sept. 17:04 tmp.txt -> essai.txt
[e1600718@ens-iutva-0027 1tin01]$ ls -l | grep e1600718 | wc -l
0
[e1600718@ens-iutva-0027 1tin01]$ OK
```

Exercice 1

Complétez le tableau

OK

Exercice 2

Afin de pouvoir mettre en pratique cet exercice vous devez créer dans votre environnement un ensemble de fichiers et de répertoires à l'image de ceux donnés dans l'exemple de l'exercice 2 vu en TD, avec au final les mêmes droits. Pour ce faire, vous devez considérer que vous êtes à la place de *durand*. Cependant, vous ne pourrez pas prendre le rôle de *dupont* pour créer les fichiers lui appartenant : vous devrez faire appel à un autre étudiant et vous rendre mutuellement ce service.....

1. Exécutez, puis commentez la suite de commandes suivantes : `cd ; ls ; ls -l ; ls -ld`

```
[e1600718@ens-iutva-0027 ltin01]$ cd
[e1600718@ens-iutva-0027 ~]$ ls
Bureau          Musique          'TP n°2 - Algo et Prog'
Documents       'PREZ CONTRAT.odt' 'TP n°2 - Algo et Prog.zip'
firefox         'PREZ PPP Antoine G.odt' TPs
gradle-app.setting 'PREZ PPP Maxime H.odt' tpsUNIX
Images          Téléchargements Vidéos
'Mes documents' 'TP n°1 - Algo et Prog'
[e1600718@ens-iutva-0027 ~]$ ls -l
total 5992
drwxr-xr-x. 2 e1600718 etud 4096 2 sept. 08:15 Bureau
drwxr-xr-x. 2 e1600718 etud 4096 23 sept. 13:08 Documents
drwx----- 3 e1600718 etud 4096 2 sept. 08:03 firefox
-rw-r--r--. 1 e1600718 etud 883 2 sept. 12:31 gradle-app.setting
drwxr-xr-x. 2 e1600718 etud 4096 2 sept. 08:38 Images
drwx----- 9 e1600718 etud 4096 21 sept. 09:51 'Mes documents'
drwxr-xr-x. 2 e1600718 etud 4096 2 sept. 08:15 Musique
-rw-r--r--. 1 e1600718 etud 1182444 21 sept. 14:57 'PREZ CONTRAT.odt'
-rw----- 1 e1600718 etud 1181477 20 sept. 13:31 'PREZ PPP Antoine G.odt'
-rw----- 1 e1600718 etud 1181387 20 sept. 13:31 'PREZ PPP Maxime H.odt'
drwxr-xr-x. 2 e1600718 etud 4096 9 sept. 17:05 Téléchargements
drwxr-xr-x. 2 e1600718 etud 4096 9 sept. 17:12 'TP n°1 - Algo et Prog'
drwxrwxr-x. 2 e1600718 etud 4096 16 sept. 16:01 'TP n°2 - Algo et Prog'
-rw-r--r--. 1 e1600718 etud 2533650 16 sept. 14:01 'TP n°2 - Algo et Prog.zip'
drwxr-xr-x. 3 e1600718 etud 4096 7 sept. 14:42 TPs
lrwxrwxrwx. 1 e1600718 etud 41 14 sept. 14:50 tpsUNIX -> /ubs/forum/prof/lti
n01/ASR/M1101/tpsUNIX/
drwxr-xr-x. 2 e1600718 etud 4096 2 sept. 08:15 Vidéos
[e1600718@ens-iutva-0027 ~]$ ls -ld
drwx----- 30 e1600718 etud 4096 23 sept. 12:55 .
[e1600718@ens-iutva-0027 ~]$ |
```

→ Une option peut totalement changé le comportement d'une commande :

`cd` : sans argument, elle nous replace dans notre répertoire d'accueil

`ls` : sans argument, il liste les fichiers et répertoires par ordre alphabétique

`ls -l` : argument « long », il liste les fichiers et répertoires par ordre alphabétique et en liste avec davantage d'informations comme les droits d'accès, le propriétaire, la dernière date d'accès etc

`ls -ld` : identique à la précédente mais avec l'option « directory » et donc affiche les répertoires avec la même présentation que les fichiers, sans lister leur contenu.

2. Créez un répertoire *TP2* sous votre répertoire d'accueil

3. Créez dans ce répertoire tous les fichiers appartenant à *durand* (respectez bien leur type) puis positionnez pour chacun les droits comme indiqués dans l'exemple de l'exercice 2 du TD

```
[e1600718@ens-iutva-0027 ~]$ mkdir TP2
[e1600718@ens-iutva-0027 ~]$ cd TP2/
[e1600718@ens-iutva-0027 TP2]$ ls
[e1600718@ens-iutva-0027 TP2]$ touch pgcd resume.txt secret.txt && mkdir reserve
vide
[e1600718@ens-iutva-0027 TP2]$ ls -l
total 8
-rw-r--r--. 1 e1600718 etud    0 23 sept. 13:20 pgcd
drwxr-xr-x. 2 e1600718 etud 4096 23 sept. 13:20 reserve
-rw-r--r--. 1 e1600718 etud    0 23 sept. 13:20 resume.txt
-rw-r--r--. 1 e1600718 etud    0 23 sept. 13:20 secret.txt
drwxr-xr-x. 2 e1600718 etud 4096 23 sept. 13:20 vide
[e1600718@ens-iutva-0027 TP2]$ chmod 544 pgcd
[e1600718@ens-iutva-0027 TP2]$ chmod 400 reserve/
[e1600718@ens-iutva-0027 TP2]$ chmod 744 resume.txt
[e1600718@ens-iutva-0027 TP2]$ chmod 000 secret.txt vide/
[e1600718@ens-iutva-0027 TP2]$ ls -l
total 8
-r-xr--r--. 1 e1600718 etud    0 23 sept. 13:20 pgcd
dr-----. 2 e1600718 etud 4096 23 sept. 13:20 reserve
-rwxr--r--. 1 e1600718 etud    0 23 sept. 13:20 resume.txt
-----. 1 e1600718 etud    0 23 sept. 13:20 secret.txt
d-----. 2 e1600718 etud 4096 23 sept. 13:20 vide
[e1600718@ens-iutva-0027 TP2]$ OK
```

4. Demandez à un autre étudiant de créer le fichier *codage* et le répertoire *bilan* dans votre répertoire TP2 puis d'y positionner les droits comme indiqués dans l'exemple de l'exercice 2 du TD (au préalable, vous aurez dû, pour ce faire, positionner les droits comme indiqués dans l'exemple de l'exercice 2 du TD, cf précisions données ci-après.

```
[e1600718@ens-iutva-0027 TP2]$ cd /ubs/home/etud/2016/e1602408/TP2/
[e1600718@ens-iutva-0027 TP2]$ touch ./codage && mkdir ./bilan
[e1600718@ens-iutva-0027 TP2]$ chmod 700 codage
[e1600718@ens-iutva-0027 TP2]$ chmod 711 bilan/
[e1600718@ens-iutva-0027 TP2]$ ls -l
total 12
drwx--x--x. 2 e1600718 etud 4096 23 sept. 13:49 bilan
-rwx-----. 1 e1600718 etud    0 23 sept. 13:49 codage
-r-xr--r--. 1 e1602408 etud    0 23 sept. 13:37 pgcd
dr-----. 2 e1602408 etud 4096 23 sept. 13:36 reserve
-rwxr--r--. 1 e1602408 etud    0 23 sept. 13:37 resume.txt
-----. 1 e1602408 etud    0 23 sept. 13:37 secret.txt
d-----. 2 e1602408 etud 4096 23 sept. 13:36 vide
```

5. Donnez une copie d'écran du contenu final du répertoire TP2


```
[e1600718@ens-iutva-0027 TP2]$ chmod 777 ~/ .
[e1600718@ens-iutva-0027 TP2]$ ls -ld
drwxrwxrwx. 4 e1600718 etud 4096 23 sept. 13:20 .
[e1600718@ens-iutva-0027 TP2]$ ls -l
total 8
-r-xr--r--. 1 e1600718 etud    0 23 sept. 13:20 pgcd
dr----- 2 e1600718 etud 4096 23 sept. 13:20 reserve
-rwxr--r--. 1 e1600718 etud    0 23 sept. 13:20 resume.txt
----- 1 e1600718 etud    0 23 sept. 13:20 secret.txt
d----- 2 e1600718 etud 4096 23 sept. 13:20 vide
[e1600718@ens-iutva-0027 TP2]$ pwd
/ubs/home/etud/2016/e1600718/TP2
[e1600718@ens-iutva-0027 TP2]$ ls -l
total 12
drwx--x--x. 2 e1602408 etud 4096 23 sept. 13:53 bilan
-rwx----- 1 e1602408 etud    0 23 sept. 13:53 codage
-r-xr--r--. 1 e1600718 etud    0 23 sept. 13:20 pgcd
dr----- 2 e1600718 etud 4096 23 sept. 13:20 reserve
-rwxr--r--. 1 e1600718 etud    0 23 sept. 13:20 resume.txt
----- 1 e1600718 etud    0 23 sept. 13:20 secret.txt
d----- 2 e1600718 etud 4096 23 sept. 13:20 vide
[e1600718@ens-iutva-0027 TP2]$ OK
```

6. Testez les différentes commandes illustrées à l'exercice 2 du TD (reportez-y vos résultats)


```
[e1600718@ens-iutva-0027 TP2]$ cat resume.txt
[e1600718@ens-iutva-0027 TP2]$ more secret.txt
more: impossible d'ouvrir secret.txt: Permission denied
[e1600718@ens-iutva-0027 TP2]$ cat pgcd
[e1600718@ens-iutva-0027 TP2]$ ./pgcd
[e1600718@ens-iutva-0027 TP2]$ ./codage
bash: ./codage: Permission denied
[e1600718@ens-iutva-0027 TP2]$ chmod 777 codage
chmod: modification des droits de 'codage': Operation not permitted
[e1600718@ens-iutva-0027 TP2]$ ls reserve/
[e1600718@ens-iutva-0027 TP2]$ cp resume.txt reserve
cp: impossible d'évaluer 'reserve/resume.txt': Permission denied
[e1600718@ens-iutva-0027 TP2]$ ls vide/
ls: impossible d'ouvrir le répertoire 'vide/': Permission denied
[e1600718@ens-iutva-0027 TP2]$ ls -ld vide/
d----- . 2 e1600718 etud 4096 23 sept. 13:20 vide/
[e1600718@ens-iutva-0027 TP2]$ cp resume.txt pgcd
cp: impossible de créer le fichier standard 'pgcd': Permission denied
[e1600718@ens-iutva-0027 TP2]$ rm pgcd
rm : supprimer fichier vide (protégé en écriture) 'pgcd' ? o
[e1600718@ens-iutva-0027 TP2]$ rm codage
rm : supprimer fichier vide (protégé en écriture) 'codage' ? o
[e1600718@ens-iutva-0027 TP2]$ cd bilan/
[e1600718@ens-iutva-0027 bilan]$ cd .. ; rmdir bilan
[e1600718@ens-iutva-0027 TP2]$ |
```

Quelques précisions :

- Vous devez modifier les droits sur votre répertoire d'accueil. Faites attention à la manière dont est redirigé celui-ci, il peut s'agir d'un lien symbolique. Dans ce cas ce n'est pas sur le lien qu'il faut modifier les droits, mais bien sur le répertoire lui-même (suivez le lien !)
- Chaque répertoire d'accueil d'étudiant a pour parent un répertoire identifié par un chiffre qui correspond au dernier chiffre du login (exemple e1400849 a pour parent le répertoire 9). Votre binôme n'appartient donc pas forcément au même répertoire que vous, pensez à en tenir compte pour les chemins relatifs
- Pour qu'un autre utilisateur du même groupe que vous puisse disposer de fichiers ou de répertoires situés dans votre répertoire d'accueil (home directory), vous devez préalablement lui (plus précisément à son groupe) accorder les droits rwx sur celui-ci par la commande appropriée

ATTENTION : n'oubliez pas, dès que possible, de remettre les droits initiaux sur votre répertoire d'accueil par la commande `chmod 700 ~`

Exercice 3

Complétez le tableau après avoir testé chacune des commandes

OK

Fin de la première séance de TP

Exercice 4

En utilisant le format octal (cf cours), indiquez la commande permettant de donner au fichier `essai` les droits suivants :

<code>rw-rwxrwx</code>	<code>chmod 777 essai</code>
<code>rw-r-----</code>	<code>chmod 640 essai</code>
<code>r-xr--r--</code>	<code>chmod 544 essai</code>
<code>r--r-xrwx</code>	<code>chmod 457 essai</code>

Exercice 5

Quels sont les droits donnés au fichier `essai` par les commandes suivantes (on considère que les commandes sont données les unes après les autres) :

<code>chmod a-rwx</code>	<code>essai</code>	<code>000</code>
<code>chmod u+r,go+x</code>	<code>essai</code>	<code>411</code>
<code>chmod g+w,o=w</code>	<code>essai</code>	<code>462</code>
<code>chmod u+x,go-x</code>	<code>essai</code>	<code>562</code>

Exercice 6

Donnez la commande permettant d'attribuer au fichier `essai` les droits précisés ci-après, en utilisant la méthode symbolique.

Pour le premier cas on envisagera 2 hypothèses :

- 1/ on démarre de la situation de l'exercice précédent,
- 2/ on part d'une situation initiale inconnue.

On considère que les commandes suivantes sont données les unes après les autres :

<code>rwX-----</code>	1/ <code>chmod u+w,go-rwx essai</code> 2/ <code>chmod 700 essai</code>
<code>---r--r--</code>	<code>chmod u-rwx,go=r essai</code>
<code>---r-x---</code>	<code>chmod g+x,o-r essai</code>
<code>r--rw-r-x</code>	<code>chmod u+r,g=rw,o=rx essai</code>
<code>-----</code>	<code>chmod a-rwx essai</code>

Exercice 7**Droits à la création, principe**

A sa création un fichier est doté de droits par défaut. Il est possible de visualiser le masque de positionnement de ces droits par la commande `umask` qui rend alors une valeur octale. La valeur octale, traduite en binaire, décrit les droits par défaut :

- 0 indique la présence du droit,
- 1 indique l'absence du droit.

Par exemple, **22 (022)** se traduit en binaire **000010010**, c'est à dire **rwxr-xr-x**. Cependant, attention, un fichier ordinaire n'est jamais créé avec le droit x, sauf s'il l'est par un compilateur ; l'utilisateur peut bien entendu les y rajouter.

La commande **umask** suivie d'une valeur octale permet de modifier le masque de positionnement des droits par défaut (et donc ceux-ci).

Questions

- 1/ Quels sont les droits par défaut à la création d'un fichier quand la commande **umask** rend **056** ?

user	rwX
group	-W-
others	--X

- 2/ Comment définir les droits **rw----r--** par défaut ?
> **umask 173**

Exercice 8

Retour sur la représentation des droits d'accès par un nombre octal

Un premier chiffre peut être ajouté pour spécifier les bits setUID, setGID et sTicky bit : 4 pour setUID, 2 setGID et 1 pour sTicky bit. Pour des raisons historiques, l'affichage de ces différents bits se fait en lieu et place du droit d'exécution. Celui-ci, lorsqu'il est présent, est remplacé par s, S et t. Lorsqu'il est absent, il est remplacé par S, S et T.

Sticky-bit

Droit représenté en notation symbolique par un t à la place du x de other (x doit être positionné, sinon T indique l'absence du droit) ou **1000** en notation octale. Il peut être affecté à un exécutable ou à un répertoire. Exemple : **chmod +t /tmp** (droit t à la place du x de other)

A l'origine il servait à maintenir un exécutable en mémoire. Aujourd'hui il sert principalement aux répertoires. Il contrôle la destruction des fichiers qui se trouvent dans un répertoire, en limitant cette action au propriétaire du fichier. Par exemple dans le répertoire /tmp tout le monde peut écrire grâce au droit w accordé à tous ; sans le droit t, chacun pourrait détruire les fichiers des autres sans en être propriétaire.

Droits d'endossement

- **SUID-bit** (Set User ID bit)

Droit représenté en notation symbolique par un s à la place du x de user (x doit être positionné, sinon S indique l'absence du droit) ou 4000 en notation octale. Exemple :
`chmod u+sessai`

L'utilisateur qui exécute un logiciel prend temporairement l'identité de son propriétaire ; pendant toute l'exécution l'utilisateur change d'UID mais pas de groupe d'appartenance. Quand l'exécution est terminée, l'utilisateur reprend son identité d'origine.

- **SGID-bit** (Set Group ID bit)

Droit représenté en notation symbolique par un s à la place du x de group (x doit être positionné, sinon S indique l'absence du droit) ou 2000 en notation octale. Exemple :
`chmod g+ssessai`

L'utilisateur qui exécute le logiciel conserve son identité mais change de groupe actif durant l'exécution du logiciel sur lequel est positionné ce droit ; quand l'exécution est terminée l'utilisateur retrouve son groupe d'origine.

Questions

- 1/ Quels seront les droits affectés au fichier *essai* par la commande donnée ci-dessous ?

```
chmod 2700essai
```

```
rwX-S---
```

- 2/ Même question avec la commande suivante :

```
chmod 1700essai
```

```
rwX-----T
```

- 3/ Même question avec la commande suivante :

```
chmod 4700essai
```

```
rws-----
```

- 4/ Dans le dernier cas, donnez la (les) commande(s) permettant de positionner le droit s pour le groupe

```
> chmod g+xsessai
```

```
[e1600718@ens-iutva-0392 e1600718]$ chmod 000essai
[e1600718@ens-iutva-0392 e1600718]$ chmod g+xsessai
[e1600718@ens-iutva-0392 e1600718]$ ls -lessai
-----s---. 1 e1600718 etud 0 30 sept. 14:09essai
```

- 5/ Enfin, donnez la (les) commande(s) permettant d'y positionner le sTicky bit

> `chmod o-x essai`

> `chmod o+t essai`

→ 6/ Quels sont les droits affectés par `chmod 700 essai` ?

`rwX-----`

En TP (TP3, 2^{ème} séance du TD-TP Droits) :

Pour réaliser les exercices en TP, vous devez recopier dans votre espace personnel certains fichiers nécessaires que vous trouverez dans

</ubs/forum/prof/1tin01/ASR/M1101/tpsUNIX>

Créez ceux qui n'y seraient pas par la commande touch.

Exercices 4 à 6

Complétez les tableaux des exercices 4 à 6 en testant chaque commande ; pour chaque exercice donnez une copie des résultats.

Pour le 4 :

```
[e1600718@ens-iutva-0392 e1600718]$ chmod 777 essai
[e1600718@ens-iutva-0392 e1600718]$ ls -l essai
-rwxrwxrwx. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod 640 essai
[e1600718@ens-iutva-0392 e1600718]$ ls -l essai
-rw-r-----. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod 544 essai
[e1600718@ens-iutva-0392 e1600718]$ ls -l essai
-r-xr--r--. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod 457 essai
```

Pour le 5 :

```
[e1600718@ens-iutva-0392 e1600718]$ chmod a-rwx essai
[e1600718@ens-iutva-0392 e1600718]$ ls -l essai
-----x. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod u+r,go+x essai
[e1600718@ens-iutva-0392 e1600718]$ ls -l essai
-r---x--x. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod g+x,o=w essai
[e1600718@ens-iutva-0392 e1600718]$ ls -l essai
-r---x-w-. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod u+x,go-x essai
[e1600718@ens-iutva-0392 e1600718]$ ls -l essai
-r-x---w-. 1 e1600718 etud 0 30 sept. 14:09 essai
```


Pour le 6 :

```
[e1600718@ens-iutva-0392 e1600718]$ chmod 700 essai && ls -l essai
-rwx-----. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod u-rwx,go=r essai && ls -l essai
---r--r--. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod g+x,o-r essai && ls -l essai
---r-x---. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod u+r,g=rw,o=rx essai && ls -l essai
-r--rw-r-x. 1 e1600718 etud 0 30 sept. 14:09 essai
[e1600718@ens-iutva-0392 e1600718]$ chmod a-rwx essai && ls -l essai
-----. 1 e1600718 etud 0 30 sept. 14:09 essai
```

Exercice 7

Reportez les réponses données en TD, mais ne les testez pas.

Exercice 8

Reportez les réponses données en TD, en testant et vérifiant chaque commande ; donnez une copie des résultats.

Exercice 9

Répondez aux différentes questions en vérifiant et commentant à chaque fois les résultats et en les illustrant par des copies appropriées.

- Grâce à la commande appropriée indiquez le masque utilisé à la création de tout fichier dans votre espace personnel et déduisez-en les droits par défaut d'un fichier ou d'un répertoire lors de leur création (ne tenez pas compte du 0 de gauche)

```
[e1600718@ens-iutva-0392 e1600718]$ umask
0022
```

022 == ---w---w- donc les droits par défauts sont rwxr-xr-x

- Lancez la séquence de commandes donnée ci-dessous, commentez :
`chmod 2700 ~ ; ls -ld`

```
[e1600718@ens-iutva-0392 ~]$ chmod 2700 ~/ ; cd ; ls -ld
drwx--S---. 40 e1600718 etud 4096 30 sept. 14:09 .
```

la commande `chmod 2700 ~/` change les droits de notre répertoire d'accueil, la commande `cd` sans argument nous replace dans notre répertoire d'accueil puis `ls -ld` sans option supplémentaire affiche les droits de notre répertoire courant c'est à dire le répertoire d'accueil.

On remarque que SGID-bit a été ajouté et les droits pour g et o ont été enlevés.

- Créez un répertoire **SYS** dans votre répertoire d'accueil
- Visualisez les droits d'accès de ce nouveau répertoire. Sont-ils conformes à vos attentes ? Pourquoi ?

```
[e1600718@ens-iutva-0392 ~]$ mkdir ~/SYS
[e1600718@ens-iutva-0392 ~]$ ls -ld ~/SYS/
drwxr-sr-x. 2 e1600718 etud 4096 30 sept. 14:38 /ubs/home/etud/2016/e1600718/SYS/
```

Le nouveau répertoire possède les droits par défauts (voir umask) mais conserve le droit en s. Cela paraît étrange car un fichier n'est pas censé hériter des droits de son parents.

- Déplacez-vous dans ce répertoire
- Créez-y un fichier **essai.txt** (commande **touch**)
- Visualisez les droits d'accès de ce nouveau fichier. Sont-ils conformes à vos attentes ? Pourquoi ?

```
[e1600718@ens-iutva-0392 ~]$ cd ~/SYS/
[e1600718@ens-iutva-0392 SYS]$ touch essai.txt
[e1600718@ens-iutva-0392 SYS]$ ls -l
total 0
-rw-r--r--. 1 e1600718 etud 0 30 sept. 14:43 essai.txt
```

Le fichier a perdu le droit en s mais aussi en x. Cela paraît étrange car le fichier n'a pas les droits spécifiés par le masque.

- Modifiez les droits du répertoire SYS afin d'en supprimer le S

```
[e1600718@ens-iutva-0392 ~]$ chmod -s ~/SYS/ && ls -ld ~/SYS/
drwxr-xr-x. 2 e1600718 etud 4096 30 sept. 14:43 /ubs/home/etud/2016/e1600718/SYS/
```

- Copiez le fichier **essai.c** qui se trouve dans /ubs/forum/prof/1tin01/ASR/M1101/tpsUNIX

```
[e1600718@ens-iutva-0392 ~]$ cp /ubs/forum/prof/1tin01/ASR/M1101/tpsUNIX/
bippeur.tar.gz  demo          pgcd          ques          version1
compilateur     essai.c         prog.c        Timbres.txt
conf            liste.html      ps            version
[e1600718@ens-iutva-0392 ~]$ cp /ubs/forum/prof/1tin01/ASR/M1101/tpsUNIX/essai.c ~/
[e1600718@ens-iutva-0392 ~]$ ls -l /ubs/forum/prof/1tin01/ASR/M1101/tpsUNIX/essai.c
-rwxr--r--. 1 portejoi prof 171 27 sept. 11:58 /ubs/forum/prof/1tin01/ASR/M1101/tpsUNIX/essai.c
[e1600718@ens-iutva-0392 ~]$ ls -l ~/essai.c
-rwxr--r--. 1 e1600718 etud 171 30 sept. 14:47 /ubs/home/etud/2016/e1600718/essai.c
[e1600718@ens-iutva-0392 ~]$ cat ~/essai.c
#include <stdio.h>
int main()
{
printf("Bonjour\n");
printf("Vous venez de lancer la commande compilateur\n");
printf("en lui passant les paramètres avec succès!\n");
}
```

- Visualisez les droits du fichier **essai.c** qui se trouve dans le forum prof et comparez-les avec ceux du fichier que vous venez de copier. Commentez

Les droits sont conservés mais mon utilisateur est devenu le propriétaire de cette copie/

- Visualisez le fichier `essai.c` ; Que contient-il ?

Il contient du code en langage C.

- Exécutez la commande `cc essai.c -o essai` (cc est la commande du compilateur C++)

```
[e1600718@ens-iutva-0392 ~]$ cc essai.c -o essai
[e1600718@ens-iutva-0392 ~]$ ls -l ~/essai
-rwxr-xr-x. 1 e1600718 etud 8624 30 sept. 14:51 /ubs/home/etud/2016/e1600718/essai
[e1600718@ens-iutva-0392 ~]$ ~/essai
Bonjour
Vous venez de lancer la commande compilateur
en lui passant les paramètres avec succès!
```

- Visualisez les droits d'accès du fichier `essai` créé. Sont-ils conformes ? Pourquoi ?

Les droits en x ce sont créés. Cela est conforme car on vient de créer un nouveau fichier exécutable donc il est plus simple d'avoir directement les droits pour l'exécuter.

- Lancez l'exécution du fichier par `./essai`
- Supprimez le droit d'exécution sur `essai`

```
[e1600718@ens-iutva-0392 ~]$ chmod -x ~/essai
[e1600718@ens-iutva-0392 ~]$ ~/essai
bash: /ubs/home/etud/2016/e1600718/essai: Permission denied
```

- Lancez l'exécution du fichier par `./essai`. Que se passe-t-il ? Pourquoi ?

L'exécution nous est refusé car nous n'avons plus les droits nécessaires.

- Remettez le droit d'exécution, mais supprimez celui de lecture sur `essai`

```
[e1600718@ens-iutva-0392 ~]$ chmod +x,-r ~/essai
[e1600718@ens-iutva-0392 ~]$ ~/essai
Bonjour
Vous venez de lancer la commande compilateur
en lui passant les paramètres avec succès!
```

- Lancez l'exécution du fichier par `./essai`. Que se passe-t-il ? Pourquoi ?

Le programme s'exécute car pour exécuter un programme, le droit en x suffit, le droit en r n'est pas nécessaire.

Exercice 10

Répondez aux différentes questions en vérifiant et commentant à chaque fois les résultats et en les illustrant par des copies appropriées.

- Placez-vous dans le répertoire père de votre répertoire d'accueil

- Quels sont vos droits sur les répertoires d'accueil des autres utilisateurs (attention : ne confondez pas les liens symboliques avec les répertoires) ?

Mon utilisateur n'a aucun droit sur les répertoires d'accueil des autres utilisateurs.

- Démontrez vos propos précédents par les commandes adaptées

```
[e1600718@ens-iutva-0392 2016]$ cd && cd ..
[e1600718@ens-iutva-0392 2016]$ whoami
e1600718
[e1600718@ens-iutva-0392 2016]$ ls -l * | grep e160071
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600712 -> /ubs/fukuisaurus/home.1/2/e1600712
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600714 -> /ubs/fukuisaurus/home.1/4/e1600714
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600715 -> /ubs/fukuisaurus/home.1/5/e1600715
lrwxrwxrwx. 1 root root 34 14 juil. 07:15 e1600716 -> /ubs/fukuisaurus/home.1/6/e1600716
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600717 -> /ubs/fukuisaurus/home.1/7/e1600717
lrwxrwxrwx. 1 root root 34 13 juil. 07:15 e1600718 -> /ubs/fukuisaurus/home.1/8/e1600718
[e1600718@ens-iutva-0392 2016]$ ls -l * /ubs/fukuisaurus/home.1/8/ | grep e160071
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600712 -> /ubs/fukuisaurus/home.1/2/e1600712
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600714 -> /ubs/fukuisaurus/home.1/4/e1600714
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600715 -> /ubs/fukuisaurus/home.1/5/e1600715
lrwxrwxrwx. 1 root root 34 14 juil. 07:15 e1600716 -> /ubs/fukuisaurus/home.1/6/e1600716
lrwxrwxrwx. 1 root root 34 7 juil. 07:15 e1600717 -> /ubs/fukuisaurus/home.1/7/e1600717
lrwxrwxrwx. 1 root root 34 13 juil. 07:15 e1600718 -> /ubs/fukuisaurus/home.1/8/e1600718
drwx--S---. 42 e1600718 etud 4096 30 sept. 14:51 e1600718
[e1600718@ens-iutva-0392 2016]$ ls -ld /ubs/fukuisaurus/home.1/7/e1600717
drwx-----. 2 e1600717 etud 4096 12 juil. 02:07 /ubs/fukuisaurus/home.1/7/e1600717
```

- Pourquoi est-il dangereux de ne pas avoir le droit x sur son répertoire d'accueil ?

Il est dangereux de supprimer le droit en x sur son répertoire d'accueil car nous avons besoin des droits en x pour se positionner dedans avec la commande cd. Donc si on a pas le droit en x, nous pourrions même s'y positionner.

- Créez un répertoire *essai* sur lequel vous mettrez tous les droits à l'utilisateur, aucun droit pour tous les autres. Vérifiez et illustrez par une copie d'écran

```
[e1600718@ens-iutva-0392 2016]$ ls -ld ~/essai/
drwx--S---. 2 e1600718 etud 4096 30 sept. 15:10 /ubs/home/etud/2016/e1600718/essai/
```

- Positionnez sur ce répertoire, en une seule commande, le droit s pour l'utilisateur et son groupe, le droit t pour les autres. Observez le résultat et commentez-le

```
[e1600718@ens-iutva-0392 2016]$ chmod +st ~/essai/
[e1600718@ens-iutva-0392 2016]$ ls -ld ~/essai/
drws--S--T. 2 e1600718 etud 4096 30 sept. 15:10 /ubs/home/etud/2016/e1600718/essai/
```

Les lettres sont en majuscules car ils n'ont pas les droits en x.

- Transformez le droit S en s et le droit T en t. Vérifiez et commentez en illustrant à chaque fois par des copies d'écran les commandes appropriées

```
[e1600718@ens-iutva-0392 2016]$ chmod go+x ~/essai/  
[e1600718@ens-iutva-0392 2016]$ ls -ld ~/essai/  
drws--s--t. 2 e1600718 etud 4096 30 sept. 15:10 /ubs/home/etud/2016/e1600718/essai/
```

Les lettres sont en minuscules car il possède à présent les droits en x.

En fin de TP, pensez à supprimer tous les fichiers et répertoires créés pour ce TP, et si vous êtes pointilleux, vous pouvez également supprimer le droit S sur votre répertoire d'accueil, par la commande appropriée

OK