TP n°10 – Gestion des données

Rappel : rédigez les manipulations de ce TP sur le wiki. Que faut-il noter : tout ce qui vous permettrait de refaire le TP entier en 15 minutes (temps d'attente des commandes en moins) avec le signalement de tous les pièges et problèmes rencontrés (ex : attention à ne pas oublier telle option...). Vous pouvez aussi rajouter des informations sur les commandes et leurs options utiles.

1) Montage de volumes

Cette partie concerne la gestion des disques durs. On va créer un nouveau disque virtuel et créer deux partitions dedans.

a) Préparation de la machine virtuelle

Avant de démarrer la machine virtuelle, créez un nouveau disque SATA virtuel, tout petit : 128Mo ou 0.1 Go. Ce nouveau disque devrait être associé au fichier spécial représentant le périphérique : `/dev/sdb`.

Comment le savoir ? Eh bien s'il est monté correctement dans la machine virtuelle, il suffit de taper la commande :

```
sudo fdisk -l
```

Elle affiche la liste des volumes, avec toutes leurs informations de formatage : liste des partitions et leurs tailles et types. Vous devez voir le disque `/dev/sda1` de 512Mo type Linux. Le disque que vous venez de rajouter doit être indiqué comme ne contenant aucune table des partitions valide, puisqu'il n'a pas été formaté. C'est à dire après la ligne I/O Size, il n'y a pas Disklabel type.

Dans la suite de l'énoncé, on prendra `/dev/sdb`, mais vous devrez modifier si ce n'est pas cela.

Regardez ce qu'est ce `/dev/sdb` avec la commande ls -l. Remarquez la première lettre : elle indique que c'est un périphérique en mode bloc.

b) Préparation

Installer le paquet `reiserfs` ou `xfs` (cf TP9).

```
sudo apt-get install reiserfsprogs xfsprogs
```

c) Formatage du disque /dev/sdb

Vous allez préparer le nouveau disque virtuel. Surtout ne vous trompez pas de disque, surtout pas `/dev/sda` qui contient votre système et vos fichiers, sinon vous avez gagné un billet gratuit pour refaire entièrement le TP8 dès maintenant et peut-être perdre votre Wiki avec. Un administrateur système est Dieu, il ne commet jamais aucune ereur.

```
sudo cfdisk /dev/sdb
```

Au tout début, il faut choisir le type des tables de partition, puisque le disque est vierge : descendez sur dos et validez (gpt est le nouveau système pour les bios UEFI, et les deux autres sont pour des systèmes Unix professionnels sur gros ordinateur).

Le menu de cfdisk est écrit en bas de la fenêtre : l'un des items est en surbrillance par exemple [NEW], vous changez de menu à l'aide du curseur gauche-droite et vous le sélectionnez à l'aide de la touche entrée. Le curseur haut-bas sert à changer de partition active.

Commencez par 1) créer la table des partitions : mettre sur [New] puis appuyer sur entrée, 2) créer deux partitions primaires dedans : la première de 80 Mo environ et la seconde de la taille restante (10 Mo ou plus). Tapez sur flèche descendante pour éditer la 2e partition. Ne mettre aucune des deux en « bootable ». Mémorisez les noms de ces partitions : `/dev/sdb1` et `/dev/sdb2` si c'est bien ce disque. Enregistrez les changements de partitions avec le menu [Write] tapez « yes », puis [Quit].

Vous allez maintenant formater ces deux nouvelles partitions. Encore une fois, attention, si vous vous trompez de disque, vous devrez refaire tout le TP8, sans compter votre Wiki. N'oubliez pas de mettre sudo devant les commandes.

- La première partition sera formatée en XFS ou ReiserFS, à vous de choisir, notez l'option -l ou -L pour spécifier le label du volume. Choisissez l'une des deux commandes suivantes :

```
mkfs.reiserfs -l HOME /dev/sdb1
mkfs.xfs -L HOME /dev/sdb1
```

  Si vous voulez reformater un volume déjà formaté, rajoutez l'option -f pour forcer l'action.

  Pour en savoir plus, cherchez « reiserfs xfs benchmark » sur un moteur de recherche. Ces formats de disque se distinguent par leurs performances sur différents types d'utilisation des disques : nombreuses écritures aléatoires, grands fichiers...

- La seconde partition sera formatée en ext4. C'est le format le plus fréquent sur Linux.

```
mkfs.ext4 -L DATA /dev/sdb2
```

Attention, c'est -L, pas -l pour la 2e commande…

Si vous voulez recommencer toute cette manip, voici comment effacer le MBR d'un seul coup. Comme toujours, si vous voulez refaire le TP8, trompez-vous de disque...

```
sudo dd if=/dev/zero ibs=512 count=1 of=/dev/sdb
```

Si vous voulez un outil très nettement plus amical, au lieu de `cfdisk`, après avoir fait startx, lancez `gparted`. C'est ce dernier qui est employé quand on installe Ubuntu, Debian ou Mint sur un disque dur. Il est très simple à utiliser.

```
sudo gparted /dev/sdb
```

d) Montage de ces partitions

On va commencer par monter les partitions dans un sous-dossier de `/mnt` puis on les mettra en place dans l'arborescence normale

i) Montage de la partition n°1 dans /mnt/home

Créer un dossier appelé `/mnt/home`, (n'oubliez pas sudo) puis employez la commande (toujours en tant que superutilisateur). Mettez xfs ou reiserfs selon ce que vous avez choisi lors du formatage.

```
sudo mount -t reiserfs /dev/sdb1 /mnt/home
```

Allez vérifier que le dossier `/mnt/home` est vide. C'est normal, la partition vient d'être formatée.

☞ Utilisez la commande `df -h` ou la commande `pydf` pour lister les volumes montés et afficher la place libre : on voit que la partition /dev/sdb1 n'est pas vraiment vide, 32 Mo occupés par reiserfs alors qu'elle est vide... ce sont des informations de gestion qui prennent toute cette place.

☞ Copiez les fichiers et dossiers de /home dans /mnt/home :

```
sudo cp -arT /home /mnt/home
```

Explication : tout à l'heure on va complètement remplacer `/home` par cette partition, il faut donc tout y recopier récursivement : c'est l'option -r qui fait ça. L'option -T indique de copier de dossier à dossier (sinon il faudrait mettre /home/*). L'option -a de cp permet de copier aussi tous les attributs des fichiers : dates, protection...

Rajoutez un fichier texte dans `/mnt/home` (c'est à dire sur /dev/sdb1) afin de pouvoir vérifier la réalité du montage quand il sera en place :

```
sudo touch /mnt/home/C_EST_L_AUTRE_VOLUME
```

Vous pouvez démonter la partition /mnt/home avec cette commande, à faire en superutilisateur, NB : il ne faut pas être dans le dossier /mnt/home pour la faire, alors avant, tapez `cd /` :

```
sudo umount /mnt/home
```

Pour la partition n°2 : montez-la dans un nouveau dossier appelé `/mnt/data`, laissez-la vide à part son dossier lost+found[1]. Créez un tout petit fichier texte (toto contenant « bonjour »). Démontez cette partition pour la suite (changez de répertoire puis faites umount). Vérifiez que le fichier toto n'est plus présent. Le point de montage reste, mais il est vide.

e) Montage automatique au démarrage

Éditez le fichier `/etc/fstab`, il faut y rajouter les lignes suivantes, attention mettez xfs ou reiserfs selon votre choix de format :

```
# montage de home sur la partition xfs
/dev/sdb1    /home       xfs         defaults    0     2
/dev/sdb2    /mnt/data   ext4        defaults    0     2
```

Avant de continuer, enregistrez votre Wiki et envoyez-le sur l'ENT. Il y a un risque de le perdre. En tous cas, fermez la fenêtre du navigateur, n'éditez plus le Wiki avant la fin du paragraphe.

Ensuite, soit vous redémarrez : `sudo reboot`, soit vous lancez la commande qui fait le montage des volumes au démarrage :

```
sudo mount -a
```

Vous devez constater que /home est maintenant associé au volume sdb1 : il y a le fichier /home/C_EST_L_AUTRE_VOLUME. Les commandes mount et df montrent également comment sont faits les montages :

```
mount
df -h
pydf
```

Vérifiez que vous avez à nouveau le fichier toto sur /mnt/data.

Pour finir, vous pouvez démonter `/dev/sdb1` ou `/home` afin de récupérer votre environnement normal. Si vous avez un message disant que le *device* est *busy*, c'est que vous êtes dedans ou un logiciel utilise ce dossier (le bureau OpenBox ou IceWeasel), faites d'abord `cd /` pour en sortir, mais il se peut que ça bloque quand même, alors faites ceci :

```
sudo umount -l /home
```

Pour vraiment finir, il faut commenter ce que vous avez rajouté dans `/etc/fstab`, sinon au prochain démarrage, ce sera le volume ReiserFS ou XFS qui sera monté sur /home. Mettez donc un # devant la ligne /dev/sdb1.

Maintenant, vous pouvez rouvrir et éditer votre Wiki (soyez méfiant si vous avez eu des messages d'erreur lors du démontage). Si jamais vous sentez que ça patauge, il vaut mieux redémarrer et récupérer votre Wiki de l'ENT.

f) Montage sans privilèges (user)

On va modifier les options de montage de /dev/sdb2 sur /mnt/data. D'abord, il faut la démonter :

1 Lost+found est un dossier qui permet de récupérer des fichiers perdus en cas de plantage du système de fichiers (c'est devenu très rare).

```
sudo umount -l /dev/sdb2
```

Essayez pour commencer de monter /dev/sdb2 en tant que simple utilisateur, pas de sudo devant la commande, en voici 3 à tester successivement :

```
mount /dev/sdb2 /mnt/data
mount /dev/sdb2
mount /mnt/data
```

Normalement, il doit chaque fois y avoir une erreur : « can't find in /etc/fstab » ou « only root can do that »

On va maintenant modifier la ligne concernée dans `/etc/fstab` qui permet à un utilisateur sans privilège de monter la 2e partition.

```
# montage de data sur la partition ext4 : possible pour un utilisateur
/dev/sdb2    /mnt/data   ext4        noauto,rw,user,exec     0     0
```

Refaites les tentatives de montage et cette fois-ci l'une des deux dernières, par le device, ou par le point de montage, doit réussir. C'est grâce au mot clé `user` dans les options de montage.

Démontez les volumes puis enlevez ou commentez les lignes dans `/etc/fstab` afin de ne plus rien monter au démarrage.

g) Montage d'une image de CD-ROM

Téléchargez `http://perso.univ-rennes1.fr/pierre.nerzic/SYS1A/data/tp10.iso` à l'aide de wget. C'est une « image iso » d'un CD-Rom, c'est à dire un contenu qui pourrait parfaitement être gravée sur un vrai CD à l'aide d'un logiciel de gravure. On veut afficher ce contenu, comme une partition.

Créer un dossier vide dans /mnt, par exemple `/mnt/iso`. Vérifiez par un coup de tree que ce dossier est véritablement vide au début.

i) Montage par mount -o loop

Monter l'image tp10.iso sur /mnt/iso par la commande suivante. L'option `ro` indique qu'on monte en lecture seule, l'option loop indique que c'est un périphérique spécial qui prend un fichier comme si c'était une sorte de lecteur CD.

```
sudo mount -o loop,ro tp10.iso /mnt/iso
```

À l'aide du navigateur de fichiers Thunar, allez voir ce qu'il y a dans /mnt/iso. Tout cela est en réalité dans le pseudo CR-Rom tp10.iso.

NB : on appelle un tel fichier iso : « image iso » parce qu'elle représente ce qu'il y a dans le CD, or justement cette « image » contient aussi des images... ne pas confondre les deux sortes d'images.

Démontez ce fichier (n'oubliez pas d'écrire la commande que vous employez dans le Wiki).

ii) Montage par /dev/loop1

Maintenant, associez l'image iso au fichier spécial /dev/loop1 puis montez-le par :

```
sudo losetup /dev/loop1 tp10.iso
sudo mount -o ro /dev/loop1 /mnt/iso
```

Ensuite vérifiez le contenu du dossier `/mnt/iso`. Vous retrouvez la même chose que précédemment. C'est une autre manière de monter le volume. Ça peut servir pour d'autres sortes de fichiers que des images iso, voir le § suivant.

Enfin, démontez ce volume et libérez le périphérique loop1 :

```
sudo umount /dev/loop1
sudo losetup -d /dev/loop1
```

h) Création d'une partition cryptée

Principe : on va créer un fichier contenant 256 blocs de 512 octets, puis associer ce fichier à /dev/loop1 avec le cryptage aes (il va demander le mot de passe), puis formater ce volume en ext4 puis le monter. A vous de vous renseigner sur les commandes employées (cours d'amphi et internet).

Voici d'abord la création du fichier qui contiendra le volume crypté :

```
dd if=/dev/zero ibs=512 count=256 of=crypt.img
```

Ensuite, il faut associer ce fichier à un périphérique spécial appelé /dev/mapper/PERSO :

```
sudo cryptsetup create -c aes -s 256 PERSO crypt.img
```

cryptsetup fait un peu le même travail que losetup : il transforme un fichier en un périphérique qui peut être formaté et monté.

Formater le volume crypté :

```
sudo mkfs.ext4 /dev/mapper/PERSO
```

Monter le volume crypté sur /mnt/perso :

```
sudo mkdir /mnt/perso
sudo mount /dev/mapper/PERSO /mnt/perso
```

Créez un fichier sur cette partition : sudo nano /mnt/perso/secret.txt. Puis démontez tout par :

```
sudo umount /dev/mapper/PERSO
sudo cryptsetup remove PERSO
```

Le test suivant consiste à remonter le cryp.img une première fois pour tester le mot de passe :

```
sudo cryptsetup create -c aes -s 256 PERSO crypt.img
sudo mount /dev/mapper/PERSO /mnt/perso
```

Vérifiez que le fichier secret est bien accessible et lisible, puis démontez tout.

À nouveau, remontez le volume mais donnez un mauvais mot de passe. Normalement, vous ne pourrez pas monter le volume, il signalera un « wrong fs type, bad option, bad superblock » qui signifie que le volume étant crypté, il est impossible de trouver un quelconque fichier dedans, c'est comme si le volume était non formaté. Par contre, à aucun moment on ne vous dit que le mot de passe n'est pas bon.

C'est ça l'astuce : cryptsetup ne sait pas du tout si le mot de passe est bon ou pas, c'est seulement au moment du montage que ça bloque ou pas. Du coup, un pirate passera beaucoup de temps à essayer tous les mots de passe possibles. Euh… hélas, l'entête d'un volume est parfaitement reconnaissable, alors il y a d'autres techniques que la force brute. Ce qu'il faudrait, c'est qu'un volume ext4 ne commence jamais par les mêmes octets, ou il faudrait rajouter des blocs aléatoires au début, avant le superbloc officiel du volume.

Faites les opérations qui défont ce qui a été mis en place dans cette manip.

2) Fin du TP

Envoyez votre Wiki sur l'ENT.

Vérifiez que `/etc/fstab` est vide ou toutes ses lignes commentées.

Fermez la machine virtuelle proprement puis quittez le logiciel VMware, et enfin supprimez le petit

disque virtuel supplémentaire qu'on a créé pour ce TP.