Monter le serveur de Telecom Etude à partir de zéro

« Comment monter un serveur de services en prenant l'exemple sur le serveur de Telecom Etude ? »

FENEUIL Thibauld

Mandat 2020

Guide du Pôle Informatique

TelecomEtude

Table des matières

Remerciements

Je tiens à remercier les membres du Pôle Informatique 2020 de Telecom Etude pour la vérification du contenu de ce guide, et plus particulièrement Tristan NEMOZ pour ses nombreuses relectures et remarques à la fois sur le fond et la forme.

Et je tiens à remercier plus globalement tous les Jetmen 2020 avec qui j'ai pris grand plaisir à gérer cette Junior-Entreprise pendant une année. J'en garderai d'agréables souvenirs. Merci à vous tous !

Première partie
Introduction

Voici un guide pour le Pôle Informatique de Telecom Etude. Il a pour vocation de former les nouveaux membres du pôle et de servir de guide (plus que détaillé) pour une réinstallation du serveur de la JE. Étant donné qu'il a une vocation de formation, il suppose que le lecteur est non-initié aux systèmes Linux. Les initiés pourront parcourir le guide plus rapidement. De plus, le guide va expliquer les différentes technologies utilisées par la Junior-Entreprise (LDAP, Samba, ...) qui sont le plus souvent méconnues même pour les initiés puisque cela touche au domaine de l'administration système.

Voici les notions Linux abordées dans ce guide :
- ☞ Installation d'un Debian
- ☞ Commandes de base de Linux
- ☞ Gestionnaire de paquets APT
- ☞ Sudo
- ☞ Cron

Voici les outils d'administration système abordés dans ce guide :
- ☞ Annuaire LDAP (*slapd*, *lam*)
- ☞ Configuration réseau
- ☞ Journalisation
- ☞ PAM (*Pluggable Authentication Modules*)
- ☞ RAID 1
- ☞ Serveur Mail (*postfix*, *dovecot*, SPF, DKIM)
- ☞ Serveur Samba (partage de fichiers)
- ☞ Serveur Web (*apache2*)
- ☞ GitLab (dans un Docker)

Cet ouvrage est une compilation de nombreux tutoriels trouvés sur le Net. Comme je ne voulais pas réinventer la roue, j'ai souvent fait des copier-coller ; mais pour des raisons de lisibilité, je n'ai pas mis les extraits recopiés entre guillemets. Cependant, tu pourras trouver les sources d'origine dans la section « Préliminaire » des différentes parties.

De plus, tout au long de ce guide, tu verras écrit « mot de passe usuel ». Cela désigne le mot de passe administrateur de Telecom Etude. A l'époque de mon mandat, il n'y en avait qu'un seul pour tout le serveur, ce qui n'est pas très sécurisé. Si tu suis ce guide et que tu en as la possibilité, je te conseille d'utiliser un mot de passe administrateur différent pour chaque application/service.

Bonne lecture !

Deuxième partie
Installation de Debian en RAID 1

1 Préliminaire

Dans cette section, va être décrite l'installation d'un Debian (guide réalisé avec un Debian 9) en mettant en place en même temps un RAID 1, technique qui permet d'améliorer la sécurité des données et la tolérance aux pannes en gardant plusieurs copies des données sur plusieurs disques séparés. Pour plus de renseignement sur l'utilité et le fonctionnement de disques en RAID, je te conseille de lire la page Wikipédia correspondante : https://fr.wikipedia.org/wiki/RAID_(informatique).

Si tu réinstalles le serveur, libre à toi d'utiliser un autre système d'exploitation GNU/Linux, mais dans ce cas, il faut que tu maîtrises ton *OS* car l'installation en RAID ne sera pas décrite dans ce guide. De plus, réfléchis bien aux futures générations du Pôle Informatique. Elles ne sauront peut-être pas utiliser ton *OS*, il te faudra écrire des tutoriels pour leur expliquer. Si tu as le choix, je te conseille de rester sur un système Debian.

Bien évidemment, pour mettre en place un RAID, il te faudra brancher **deux disques durs** (ou plus) sur le serveur.

Ressources :
— https://www.carnetdumaker.net/...-raid-1-pour-les-nuls
— https://wiki.archlinux.fr/chroot

Avoir deux disques durs sur VirtualBox

Si tu souhaites installer le serveur en RAID sur VirtualBox, voici comment créer une machine virtuelle avec deux disques durs : tout d'abord, crée la machine en choisissant l'option « sans disque » ; puis, va dans sa configuration, dans « Stockage ». Clique sur « Contrôleur : SATA » et ajoute deux fois des disques durs.

2 Installation de Debian en RAID 1

La première étape d'installation du système d'exploitation Debian est bien évidemment le téléchargement de l'installateur sur le site officiel du projet Debian : https://www.debian.org/distrib/netinst. Tu auras besoin d'accéder à Internet pendant l'installation. Si ce n'est pas possible, tu peux récupérer l'image iso sur https://www.debian.org/CD/http-ftp/.

Choisis la version qui convient le mieux pour ta machine : *amd64* pour les processeurs 64 bits ou *i386* pour les processeurs 32 bits.

A présent, plusieurs options s'offrent à toi : tu peux soit graver l'installateur sur un CD/DVD, soit mettre l'installateur sur une clé USB et rendre celle-ci bootable. De nos jours, l'utilisation d'une clé USB bootable est nettement plus facile. Pour

préparer la clé USB, utilise par exemple l'utilitaire Rufus (https://rufus.akeo.ie/). Attention, toutes les données qui étaient sur ta clé seront supprimées.

Démarre le serveur sur ton CD ou ta clé en **mode UEFI**. Tu peux utiliser l'« installateur graphique ». Rappel : tu auras besoin d'avoir accès à Internet pendant l'installation si tu as choisi de passer par https://www.debian.org/distrib/netinst.

A présent, suis ce tableau :

Langue de l'installation	Français
Situation géographique	France
Disposition du clavier	Français
Type interface réseau	Giga Ethernet
Nom de la machine	je
Domaine	*rien*
Mot de passe Root	*Mot de passe usuel*
(Nouvel utilisateur) Nom complet	JE
(Nouvel utilisateur) identifiant	je
(Nouvel utilisateur)	*Mot de passe usuel*
Méthode de partitionnement	Manuel

Active le partitionnement des disques (il suffit de double-cliquer dessus). Sur un disque, on a au maximum 4 partitions primaires. Voici les partitions qu'on va créer :

Nom	**Taille**	**Type**	**Emplac.**	**Utiliser comme**	**Amorçage**
EFI	512 MB	Primaire	Début	Partition système EFI	Présent
`/tmp`	10 GB	Primaire	Début	Volume pour RAID[1]	Absent
swap	$1.5 \times$ RAM[2]	Primaire	Début	Volume pour RAID[1]	Absent
`/`	Le reste...	Primaire	Début	Volume pour RAID[1]	Absent

Sur le premier disque, crée les partitions **dans le même ordre que dans le tableau** en suivant les options indiquées. Pour la dernière partition, tu peux mettre « max » comme taille.

Et à présent, fais **exactement** la même chose avec le second disque ; il faut que les tailles soient les mêmes.

Maintenant, clique sur « Configurer le RAID avec gestion logicielle ». Dis « Oui » pour appliquer les changements et suis la liste d'instructions suivante :

1. « Créer un périphérique multidisque »
2. « RAID 1 »
3. Périphériques actifs : 2 *(nb de disques utilisés dans le montage du RAID 1)*
4. Périphériques en réserve : 0
5. Sélectionne deux partitions (une de chaque disque) qui vont être synchronisées. Commence par `/dev/sda2` et `/dev/sdb2`.

1. Volume physique pour RAID
2. Taille du swap : $1.5 \times$ la taille de la RAM

Et réitère la liste d'instructions avec successivement :
— `/dev/sda3` et `/dev/sdb3`
— `/dev/sda4` et `/dev/sdb4`
Puis clique sur « Terminer ».

Normalement, tu vas voir trois périphériques RAID1. Il s'agit des partitions virtuelles qui réunissent les partitions RAID physiques des disques.

Pour le périphérique n° 0 RAID 1 :
— Modifie la partition (double-clique dessus)
— Utiliser comme « Système de fichiers jounalisé ext4 »
— Point de montage : `/tmp`
— Option de montage : *nosuid*, *nodev*, *noexec*, *noatime*
— étiquette : « TEMP »
Pour le périphérique n° 1 RAID 1 :
— Modifie la partition (double-clique dessus)
— Utiliser comme « Espace d'échange « swap » »
Pour le périphérique n° 2 RAID 1 :
— Modifie la partition (double-clique dessus)
— Utiliser comme « Système de fichiers journalisé ext4 »
— Point de montage : `/`
— Option de montage : *noatime*
— étiquette : « SYSTEM »
Puis clique sur « Terminer le partionnement et appliquer les changements » et suis ce tableau :

Analyser un autre DVD	Non
Utiliser un miroir sur le réseau	Oui
Pays du miroir de l'archive Debian	France
Miroir de l'archive	`ftp.fr.debian.org`
Mandataire HTTP	*rien*
Participer à l'étude statistique sur l'utilisation des paquets	Non
Logiciel à installer	Serveur SSH Utilitaires usuels du système[1]
Installer le programme de démarrage GRUB sur le secteur d'amorçage	Oui, puis sélectionne un des disques physiques.

Et c'est terminé ! Ton système d'exploitation est installé en RAID 1 !

3 Gestion de l'amorçage UEFI

EFIBootmgr est un utilitaire utilisable en ligne de commande permettant de gérer le chargeur de démarrage EFI. Il permet de :

— Modifier l'ordre de démarrage des systèmes d'exploitation disponibles ;
— Créer ou supprimer des entrées ;
— Modifier les options d'exécution du prochain démarrage ;
— Etc...

1. Ne pas cliquer sur le serveur Web, il est plus propre de l'installer après.

3.1 Utilisation de EFIBootmgr

3.1.1 Connaître l'ordre d'amorçage

Avant de commencer toute modification sur l'UEFI de ton ordinateur, il est nécessaire d'en connaître un peu plus sur les systèmes disponibles en mode UEFI et leur ordre de démarrage. Pour cela, il suffit d'utiliser l'option `-v`.

Dans un terminal, saisis la commande suivante :

```
efibootmgr -v
```

Le terminal te renverra une réponse, comme par exemple :

```
BootCurrent: 0001
Timeout: 2 seconds
BootOrder: 0001,3001,0002,2001,2002,2004
Boot0000* Disque dur USB (UEFI) - Generic Flash Disk    ACPI(
    a0341d0,0)PCI(14,0)USB(1,0)HD(1,3e,3b5b92,000bb565)RC
Boot0001* ubuntu        HD(1,145800,82000,393abc6a-5b46-4392-
    a2fa-aebd5ee7d640)File(\EFI\ubuntu\shimx64.efi)
Boot0002* Windows Boot Manager  HD(1,145800,82000,393abc6a-5b46
    -4392-a2fa-aebd5ee7d640)File(\EFI\Microsoft\Boot\bootmgfw.
    efi)WINDOWS.......
Boot2001* EFI USB Device        RC
Boot2002* EFI DVD/CDROM RC
Boot3001* Internal Hard Disk or Solid State Disk        RC
```

Explications

- La 1ère ligne `BootCurrent` indique le système amorcé. Dans notre cas, `0001` correspondant à Ubuntu.
- La 2ème ligne `Timeout` indique le temps avant amorçage. 2 secondes dans notre cas.
- La 3ème ligne `BootOrder` indique l'ordre dans lequel sont amorcés les systèmes UEFI.
- Les autres lignes listent toutes les possibilités d'amorçage avec leur nombre hexadécimal correspondant chacun à une entrée dans l'UEFI.

Après chaque manipulation de l'UEFI et ce avant de redémarrer, il est très intéressant de vérifier tes modifications avec la commande citée précédemment.

3.1.2 Remettre le système d'exploitation du serveur en priorité

Ton ordinateur ne démarre plus sous le bon système d'exploitation, et pourtant il a été installé correctement en mode UEFI ? Voici une solution :

1. Éteindre complètement le serveur ;
2. Démarrer ton ordinateur en faisant apparaître le menu de démarrage du BIOS ;
3. Choisir le bon système d'exploitation dans la liste présentée et valider.

Une fois l'*OS* démarré, il est nécessaire de réécrire son entrée dans l'UEFI et ce, en priorité. Le plus simple est de ré-installer GRUB, (le chargeur de démarrage d'Ubuntu) avec la commande :

```
grub-install
```

Cette commande réinstallera le GRUB dans la partition UEFI et remettra ainsi l'accès au bon système d'exploitation en priorité au démarrage. Une fois lancée cette commande, le terminal te renverra

```
Installing for x86_64-efi platform.
Installation terminée, sans erreur.
```

La réinstallation du GRUB est terminée, au prochain redémarrage ton ordinateur s'amorcera correctement.

Et si le serveur ne redémarre plus, comment faire (à part pleurer, *of course*) ?

Lance le système Unix que tu peux, en connectant les disques durs du serveur. Si tu n'en as aucun de disponible, installe **ArchLinux** sur une clé USB et rends-la bootable.

Identifie la partition RAID qui était montée sur la racine du serveur. Dans mon exemple, c'est `/dev/md126`.

```
df -h
```

Nous allons monter le serveur sur `/mnt/`.

```
mount /dev/md126 /mnt
mount -t proc /proc /mnt/proc/
mount --rbind /sys /mnt/sys
mount --rbind /dev /mnt/dev
mount --rbind /run /mnt/run
```

Et là, tu peux accéder au serveur

```
chroot /mnt/ /bin/bash
```

et tu peux lancer la réinstallation de GRUB

```
grub-install
```

3.1.3 Modifier l'ordre d'amorçage

Si tu souhaites ou dois modifier l'ordre d'amorçage des entrées UEFI, il suffit, dans la commande suivante, de classer les valeurs hexadécimales (de chaque entrée UEFI disponible) de la première à la dernière en les séparant par une virgule. Par exemple :

```
efibootmgr -o 0001,3001,0002,2002,2001
```

3.1.4 Supprimer une entrée

Pour supprimer une entrée de l'UEFI, il suffit d'utiliser l'option `-B` suivie de la valeur hexadécimale de l'entrée à supprimer.

Exemple 1 :

```
efibootmgr -B 2002
```

Dans cet exemple de commande, la valeur hexadécimale `2002` supprime l'entrée du périphérique DVD/CDROM de l'exemple précédent.

Exemple 2 :

Supprimer l'entrée « Boot000E » de valeur hexadécimale E :

```
efibootmgr -b E -B
```

3.1.5 Changer le temps avant amorçage

Pour changer le temps avant que l'UEFI ne démarre, il suffit d'utiliser l'option `-t` suivie du nombre de secondes. Exemple de commande pour passer à 5 secondes :

```
efibootmgr -t 5
```

Pour ceux qui souhaitent un amorçage rapide de l'UEFI, on peut supprimer ce temps d'attente avec la commande :

```
efibootmgr -T
```

3.2 Configuration du chargeur de démarrage EFI

L'idée est de créer une entrée dans le chargeur avec l'identifiant « prod-je » au lieu de « debian ». Pour cela, on va modifier la configuration de GRUB. Modifie l'option `GRUB_DISTRIBUTOR` du fichier `/etc/default/grub`.

```
GRUB_DISTRIBUTOR="prod-je"
```

Réinstalle GRUB...

```
grub-install
```

Cela a créé une nouvelle entrée dans le chargeur de démarrage EFI. Il faut que tu supprimes la précédente entrée. Utilise la commande `efibootmgr-v` pour identifier l'entrée à supprimer et supprime-la (remplace `001` pour le bon identifiant hexadécimal) :

```
efibootmrg -b 001 -B
```

A titre indicatif, voici comment réinstaller GRUB en rajoutant une entrée sans *hard-coder* la configuration.

```
grub-install --target=x86_64-efi --efi-directory=/boot/efi --
    bootloader-id=prod-je
```

4 Gérer une grappe de disques en RAID 1

4.1 Voir l'état d'une grappe de disques

Pour voir l'état des grappes de disques, tu peux utiliser la commande suivante :

```
cat /proc/mdstat
```

Avec cette commande, tu as un aperçu de l'état de tous les disques et de toutes les grappes de disques.

```
Personalities : [raid1] [linear] [multipath] [raid0] [raid6] [
    raid5] [raid4] [raid10]
md3 : active raid1 sdb4[0] sda4[1]
      1931373568 blocks super 1.2 [2/2] [UU]
      bitmap: 5/15 pages [20KB], 65536KB chunk

md2 : active raid1 sda3[2] sdb3[0]
      9757696 blocks super 1.2 [2/2] [UU]

unused devices: <none>
```

Le plus important est le UU en fin de ligne. Un U majuscule signifie que le disque correspondant est sain, un *underscore* signifie que le disque est défaillant. En cas de problème, un (F) doit s'afficher à côté du nom du disque défaillant sur la même ligne.

Pour avoir un aperçu en temps réel de l'état des disques, tu peux utiliser la commande suivante :

```
watch cat /proc/mdstat
```

Celle-ci permet d'actualiser le résultat de la commande précédente toutes les deux secondes par défaut.

Pour avoir plus de détails sur un disque en particulier, tu peux utiliser la commande suivante (pense à remplacer `N` par le numéro de disque) :

```
mdadm --detail /dev/mdN
```

4.2 Éjecter un disque d'une grappe de disques

Si un disque est mourant, ne redémarre surtout pas le serveur. C'est la pire idée qui soit. Avant d'éjecter un disque défaillant, il faut le retirer de la (les)

grappe(s) de disques dont il fait partie. *Si tu ne le fais pas, de grands malheurs s'abattront sur toi.*

Pour retirer un disque `/dev/sdX`, il faut d'abord passer la grappe de disques en mode dégradé en marquant le disque comme "défaillant" en exécutant cette commande (après avoir remplacé successivement `N` par le numéro des partitions) :

```
mdadm --fail /dev/md{N-1} /dev/sdX{N}
```

Une fois le disque marqué comme "défaillant", on peut retirer le disque de la grappe sans souci avec la commande suivante :

```
mdadm --remove /dev/md{N-1} /dev/sdX{N}
```

À partir de là, tu peux éjecter physiquement le disque du serveur, le remplacer et mettre un nouveau disque dans le serveur à la place de l'ancien.

4.3 (Re)Ajouter un disque à une grappe de disques

Il faut d'abord cloner les partitions d'un disque existant de la grappe vers le disque neuf (attention, s'il n'était pas vierge, les données seront perdues).

Exemple de clonage des partitions du disque `sda` vers `sdb` (à exécuter avec les droits de super-utilisateur) :

```
sfdisk -d /dev/sda | sfdisk /dev/sdb
```

N.B. Le nouveau disque doit être de même contenance et idéalement de même modèle / performance que les autres disques de la grappe.

Une fois le disque prêt à être intégré dans la grappe, il suffit d'exécuter la commande suivante (remplacer `N` par le numéro de disque / partition adéquat) :

```
mdadm --add /dev/mdN /dev/sdbN
```

Le disque va alors entrer en mode *recovery* et commencer un très long processus de synchronisation avec les disques existants. Inutile d'aller te faire un café, cela peut durer plusieurs dizaines de jours. Le serveur reste cependant parfaitement fonctionnel durant la phase de synchronisation des disques.

4.3.1 Accélérer la synchronisation des disques

Une synchronisation de disques peut prendre plusieurs dizaines de jours. Cela est dû à la façon dont `mdadm` gère les ressources système par défaut.

`mdadm` est configuré par défaut pour utiliser le moins de ressources système possibles, dans le but de ne pas impacter les applications utilisateurs lors d'une synchronisation automatique en tâche de fond.

Il est cependant possible de lever temporairement la limitation en ressources de `mdadm` avec la commande suivante :

```
echo 125000 > /proc/sys/dev/raid/speed_limit_min
echo 400000 > /proc/sys/dev/raid/speed_limit_max
```

Cette commande autorise mdadm à utiliser un minimum de 125Mo/s de bande passante pour les disques durs, avec un maximum de 400Mo/s (pratique pour les SSD).

Grâce à cette autorisation exceptionnelle à pomper chaque Mo/s possible en provenance des disques, `mdadm` peut faire une synchronisation complète d'une grappe de disques de 1To en 45 minutes (pour peu que les disques soient récents).

Pour revenir aux valeurs par défaut (1 Mo/s) :

```
echo 1000 > /proc/sys/dev/raid/speed_limit_min
```

4.3.2 Vérification à faire après chaque remplacement de disque dur

Pour qu'un disque soit *bootable* (démarrable), il faut qu'un petit bout de code (le *bootloader*) soit inscrit dans le premier secteur du disque.

Et oui, quand on a deux disques (ou plus) dans une configuration RAID 1, il ne faut pas oublier d'installer le code de démarrage sur les disques miroirs. Sinon, une fois que le disque d'origine ayant servi à l'installation du système est remplacé, plus rien ne démarre. Et paf ! Le serveur te sort un « Aucun disque de démarrage ».

Pour choisir les disques sur lesquels le code de démarrage doit être installé, il faut exécuter la commande suivante :

```
dpkg-reconfigure -plow grub-pc
```

Laisse les deux premiers champs intacts, appuie simplement sur entrée pour passer au champ / formulaire suivant.

Une fois sur le formulaire de sélection des disques, utilise la touche « espace » pour sélectionner les disques adéquats (ne pas sélectionner les disques `mdN`, qui sont les disques virtuels du RAID). Une fois les disques sélectionnés, appuie sur la touche entrée pour lancer l'installation.

Cette opération doit être effectuée **après chaque remplacement de disque**. Sinon, tu auras une vilaine surprise au pire moment possible plus tard.

Troisième partie
Kit de survie du guide

Ressources :
— https://openclassrooms.com/.../37813-la-console-ca-se-mange
— https://doc.ubuntu-fr.org/arborescence
— https://doc.ubuntu-fr.org/pipe
— https://doc.lagout.org/.../kit-de-survie-linux-8386-ksj4ri.pdf
— https://doc.ubuntu-fr.org/permissions

1 La console

Pour ce tutoriel, je pars du principe que tu ne connais rien à Linux. Voici donc une petite présentation de la console dans un système d'exploitation Unix par Mathieu Nebra.

> Depuis l'invention de l'interface graphique, on pourrait se demander **pourquoi on n'a pas supprimé la console** (sous-entendu : « Elle ne sert plus à rien »). C'est là que beaucoup se trompent complètement : on met un peu de temps à s'y faire, mais quand on sait s'en servir, on va beaucoup plus vite avec la console qu'avec l'interface graphique. C'est même pire en fait : vous vous rendrez compte à un moment qu'il y a des choses que seule la console peut faire et qu'il serait de toute façon vraiment inutile de recourir à une interface graphique pour les effectuer.
>
> Un exemple ? En mode graphique, allez dans un répertoire qui contient beaucoup de fichiers en tout genre : des fichiers texte, des images, des vidéos... Vous voudriez savoir combien il y a d'images JPEG dans ce dossier : pas facile hein ? En console, en assemblant quelques commandes, on peut obtenir ce résultat sans problème !

```
ls -l | grep jpg | wc -l
510
```

> La première ligne est la commande que j'ai tapée, la seconde le résultat. Il y avait donc 510 images JPEG dans le dossier, et on a obtenu le résultat en moins d'une seconde !
>
> On peut même faire encore plus fort et enregistrer directement ce nombre dans un fichier texte :

```
ls -l | grep jpg | wc -l > nb_jpg.txt
```

> ... et on peut aussi envoyer le fichier nb_jpg.txt sur Internet par FTP ou à un ami par e-mail, le tout en une ligne ! La

> console n'est donc pas morte et n'a pas du tout prévu de l'être !
>
> La plupart des commandes de la console de Linux sont des « copies » d'Unix, ancêtre parmi les ancêtres. N'allez pas croire que les programmes d'Unix ont été copiés ou « piratés » par Linux ; c'est juste que leur mode d'emploi est le même. Les programmes ont été réécrits par un groupement de programmeurs issus de ce qu'on appelle le projet GNU. Ce projet a fusionné au bout de quelque temps avec le coeur du système d'exploitation Linux pour donner au final GNU/Linux, qu'on écrit en pratique juste « Linux » car c'est plus court.
>
> L'avantage ? Les commandes n'ont pas bougé et ne bougent pas depuis l'époque d'Unix (soit depuis les années 60). Ce sont les mêmes. Quelqu'un qui utilisait Unix dans les années 60 est capable de se débrouiller avec un Linux d'aujourd'hui. Et il y a fort à parier que ce sera pareil pour les nombreuses années à venir. Vous avez donc juste à apprendre à vous en servir une fois. O.K., il y aura du boulot, mais après ce sera quelque chose qui pourra vous servir toute votre vie !
>
> *— Mathieu Nebra sur OpenClassrooms*

J'espère que tout cela t'a convaincu de l'utilité de la console. Il faut donc à présent que tu apprennes à l'utiliser. Tout d'abord, connecte-toi sur un compte. A ce stade du guide, tu peux te connecter au compte de `je` avec le *mot de passe usuel*. Plus tard dans le guide, tu pourras te connecter à ton compte personnel.

2 L'arborescence des fichiers

Avec la console, tu navigues à travers une arborescence de fichiers. La racine symbolise une partition définissant la base du stockage des fichiers. Puis cette base se sépare (comme les branches d'un arbre) logiquement en répertoires (dossiers), eux-mêmes séparés en sous-répertoires et sous-sous-répertoires, etc. dans lesquels sont enregistrés les fichiers (symboliquement, les feuilles de l'arbre).

La racine de l'arborescence des fichiers est notée `/`. Le répertoire `tfeneuil`, qui est dans le répertoire `home`, est noté `/home/tfeneuil`. Et ainsi de suite...

Voici la liste des répertoires les plus importants d'une arborescence standard d'une distribution Linux :

Répertoire	Signification	Contenu
/		Racine du système, hiérarchie primaire

Continue sur la page suivante...

/bin	**bin**ary utilities	Exécutables des commandes essentielles disponibles pour tous les utilisateurs (ex : `cd`, `cat`, `ls`...)
/boot	**boot**strap	Fichiers statiques du chargeur d'amorçage (noyaux, images ramdisk, fichiers de configuration du chargeur d'amorçage...)
/dev	**dev**ice	Fichiers spéciaux des périphériques
/etc	**e**diting **t**ext **c**onfig	Fichiers de configuration au format textuel de plusieurs programmes et services du système
/home	**home** directory	Répertoires personnels des utilisateurs
/lib	**lib**rairies	Bibliothèques partagées essentielles et modules du noyau
/media		Contient les points de montages pour les médias amovibles
/mnt	**m**ou**nt**	Point de montage pour monter temporairement un système de fichiers
/opt	**opt**ional	Emplacement pour des applications installées hors gestionnaire de paquets (logiciels optionnels)
/proc	**proc**esses	Répertoire virtuel pour les informations système (états du noyau et des processus système)
/root	**root**	Répertoire personnel du *super-utilisateur*
/run	**run**time system	Informations relatives au système depuis son dernier démarrage (ex : utilisateurs actifs, services en cours d'exécution, etc.)
/sbin	**s**uper **bin**aries	Exécutables système essentiels
/srv	**s**er**v**ices	Données pour les services du système
/tmp	t**emp**orary	Fichiers temporaires des applications
/usr	**U**nix **s**ystem **r**esources	Hiérarchie secondaire, pour des données en lecture seule par les utilisateurs. Ce répertoire contient la vaste majorité des applications usuelles des utilisateurs et leurs fichiers
/usr/bin		Exécutables des programmes additionnels disponibles pour tous les utilisateurs (ex : gestionnaire de fichiers, lecteur de musique, navigateur Web...)
/usr/lib		Bibliothèques partagées par les applications additionnelles de `/usr/bin` et `/usr/sbin`
/usr/local		Hiérarchie tertiaire. Emplacement où

Continue sur la page suivante...

		les utilisateurs doivent installer les applications qu'ils compilent
/usr/share		Fichiers, non reliés à l'architecture, partagés par les applications de `/usr/bin` et `/usr/sbin` (ex : thèmes, documentation...)
/var	**var**iable	Données variables et diverses

Et maintenant, il faut se jeter dans le grand bain ! Je vais t'apprendre tes premières commandes.

3 Les premières commandes

N'hésite surtout pas à les tester !

3.1 Se déplacer dans l'arborescence de fichiers

Avant tout déplacement dans l'arborescence, il te faut savoir où tu te trouves. Pour cela, tape `pwd`. Il s'agit d'une commande pour afficher ta position dans les différents répertoires.

```
je@je:~$ pwd
/home/je
```

Pour te déplacer, utilise la commande `cd` suivie du nom du répertoire où tu veux aller.

```
je@je:~$ cd / # Je vais dans le répertoire racine
je@je:/$ pwd
/

je@je:/$ cd home # Je vais dans le sous-répertoire "home"
je@je:/home$ pwd
/home

je@je:/$ cd /etc/apt # Je vais dans le répertoire "/etc/apt"
je@je:/etc/apt$
```

Certains dossiers ont des noms spéciaux : le répertoire courant (celui sur lequel tu es) est désigné par `.`, le répertoire parent par `..` et le répertoire personnel du compte par `~`.

```
je@je:/etc/apt$ cd . # Je fais du sur-place
je@je:/etc/apt$ cd .. # Je vais dans le répertoire parent
je@je:/etc$ cd ~ # Je vais dans mon répertoire personnel
je@je:~$ pwd
/home/je
```

Et, dernière commande, si tu veux connaître la liste des fichiers dans un répertoire, utilise `ls`.

```
je@je:~$ cd /
je@je:/$ ls # Affiche la liste des dossiers et fichiers du ré
    pertoire courant
bin   boot_bak  etc   lib    lost+found  mnt  proc  run   srv
    tmp  var
boot  dev       home  lib64  media       opt  root  sbin  sys
    usr
je@je:/$ ls -l # Version détaillée
drwxr-xr-x   2 root root  4096 janv. 16 06:08 bin
drwxr-xr-x   5 root root  4096 janv.  6 15:13 boot
drwxr-xr-x   6 root root  4096 janv.  6 03:10 boot_bak
drwxr-xr-x  19 root root  3280 janv. 16 01:07 dev
drwxr-xr-x 103 root root  4096 janv. 28 06:51 etc
drwxr-xr-x   9 root root  4096 janv. 12 20:29 home
drwxr-xr-x  15 root root  4096 sept. 10 15:34 lib
drwxr-xr-x   2 root root  4096 sept.  7 19:54 lib64
drwx------   2 root root 16384 sept.  7 19:52 lost+found
drwxr-xr-x   3 root root  4096 sept.  7 20:05 media
drwxr-xr-x   2 root root  4096 sept.  7 19:53 mnt
drwxr-xr-x   5 root root  4096 nov.  17 18:52 opt
dr-xr-xr-x 271 root root     0 janv. 16 01:06 proc
drwx------   6 root root  4096 janv. 28 17:49 root
drwxr-xr-x  24 root root   800 janv. 28 19:39 run
drwxr-xr-x   2 root root  4096 janv. 16 06:08 sbin
drwxr-xr-x   3 root root  4096 nov.  17 19:20 srv
dr-xr-xr-x  13 root root     0 janv. 16 01:06 sys
drwxrwxrwt   9 root root  4096 janv. 28 19:58 tmp
drwxr-xr-x  10 root root  4096 sept.  7 19:53 usr
drwxr-xr-x  12 root root  4096 sept.  7 21:37 var
je@je:/$ ls -l /etc/apt
apt.conf.d        preferences.d  sources.list.d  trusted.gpg.d
listchanges.conf  sources.list   trusted.gpg
```

3.2 Les permissions des fichiers et des répertoires

Les droits des fichiers d'un répertoire peuvent être affichés par la commande

```
ls -l
```

Les droits d'accès apparaissent alors comme une liste de 10 symboles :

```
drwxr-xr-x
```

Le premier symbole peut être « - », « d », soit « l », entre autres. Il indique la nature du fichier :

- : fichier classique

d : répertoire

l : lien symbolique

Suivent ensuite 3 groupes de 3 symboles chacun, indiquant si le fichier (ou répertoire) est autorisé en lecture, écriture ou exécution. Les 3 groupes correspondent, dans cet ordre, aux droits du propriétaire, du groupe puis du reste des utilisateurs. Dans le paragraphe introductif, tu auras remarqué des lettres en gras dans les termes anglais. Ce sont ces lettres qui sont utilisées pour symboliser lesdites permissions. Si la permission n'est pas accordée, la lettre en question est remplacée par « - ». Si l'on reprend les lettres données pour lecture/écriture/exécution (**r**ead/**w**rite/e**x**ecute), nous obtenons : **rwx**.

Reprenons l'exemple théorique précédent :

```
drwxr-xr-x
```

Il se traduit de la manière suivante :

d : c'est un répertoire.

rwx pour le 1er groupe de 3 symboles : son propriétaire peut lire, écrire et exécuter.

r-x pour le 2nd groupe de 3 symboles : le groupe peut uniquement lire et exécuter le fichier, sans pouvoir le modifier.

r-x pour le 3ème groupe de 3 symboles : le reste des utilisateurs peut uniquement lire et exécuter le fichier, sans pouvoir le modifier.

3.3 Informations au début de la commande

A ce stade, tu peux voir dans les informations présentes avant le curseur dans la console

```
tfeneuil@je:/etc/apt$
```

Il s'agit du nom d'utilisateur « tfeneuil » suivi du nom de la machine `je`, puis suivi de la position dans l'arborescence `/etc/apt`. C'est assez courant de trouver ces informations sur un système Linux, mais ce peut être d'autres informations (c'est configurable), donc ne t'étonne pas de voir autre chose sur d'autres Linux (je pense en particulier à la console de ton compte de Télécom ParisTech).

3.4 Lire et éditer des fichiers

Pour lire un fichier, tu peux utiliser la commande `less`. Elle te permet de parcourir un fichier.

```
je@je:/$ cd /etc/apt
je@je:/etc/apt$ less smbd.conf # Appuie sur "Q" pour quitter la
    lecture
je@je:/etc/apt$ less ../host.conf
```

Pour lire un fichier **court**, tu peux utiliser la commande `cat`, qui imprime le contenu du fichier sur la console.

```
je@je:/etc/apt$ cd ..
je@je:/etc$ cat host.conf
multi on
```

Et pour éditer ? Je te propose d'utiliser provisoirement `nano`.

```
je@je:/etc/apt$ cd ~
je@je:~$ nano test.txt # Crée le fichier s'il n'existe pas et l'
    édite
```

Pourquoi dis-je « provisoirement » ? C'est parce que `nano` est un utilitaire peu complet (idéal pour les débutants sur Linux). Il te faudra passer rapidement à `vim` ! Mais, à ce moment de ton apprentissage, `vim` n'est pas encore installé sur le serveur.

Pour supprimer un fichier, utilise `rm`. Attention, cette commande est dangereuse car elle supprime **définitivement** les données.

```
je@je:~$ ls
test.txt
je@je:~$ rm test.txt
je@je:~$ ls
```

3.5 La commande la plus importante !

Il existe un immense nombre de commandes, mais celle que tu dois retenir plus que tout, c'est `man` ! C'est le manuel de toutes les commandes. En gros, si tu ne sais pas à quoi sert ou comment utiliser une commande, tu peux appeler `man` au secours !

```
je@je:~$ man ls # Pour afficher l'aide de "ls"
NOM
       ls - Afficher le contenu de répertoires

SYNOPSIS
       ls [OPTION] ... [FICHIER] ...

DESCRIPTION
       Afficher  les informations des FICHIERs (du répertoire
           courant par défaut). Les entrées sont triées alphabé
           tiquement si aucune des options -cftuvSUX ou --sort
           n'est indiquée.

       Les paramètres obligatoires pour les options de forme
           longue le sont aussi pour les options de forme
           courte.

       -a, --all
              inclure les entrées débutant par Â« . Â»

       -A, --almost-all
```

```
              omettre les fichiers Â« . Â» et Â« .. Â»

       --author
              avec -l, afficher l'auteur de chaque fichier

       (...)
```

```
je@je:~$ man man # Et tu peux même demander l'aide de l'aide !
NOM
       man - Interface de consultation des manuels de référence
           en ligne

SYNOPSIS
       man  [-C file] [-d] [-D] [--warnings[=warnings]] [-R
           encoding] [-L locale] [-m system[,...]] [-M path] [-
           S list] [-e extension]
       [-i|-I] [--regex|--wildcard] [--names-only] [-a] [-u] [--
           no-subpages] [-P pager] [-r prompt] [-7] [-E
           encoding] [--no-hyphena-
       tion] [--no-justification] [-p string] [-t] [-T[device]]
           [-H[browser]] [-X[dpi]] [-Z] [[section] page[.
           section] ...] ...
       man -k [options d'apropos] expression_rationnelle ...
       man -K [-w|-W] [-S liste] [-i|-I] [--regex] [section]
           term ...
       man -f [options de whatis] page ...
       man -l [-C fichier] [-d] [-D] [--warnings[=avertissements
           ]] [-R encodage] [-L locale] [-P afficheur] [-r
           invite] [-7] [-E enco-
       dage] [-p chaîne] [-t] [-T[périphérique]] [-H[navigateur
           ]] [-X[ppp]] [-Z] fichier ...
       man -w|-W [-C fichier] [-d] [-D] page ...
       man -c [-C fichier] [-d] [-D] page ...
       man [-?V]

DESCRIPTION
       man est le programme de visualisation des pages de manuel
           . Chacun des arguments page, indiqué dans la ligne
           de commande de man,
       porte, en principe, le nom d'un programme, d'un
           utilitaire ou d'une fonction. La page de manuel
           correspondant à chaque argument
       est alors trouvée et affichée. Si une section est précisé
           e alors man limite la  recherche  à  cette  section.
             Par  défaut,  il
       recherche  dans  toutes  les sections disponibles en
           suivant un ordre prédéfini (Â« 1 n l 8 3 2 3posix 3
           pm 3perl 3am 5 4 9 6 7 Â»
       par défaut, à moins d'être écrasée par la directive
           SECTION dans /etc/manpath.config). Il n'affiche que
            la  première  page  de
```

```
       manuel trouvée, même si d'autres pages de manuel existent
            dans d'autres sections.

       (...)
```

3.6 Faire des recherches

Pour faire des recherches sur les dossiers et les fichiers eux-mêmes, utilise `find`.

— Tu cherches un dossier dont le nom est `archive` dans `/etc` ?

```
je@je:~$ find /etc -type d -name archive
/etc/letsencrypt/archive
```

— Tu cherches un fichier dont le nom est `argentina` et dont le propriétaire est « tfeneuil » ?

```
je@je:~$ find -type f -name argentina -user tfeneuil
./test/argentina
```

— Tu cherches un fichier dont le nom finit par `.conf` ?

```
je@je:/etc$ find -type f -name "*.conf"
./host.conf
./modprobe.d/dkms.conf
./modprobe.d/mdadm.conf
./udev/udev.conf
./hdparm.conf
./libnss-ldap.conf
./pam_ldap.conf
  (...)
```

Les possibilités de recherche sont quasiment infinies ! Je te conseille de regarder dans le manuel pour avoir un aperçu des options.

Si tu cherches du texte dans un fichier, c'est `grep` qu'il faut utiliser.

```
# Recherche "grubdir" dans le fichier "00_header"
je@je:/etc/grub$ grep "grubdir" 00_header
grubdir="`echo "/boot/grub" | sed 's,//*,/,g'`"
abstractions="$(grub-probe --target=abstraction "${grubdir}")"
FS="$(grub-probe --target=fs "${grubdir}")"
```

```
# Recherche "ldap" dans tous les fichiers du répertoire courant
root@je:/etc$ grep -r "ldap" .
./postfix/postfix-files.d/ldap.files:$manpage_directory/man5/
    ldap_table.5.gz:f:root:-:644
./postfix/dynamicmaps.cf:ldap    postfix-ldap.so dict_ldap_open
./gshadow-:openldap:!::
./ldap-account-manager/apache.conf:Alias /lam /usr/share/ldap-
    account-manager
```

```
./ldap-account-manager/apache.conf:<Directory /usr/share/ldap-
    account-manager>
    (...)
```

3.7 Chaînage de commandes

Un outil pour mélanger des commandes, c'est le « *pipe* » ou « tuyau », symbolisé par la barre verticale `|`. Cela permet de passer le résultat d'une commande à une autre commande. Par exemple, pour connaître la liste des fichiers et dossiers de `/etc/` avec « la » dans le nom, on peut faire

```
je@je:/etc$ ls | grep "la"
```

Le résultat de la commande `ls` (qui est censée afficher la liste des répertoires et fichiers) est envoyé dans `grep "la"` (qui va sélectionner les fichiers avec « la »). Tu pourrais me dire qu'on peut avoir directement le résultat avec `find`, et tu aurais raison. D'ailleurs, je te laisse trouver, en guise d'exercice, la commande équivalente avec `find` (attention, il y a un petit piège).

Supposons que la commande `wc -l` compte le nombre de lignes, on peut compter le nombre de fois que 42 apparaît dans le dossier `/etc`.

```
root@je:/etc$ grep -r "42" /etc | wc -l
783
```

Et on peut chaîner deux, trois, quatre (ou plus) commandes !

3.8 Se connecter sur un autre compte

La commande `su` te permet de te connecter à un autre compte. Par exemple, si tu veux te connecter au compte de « ydupont », il te suffit de faire

```
je@je:~$ su ydupont
```

Et si tu veux te connecter au compte *super-utilisateur*, tu peux faire simplement

```
je@je:~$ su
```

3.9 Pot-pourri de commandes

A présent, voici une liste de commandes potentiellement intéressantes. Malheureusement pour toi, je n'ai pas le courage de les décrire. Je te laisse donc le plaisir d'utiliser la commande `man` pour savoir à quoi elles servent et comment elles fonctionnent. Un informaticien doit savoir utiliser l'aide pour se renseigner !

3.9.1 Afficher

- `cat` : Afficher un fichier
- `more` : Afficher page par page
- `less` : Afficher ligne par ligne
- `tail -5 fichier` : Afficher les 5 dernières lignes d'un fichier
- `tail -5f fichier` : Lire en temps réel les 5 dernières lignes d'un fichier
- `head -5 fichier` : Afficher les 5 premières lignes d'un fichier
- `tac fichier` : Afficher un fichier à l'envers
- `file fichier` : Connaître le type d'un fichier
- `ls, ls-l` : Lister un répertoire
- `ls *, ls-R` : Afficher récursivement
- `ls -a` : Afficher les fichiers cachés
- `tr -d "0" < fichier` : Afficher le fichier en supprimant le caractère `0`
- `sed 's/0/gag/g' fichier` :
  Afficher le fichier en remplaçant le caractère `0` par `gag`

3.9.2 Créer

- `> fichier, touch fichier` : Créer un fichier vide
- `mkdir repertoire` : Créer un répertoire
- `mkdir -p rep_parent/rep1/rep2/rep3` :
  Créer un répertoire et ses sous-répertoires

3.9.3 Rediriger

- `ls -R /home/$USER/*.txt > liste.txt` : Dans un fichier
- `ls -R /home/$USER/*.txt >> liste.txt` :
  Dans un fichier, mais en ajoutant à la suite
- `ls -R /home/$USER/*.txt | tee liste.txt` :
  A la fois à l'écran et dans un fichier
- `ls -R /home/$USER/*.txt | tee-aliste.txt` :
  A la fois à l'écran et dans un fichier, mais en ajoutant à la suite

3.9.4 Copier

- `cp fichier-source fichier-destination` : Copier un fichier
- `cp /rep-source/*.html /rep-destination` : Copier un ensemble de fichiers
- `cp -R /rep-source /rep-destination` : Copier un répertoire

3.9.5 Déplacer, renommer

- `mv -i /chemin/fichier /chemin` : Déplacer un fichier
- `mv /chemin/rep-a-deplacer/ /chemin/destination/` : Déplacer un répertoire
- `mv fichier-source fichier-destination` : Renommer un fichier

3.9.6 Trier

— `sort fichier` : Trier alphabétiquement
— `sort -n fichier` : Trier numériquement
— `sort fichier | uniq, sort-ufichier` : Éliminer les doublons
— `sort fichier | uniq-d` : Afficher uniquement les doublons

3.9.7 Compter

— `wc -c fichier` : Compter les caractères d'un fichier
— `wc -w fichier` : Compter les mots d'un fichier
— `wc -l fichier, sed-n '$=' fichier` : Compter les lignes d'un fichier

Quatrième partie
Gestionnaire de paquets

1 Préliminaire

Au cours du montage du serveur, tu vas installer de nombreux paquets. Il faut donc avoir un gestionnaire des paquets fonctionnel et mis à jour.

Ressources : https://wiki.debian.org/fr/SourcesList

2 Configuration des sources

Édite le fichier `/etc/apt/sources.list` (après t'être connecté à `root` puisqu'il n'y a pas encore *sudo*)...

```
nano /etc/apt/sources.list
```

... et modifie-le de manière à obtenir un fichier semblable à celui-ci :

```
deb http://ftp.fr.debian.org/debian stretch main
deb-src http://ftp.fr.debian.org/debian stretch main

deb http://ftp.fr.debian.org/debian stretch-updates main
deb-src http://ftp.fr.debian.org/debian stretch-updates main

deb http://security.debian.org/debian-security/ stretch/updates
    main
deb-src http://security.debian.org/debian-security/ stretch/
    updates main
```

En particulier, supprime (ou mets en commentaire) les lignes relatives au CDROM.

Ce que tu viens de faire, c'est d'indiquer à APT où aller chercher les paquets à télécharger.

Une autre méthode pour modifier ces *repositories*, c'est d'utiliser la commande `apt-add-repository`pour les ajouter et retirer. C'est souvent la méthode conseillée après la configuration initiale du serveur.

Mise à jour des paquets

```
apt-get update # Mise à jour du gestionnaire de paquets
apt-get upgrade # Mise à jour des paquets eux-mêmes
```

3 Installation de quelques paquets importants

À présent, pour installer des paquets avec le gestionnaire de paquets, il te suffit d'exécuter la commande

```
apt-get install name-of-the-package
```

Sudo : Le paquet *sudo* te permettra t'exécuter des commandes en tant qu'administrateur sans avoir à te connecter en tant que *root*.

```
apt-get install sudo
```

Pour ajouter l'utilisateur « je », il faut l'ajouter dans le groupe « sudo » avec la commande `adduser je sudo`.

Vim : La commande `vim` te permettra d'éditer les fichiers avec plus d'aisance que la commande `nano`. Tu peux également utiliser un autre éditeur si tu as déjà une préférence. Mais j'utiliserai `vim` dans la suite de ce guide.

```
apt-get install vim
```

`vim` est un éditeur de texte très puissant. Malheureusement, il n'est pas tellement intuitif car il faut connaître les raccourcis du programme. Il faut prendre un peu de temps pour le maîtriser, mais cela vaut le coup ! Pour commencer à apprendre les commandes, je te conseille de suivre le tutoriel de `vim` accessible via la commande

```
vimtutor
```

Si tu n'as pas le temps d'apprendre, voici une toute petite liste de raccourcis :

— « Echap » : Rentrer dans le mode console
— « i » : Rentrer dans le mode insertion
— « u » : Annuler la dernière modification
— « dd » : Supprimer/couper la ligne courante
— « yy » : Copier la ligne courante
— « p » : Coller
— « :w » : Sauvegarder les modifications
— « :q » : Quitter `vim`
— « /untruc » : Rechercher `untruc` dans le fichier
— « :!exec » : Exécuter la commande `exec` dans un terminal

Cinquième partie
Configuration réseau

Ressources : https://wiki.debian.org/fr/NetworkConfiguration

1 Pour configurer

Depuis Stretch, les nommages d'interfaces réseau comme `eth0`, `eth1` ont disparu puisque le nom peut changer. Les nouveaux noms ressemblent à : `enp6s0`, `enp8s0`, `enp0s31f6`, `enp5s0`. Il faut donc trouver les noms des interfaces de notre système. Pour cela, tape

```
ls /sys/class/net/
```

Dans mon cas, je trouve `enp0s25`. Le Pôle Informatique ne souhaite pas que l'adresse IP du serveur soit gérée par DHCP ; en effet, nous souhaitons que le serveur ait une adresse fixe. Pour cela, édite le fichier `/etc/network/interfaces`.

Actuellement, dans le local JE, l'IP publique arrive sur le routeur qui redirige les paquets réseau vers le serveur dans le sous-réseau.

```
#/etc/network/interfaces

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp0s25
iface enp0s25 inet static
        address 192.168.1.254
        gateway 192.168.1.1
        netmask 255.255.255.0
        network 192.168.1.0
        broadcast 192.168.1.255
```

Il ne reste plus qu'à indiquer au système un serveur DNS. Pour cela, va dans `/etc/`. Lance la commande suivante pour afficher les permissions du fichier `resolv.conf`

```
ls -l | grep "resolv.conf"
```

Vérifie le premier caractère des permissions du fichier.

— Si c'est « l », c'est qu'il s'agit d'un lien symbolique. Supprime `resolv.conf` et recrée-le.

```
lrw-r--r-- 1 root root resolv.conf
```

— Si ce n'est pas « l », édite-le.

```
-rw-r--r-- 1 root root resolv.conf
```

— S'il n'y avait pas de fichier, crée-le.

In fine, le contenu du fichier `resolv.conf` doit être

```
nameserver 192.168.1.1
```

Il s'agit de l'IP du routeur.

A présent, il faut redémarrer le serveur pour que les modifications soient prises en compte !

Pour tester

```
ping 8.8.8.8
ping www.google.fr
```

Si la seconde commande ne fonctionne pas alors que la première fonctionne, c'est qu'il y a un problème avec la configuration du serveur DNS.

Sixième partie
Gestion des utilisateurs et des groupes

Sur un système Linux, comme sur tout autre système d'explotation, il y a des utilisateurs et des groupes. Il est possible de les gérer via une série de commandes.

1 Les commandes indispensables

1.1 Commandes pour gérer les utilisateurs

— Pour lister les utilisateurs :

```
cat /etc/passwd | awk -F: '{print $ 1}'
```

— Pour ajouter des utilisateurs :

```
adduser the_name_of_the_user
```

— Pour retirer des utilisateurs :

```
deluser the_name_of_the_user
```

Attention ! Lorsqu'on supprime un utilisateur, son dossier `home` n'est pas supprimé. Si tu veux vraiment le supprimer, utilise la commande `rm`.

1.2 Commandes pour gérer les groupes

— Pour lister les groupes :

```
cat /etc/group | awk -F: '{print $ 1}'
```

— Pour créer des groupes :

```
addgroup the_name_of_the_group
```

— Pour retirer des utilisateurs :

```
delgroup the_name_of_the_group
```

— Pour ajouter un utilisateur à un groupe :

```
adduser the_name_of_the_user the_name_of_the_group
```

2 Et le serveur dans tout ça ?

On pourrait gérer tous les utilisateurs avec toutes ces commandes, mais cela ne serait pas pratique. A la place, on va utiliser un annuaire LDAP.

Septième partie
Serveur Web et site vitrine

Ressources :
— https://doc.ubuntu-fr.org/apache2
— https://letsencrypt.org/
— https://forum.ubuntu-fr.org/viewtopic.php?id=2012936

1 Serveur Web

1.1 Installation de Apache2

Un serveur HTTP permet à un serveur web de communiquer avec un navigateur en utilisant le protocole HTTP(S) et ses extensions (WebDAV, etc.). Apache est probablement le serveur HTTP le plus populaire.

Pour installer Apache et PHP, tape

```
apt-get install apache2 php
```

À la suite de cette installation, ton serveur Web devrait fonctionner et donc être accessible à l'adresse `http://localhost` (à partir de la même machine). Un message « It Works ! » devrait s'afficher dans ton navigateur. Il s'agit du contenu du fichier `/var/www/html/index.html` qui est affiché par défaut.

Si le serveur est derrière un routeur (ce qui était le cas à mon époque), il faut que tu rediriges les flux *HTTP* et *HTTPS* vers le serveur (dans la configuration du routeur), du moins si ce n'est pas déjà fait.

1.2 Mode de fonctionnement sommaire

Lorsqu'il démarre, Apache charge les fichiers de configuration et se met en attente de requêtes sur les interfaces réseaux. On dit qu'il écoute (*listen* en anglais) certains ports.

Lorsqu'on utilise un navigateur web, que l'on clique sur un lien ou qu'on rentre directement une URL dans la barre d'adresse, on effectue une requête :

1. Le navigateur résout le nom de domaine (il obtient l'adresse IP du serveur) ;
2. Il envoie une requête HTTP avec la méthode GET à l'IP du serveur sur le port 80 (ou HTTPS sur le port 443) pour lui demander de retourner un contenu particulier ;
3. Le serveur HTTP reçoit la requête et, en fonction de divers paramètres (URL appelée, configuration du serveur, . . .), va chercher un contenu dans un fichier ou lance un script qui va générer un contenu ;

4. Le serveur renvoie ce contenu à l'IP du navigateur ;
5. Le navigateur traite le contenu et le rend accessible à l'internaute (en l'affichant à l'écran par exemple).

Voilà comment fonctionne un serveur web.

1.3 Utilisation

1.3.1 Gestion globale

Pour arrêter Apache2 :

```
systemctl stop apache2
```

Pour lancer Apache2 :

```
systemctl start apache2
```

Pour relancer Apache2 :

```
systemctl restart apache2
```

Pour recharger la configuration d'Apache2 :

```
systemctl reload apache2
```

Pour connaître la version d'Apache utilisée :

```
apache2ctl -v
```

Pour tester l'ensemble de la configuration d'Apache :

```
apache2ctl -t
```

Pour tester la configuration des hôtes virtuels :

```
apache2ctl -t -D DUMP_VHOSTS
```

Pour connaître les modules d'Apache chargés :

```
apache2ctl -M
```

1.3.2 Gestion des fichiers de configuration

Un seul serveur Apache permet de déployer simultanément plusieurs sites et services qu'il faut configurer individuellement. Pour plus de clarté, la configuration d'Apache2 est donc morcelée, mais tous les fichiers de configuration se situent dans le répertoire `/etc/apache2` :

— `sites-available` contient les fichiers de configuration des sites disponibles ;

— `sites-enabled` contient des liens symboliques vers les configurations, dans `site-available`, des sites activés ;
— `conf-available` contient les fichiers de configuration des autres services disponibles ;
— `conf-enabled` contient des liens symboliques vers les configurations, dans `conf-available`, des autres services activés ;
— `mods-available` contient les fichiers de configuration des modules d'Apache disponibles ;
— `mods-enabled` contient des liens symboliques vers les configurations, dans `mods-available`, des modules activés.

Normalement, les fichiers de configuration globale `apache2.conf`, `envvars` et `ports.conf` n'ont pas à être modifiés. Toute la configuration devrait se faire dans les sous-dossiers `xxx-available`.

Les diverses configurations sont activées (`a2en` pour *Apache 2 enable*) ou désactivées (`a2dis` pour *Apache 2 disable*) avec les commandes suivantes :

```
a2ensite [configuration d'un site à activer]
a2dissite [configuration d'un site à désactiver]

a2enconf [configuration d'un service à activer]
a2disconf [configuration d'un service à désactiver]

a2enmod [configuration d'un module à activer]
a2dismod [configuration d'un module à désactiver]
```

Cela aura pour effet de créer ou supprimer les liens symboliques correspondants dans les répertoires `xxx-enabled`. Apache prendra alors en compte, ou pas, les fichiers de configuration concernés après rechargement :

```
systemctl reload apache2
```

Par défaut, Apache ne prend en compte que les fichiers portant l'extension `.conf` (ou `.load`, seulement pour les modules).

2 Gestionnaire des certificats SSL

Pour permettre aux utilisateurs d'accéder aux différents sites de manière sécurisée (via HTTPS), il faut avoir des certificats SSL. Pour t'en procurer, tu vas installer un petit utilitaire libre nommé *Let's Encrypt*.

2.1 Installation de Let's Encrypt

Pour installer l'utilitaire, tape

```
apt-get install git
git clone https://github.com/letsencrypt/letsencrypt /opt/
    letsencrypt --depth=1
/opt/letsencrypt/letsencrypt-auto
```

Quand on te demande une adresse mail, mets admin@telecom-etude.com. Cela te permettra de recevoir des notifications quand les certificats seront bientôt périmés ! Bien évidemment, accepte les *conditions d'utilisation*. Par contre, si on te demande si l'adresse mail peut être utilisée à des fins particulières, refuse.

Et pour terminer, quand il va t'être demandé de sélectionner des domaines pour créer des certificats, saute l'étape (on va utiliser une autre commande pour créer nos certificats).

2.2 Utilisation

2.2.1 Créer un certificat pour un nouveau domaine

Tape

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d mydomain.com -d mydomain.fr
/etc/init.d/apache2 start
```

2.2.2 Renouveler les certificats

Tape

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto renew
/etc/init.d/apache2 start
```

3 Serveur MySQL et PHPmyAdmin

3.1 Installation

Pour certaines applications, il est intéressant d'avoir une base de données SQL dans le serveur. *phpMyAdmin* est une application Web de gestion d'une base de données *MySQL* réalisée principalement en PHP. Comme il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur PHP, nous l'avons choisie pour répondre à nos besoins.

Pour installer *phpMyAdmin*, tape les commandes suivantes :

```
apt-get install mysql-server
apt-get install phpmyadmin
```

et utilise les informations suivantes pour le configurer :

Serveur web à reconfigurer automatiquement	Aucun
Configuration avec *dbconfig-common*	Oui
Mot de passe pour phpmyadmin	*Mot de passe usuel*

Configure Apache2 pour accéder à *phpMyAdmin*. Pour cela, les développeurs de *phpMyAdmin* ont écrit une configuration Apache2 et il te suffit de la mettre au bon endroit.

```
cp /etc/phpmyadmin/apache.conf /etc/apache2/conf-available/
    phpmyadmin.conf
a2enconf phpmyadmin
systemctl reload apache2
```

A présent, crée un compte administrateur pour *phpMyAdmin*. Pour cela, tape les commandes suivantes :

```
mysql
> GRANT ALL ON *.* TO 'admin'@'localhost' IDENTIFIED BY 'Mot de
    passe usuel' WITH GRANT OPTION;
> FLUSH PRIVILEGES;
> QUIT;
```

Et à présent, *phpMyAdmin* est prêt à l'emploi !

4 Site vitrine

Mettre en ligne le site vitrine est relativement simple. Il suffit de :
— Récupérer les fichiers constituant le site vitrine ;
— Rétablir le bon propriétaire des fichiers (`www-data`) ;
— Générer le certificat du site vitrine ;
— Ajouter un nouvel hôte virtuel dans *Apache2*.
Tu peux récupérer les fichiers en utilisant, par exemple, les sauvegardes automatisées du serveur.

```
scp -rp backup-server@192.168.1.250:/path-backup/var/www/website
    /var/www/
```

Définis `www-data` comme propriétaire des fichiers du dossier `/var/www/website` :

```
chown -R www-data:www-data /var/www/website
```

Génère un certificat pour le site vitrine :

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d telecom-etude.com -d telecom-etude.fr -d
    www.telecom-etude.com -d www.telecom-etude.fr
/etc/init.d/apache2 start
```

Crée le fichier `/etc/apache2/sites-available/website.conf` :

```
<VirtualHost *:443>
        ServerName www.telecom-etude.com
        ServerAlias telecom-etude.com telecom-etude.fr www.
            telecom-etude.fr
```

```
        ServerAdmin admin@telecom-etude.com

        DocumentRoot /var/www/website/
        <Directory ~ "/var/www/website/\.(?!well-know\/)">
                Order deny,allow
                Deny from all
        </Directory>

        Include /etc/letsencrypt/options-ssl-apache.conf
        SSLCertificateFile /etc/letsencrypt/live/telecom-etude.
            com/fullchain.pem
        SSLCertificateKeyFile /etc/letsencrypt/live/telecom-
            etude.com/privkey.pem

        ErrorLog /var/log/apache2/website.log
        CustomLog /var/log/apache2/access/website.log combined
        LogLevel warn
</VirtualHost>

<VirtualHost *:80>
        ServerName www.telecom-etude.com
        ServerAlias telecom-etude.com telecom-etude.fr www.
            telecom-etude.fr
        ServerAdmin admin@telecom-etude.com

        RewriteEngine On
        RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
</VirtualHost>
```

Et charge ce nouvel hôte virtuel !

```
a2enmod rewrite ssl # Si nécessaire
a2ensite website
/etc/init.d/apache2 reload
```

Huitième partie
L'annuaire LDAP

Ressources :
— https://www.digitalocean.com/.../how-to-install-and-configure-openldap
— https://guide.ubuntu-fr.org/server/samba-ldap.html
— http://wiki.gandi.net/fr/hosting/.../debian/mail-server-ldap
— http://wawadeb.crdp.ac-caen.fr/.../divers/ldap/sauvegarde_ldap.html
— http://tutoriels.meddeb.net/openldap-log-2/
— http://nadir.is.online.fr/index2.php?option=com_content&do_pdf=1&id=120

1 Préliminaire

Mais qu'est-ce que LDAP ?

> **Lightweight Directory Access Protocol (LDAP)** est à l'origine un protocole permettant l'interrogation et la modification des services d'annuaire (il est une évolution du protocole DAP). Ce protocole repose sur TCP/IP. Il a cependant évolué pour représenter une norme pour les systèmes d'annuaires, incluant un modèle de données, un modèle de nommage, un modèle fonctionnel basé sur le protocole LDAP, un modèle de sécurité et un modèle de réplication. C'est une structure arborescente dont chacun des noeuds est constitué d'attributs associés à leurs valeurs.
>
> — *Wikipédia, l'encyclopédie libre*

En gros, LDAP est une base de données qui contient (dans notre cas) la liste de tous les membres de Telecom Etude, la liste des groupes, la liste des appareils, la liste des emails, . . . Un annuaire est une base de données optimisée pour la lecture et la recherche. On y fait peu d'insertions et de modifications. LDAP est un annuaire avec tout un protocole pour communiquer. Celui-ci a la forme d'un arbre (graphe connexe acyclique avec une racine). C'est un peu comme le système de fichiers d'un ordinateur. Il y aura des répertoires et des sous-répertoires (ce sont les noeuds de l'annuaire), et il y aura des données comme les utilisateurs (ce sont les feuilles de l'annuaire).

Mais à quoi cela sert-il ?

Un tel annuaire permet de regrouper toutes les données des utilisateurs (et autres) au même endroit. Ainsi, en liant tous les services informatiques à l'annuaire LDAP, les utilisateurs pourront se connecter avec les mêmes identifiants. Cela est pratique et évite la redondance.

Si tu souhaites avoir plus de renseignements sur ce qu'est un annuaire LDAP, je peux te proposer de lire la page Wikipédia associée ou le mini-tutoriel « Présentation du concept d'annuaire LDAP » d'OpenClassrooms. Personnellement,

quasiment tout ce que je sais sur LDAP, je l'ai appris en lisant le livre « LDAP, administration système » de Gerald Carter. Le livre est plus complet que les deux sites proposés et présente plus que la théorie. Néanmoins, il commence à dater. Il faut donc être conscient en le lisant que la partie pratique a un peu évolué.

2 Installation du serveur LDAP

2.1 Installation du noyau de base

Commence par installer les paquets relatifs à LDAP :

```
apt-get install slapd ldap-utils
```

Durant l'installation, on te demandera de renseigner des informations. Tu peux rentrer n'importe quoi (tant que ce sont des données cohérentes) car tu auras l'opportunité de modifier ces données très rapidement.

A présent, tu peux reconfigurer le paquet *slapd*.

```
dpkg-reconfigure slapd
```

Cette fois-ci, renseigne les données suivantes (si on te propose d'omettre la configuration d'OpenLDAP, réponds « Non ») :

Nom de domaine	telecom-etude.com [1]
Nom d'entité	Telecom Etude
Mot de passe administrateur	*Mot de passe usuel*
Module	HDB [2]
Supprimer la base de données lors d'une purge	Non
Déplacer l'ancienne base de données	Oui
Autoriser LDAP2 (s'il apparaît)	Non

2.2 Installation des dépendances

L'annuaire LDAP est très typé ; chaque entrée dans l'annuaire possède un type précis. Certains types d'entrées doivent être installés manuellement pour être utilisables par la suite. Dans le cas du serveur de Telecom Etude, nous utilisons Samba et Postfix (qui seront installés plus tard dans ce guide). Néanmoins, nous allons d'ores et déjà installer les types correspondant à ces deux applications. Cela te permettra d'importer l'annuaire de la JE à la fin de cette section.

2.2.1 Préparer LDAP pour supporter Samba

Tout d'abord, installe les paquets suivants :

1. A partir de ce moment-là, un annuaire vide avec comme racine `dc=telecom-etude,dc=com` est créé.
2. D'après ce `site`, il vaudrait mieux choisir l'option MDB pour les versions les plus récentes de OpenLDAP.

```
apt-get install samba smbldap-tools
```

Pour pouvoir stocker des informations relatives à Samba dans LDAP, il faut définir des types spécifiques. Leur définition se trouve dans le fichier `/usr/share/doc/samba/examples/LDAP/samba.ldif.gz`. Pour les intégrer, exécute la commande suivante :

```
zcat /usr/share/doc/samba/examples/LDAP/samba.ldif.gz | ldapadd
    -Q -Y EXTERNAL -H ldapi:///
```

Pour interroger et voir ce nouveau schéma (pour voir si la commande a fonctionné), tape

```
ldapsearch -Q -LLL -Y EXTERNAL -H ldapi:/// -b cn=schema,cn=
    config 'cn=*samba*'
```

Maintenant que slapd connaît les attributs Samba, nous pouvons définir quelques indices basés sur ces attributs. L'indexation des entrées est un moyen d'améliorer les performances lorsqu'un client effectue une recherche filtrée dans le DIT.

Crée un fichier `samba_indices.ldif`...

```
vim samba_indices.ldif
```

... et ajoute le contenu suivant :

```
dn: olcDatabase={1}hdb,cn=config
changetype: modify
replace: olcDbIndex
olcDbIndex: objectClass eq
olcDbIndex: uidNumber,gidNumber eq
olcDbIndex: loginShell eq
olcDbIndex: uid,cn eq,sub
olcDbIndex: memberUid eq,sub
olcDbIndex: member,uniqueMember eq
olcDbIndex: sambaSID eq
olcDbIndex: sambaPrimaryGroupSID eq
olcDbIndex: sambaGroupType eq
olcDbIndex: sambaSIDList eq
olcDbIndex: sambaDomainName eq
olcDbIndex: default sub,eq
```

Si tu fais du "copier-coller" du site-ressource (`guide.ubuntu-fr.org`), adapte bien la première ligne.

A l'avenir, il faudra peut-être mieux adapter le fichier `samba_indices.ldif` en fonction du besoin.

Il ne reste plus qu'à charger ces indices dans LDAP :

```
ldapmodify -Q -Y EXTERNAL -H ldapi:/// -f samba_indices.ldif
```

Pour vérifier, exécute cette commande et tu devrais voir les nouveaux indices :

```
ldapsearch -Q -LLL -Y EXTERNAL -H ldapi:/// -b cn=config
    olcDatabase={1}hdb olcDbIndex
```

2.2.2 Préparer LDAP pour supporter Postfix

Tout d'abord, installe le paquet suivant :

```
apt-get install courier-ldap
```

On utilise ce paquet seulement pour avoir certains fichiers, donc **minimise l'installation** (pour « écriture d'un dossier . . . », mets « non »).

Va récupérer le fichier `authldap.schema` (décompresse-le s'il le faut) et mets-le dans le dossier `/etc/ldap/schema`.

```
gunzip /usr/share/doc/courier-authlib-ldap/authldap.schema.gz #
    si nécessaire
cp /usr/share/doc/courier-authlib-ldap/authldap.schema /etc/ldap
    /schema
```

Cela aurait été trop beau si le fichier ne contenait pas d'erreurs. Dans celui-ci, on nous décrit l'objet de type `CourierMailAccount` en nous indiquant qu'il peut éventuellement contenir un attribut de type `mailhost`. Or la description de cet attribut, qui se situe juste au-dessus dans le code, est commentée. Je t'invite donc à éditer le fichier `/etc/ldap/schema/authldap.schema`, et à supprimer les `#` devant les 4 premières lignes commentées décrivant cet attribut (attention, la ligne qui commence par Objects : 1.3.6(...) est, quant à elle, bel et bien un commentaire à garder).

Ce fichier a pour extension `.schema`, alors que, pour Samba, le fichier récupéré avait l'extension `.ldif`. Les fichiers `.schema` correspondent aux fichiers pour les vieilles versions du serveur LDAP. Maintenant, il faut des fichiers `.ldif`. Tu vas donc devoir convertir ce fichier.

Mais avant ceci, nous avons besoin de savoir quels sont les schémas que OpenLDAP intègre déjà. Pour ce faire, nous allons parcourir la base `cn=schema,cn=config` de notre arbre, et nous allons nous limiter à l'attribut `cn` (Common Name) car, les entrées correspondant aux schémas étant très fournies, le résultat en serait vite indigeste. Toujours dans le souci de rendre le schéma plus digeste et de t'en faire connaître davantage sur `ldapsearch`, nous allons également utiliser `-LLL` qui va nous permettre de supprimer les commentaires et versions LDIF. Ainsi, notre commande sera :

```
ldapsearch -Y EXTERNAL -H ldapi:// -b "cn=schema,cn=config" -LLL
    "(objectClass=*)" cn
```

Le résultat devrait être semblable à celui-ci :

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,
    cn=auth
SASL SSF: 0
```

```
dn: cn=schema,cn=config
cn: schema

dn: cn={0}core,cn=schema,cn=config
cn: {0}core

dn: cn={1}cosine,cn=schema,cn=config
cn: {1}cosine

dn: cn={2}nis,cn=schema,cn=config
cn: {2}nis

dn: cn={3}inetorgperson,cn=schema,cn=config
cn: {3}inetorgperson

dn: cn={4}samba,cn=schema,cn=config
cn: {4}samba
```

Ainsi, de notre résultat, nous pouvons conclure qu'OopenLDAP intègre cinq schémas (dans cette configuration), à savoir `core`, `cosine`, `nis`, `inetorgperson` et `samba`. Il nous faudra également ajouter l'extension `courier` que nous venons d'ajouter au répertoire `/etc/ldap/schema`.

Pour embrouiller les futurs membres du Pôle Informatique, nos prédécesseurs ont décidé d'ajouter à la main d'autres types pour gérer les alias mail. Il faut donc les rajouter !

Pour cela, crée le fichier `/etc/ldap/schema/authldap_extension.schema` et mets-y le contenu suivant :

```
#$Id: authldap_extension.schema,v 1.0 19/07/18 tfeneuil $
#
# OID prefix: 1.3.6.1.4.1.10018
#
# Attributes:  -
#
# Depends on: authldap.schema (which depends on nis.schema and
    cosine.schema), core.schema

objectclass ( 1.3.6.1.4.1.10018.1.2.4 NAME 'CourierGroupAlias'
  DESC 'Group mail aliasing/forwarding entry'
  SUP top AUXILIARY
  MUST ( mail $ memberUid )
  MAY ( maildrop $ mailsource $ description ) )

objectclass ( 1.3.6.1.4.1.10018.1.2.5 NAME '
    CourierMailMemberAlias'
  DESC 'Group mail aliasing/forwarding entry'
  SUP top STRUCTURAL
  MUST ( cn $ mail )
  MAY ( member $ maildrop $ mailsource $ description ) )
```

Une fois le fichier créé et rempli, tu peux créer un fichier qui inclura tous ces schémas (ceux déjà présents, ainsi que `courier` et notre schéma personnalisé).

```
mkdir /tmp/ldapconfig
cat > /tmp/ldapconfig/schemaInclude.conf << EOF
include /etc/ldap/schema/core.schema
include /etc/ldap/schema/cosine.schema
include /etc/ldap/schema/nis.schema
include /etc/ldap/schema/inetorgperson.schema
include /etc/ldap/schema/authldap.schema
include /etc/ldap/schema/authldap_extension.schema
EOF
```

Maintenant, nous allons exécuter la commande `slaptest`, qui va convertir ce fichier au format LDIF, ce qui permettra de l'inclure à notre configuration.

```
slaptest -f /tmp/ldapconfig/schemaInclude.conf
    -F /tmp/ldapconfig
```

Cette commande doit normalement te renvoyer un message de succès. `slaptest` est une commande uniquement reconnue par *root*. Il en résulte que la commande doit être exécutée avec `sudo` et que l'arborescence créée dans `/tmp/ldapconfig` ne peut être accessible que par *root*, à moins d'en changer les droits.

La commande `slaptest` a créé, dans le répertoire `/tmp/ldapconfig`, toute une arborescence correspondant à celle de la configuration de LDAP. Je t'encourage à te rendre dans le répertoire contenant nos schémas et à en observer le contenu :

```
cd /tmp/ldapconfig/cn=config/cn=schema
ls -l
```

Tu peux constater la présence d'un fichier `cn={x}authldap.ldif` (où `x` est un entier variable). Nous allons effectuer quelques modifications à ce fichier. Édite-le :

```
vim cn={x}authldap.ldif
```

Aux lignes 1 et 3, supprime les parties `{x}` (crochets inclus). En fait, il s'agit de nombres que ton serveur va attribuer automatiquement, en fonction des dépendances ; nous ne souhaitons pas intervenir là-dessus.

Sur la première ligne, complète le nom du chemin, en ajoutant :

```
,cn=schema,cn=config
```

Maintenant, rends-toi à la fin du fichier (avec *vim*, tape G). Supprime les 7 dernières lignes, qui sont là aussi des lignes que notre serveur LDAP va générer automatiquement (sur vim, tape simplement `7dd` après être monté au niveau de la première ligne à supprimer).

Fais les mêmes opérations pour le fichier `cn={x}authldap_extension.ldif` : suppression des `{x}` aux lignes 1 et 3, ajout de `,cn=schema,cn=config` à la ligne 1 et suppression des 7 dernières lignes.

À l'heure actuelle, tu as tes fichiers `cn={x}authldap.ldif` et `cn={x}authldap_extension.ldif`, prêts à être importés dans ta configuration LDAP. Pour ce faire, exécute la commande `ldapadd`, dont la syntaxe est assez proche de `ldapsearch` et `ldapmodify` :

```
ldapadd -Y EXTERNAL -H ldapi:// -f /tmp/ldapconfig/cn=config/cn=
    schema/cn={4}authldap.ldif
ldapadd -Y EXTERNAL -H ldapi:// -f /tmp/ldapconfig/cn=config/cn=
    schema/cn={5}authldap_extension.ldif
```

Normalement, si tout se passe bien, cette commande devrait te confirmer qu'une entrée a été ajoutée. Vérifie cela par toi-même en exécutant la commande suivante :

```
ldapsearch -Y EXTERNAL -H ldapi:// -b "cn=schema,cn=config" -LLL
     "(objectClass=*)" cn
```

Normalement, dans le retour renvoyé, tu devrais voir les schémas `authldap` et `authldap_extension`.

Voilà, il ne te reste plus qu'à supprimer les paquets installés lorsque nous avons installé *courier-ldap*, à savoir :

```
apt-get purge -y courier-ldap courier-authlib courier-authlib-
    ldap courier-base courier-doc
```

2.3 Journalisation du serveur OpenLDAP

Il faut absolument mettre en place une journalisation du serveur LDAP.

2.3.1 Du côté du serveur LDAP

Lance la commande suivante :

```
ldapsearch -Y EXTERNAL -H ldapi:/// -b cn=config "(objectClass=
    olcGlobal)" olcLogLevel -LLL > slapdlog.ldif
```

Cette commande crée le fichier slapdlog.ldif dont le contenu est le résultat de la requête LDAP exécutée par l'utilitaire `ldapsearch` :

```
dn: cn=config
olcLogLevel: none
```

La première ligne contient le **DN** (*distinguished name*) qui est l'identifiant unique de l'entrée. La deuxième ligne contient l'unique attribut demandé par la requête avec comme valeur : `none`. La génération de log est **désactivée** par défaut. Modifie ce fichier pour que son contenu devienne :

```
dn: cn=config
changeType: modify
replace: olcLogLevel
olcLogLevel: stats
```

Maintenant ce fichier **LDIF** contient une commande de modification : la deuxième ligne déclare qu'on veut modifier l'entrée, la troisième indique qu'il s'agit d'un remplacement de contenu de l'attribut `olcLogLevel` de cette entrée et la quatrième indique la nouvelle valeur de cet attribut. `stats` permet de générer les logs des connexions, des opérations et de leurs résultats ce qui est parfait pour une surveillance quotidienne. Pour exécuter la commande de ce fichier sur le serveur, on utilise :

```
ldapmodify -Y EXTERNAL -H ldapi:/// -f slapdlog.ldif
```

Si tu obtiens le message : `modifying entry "cn=config"`, l'opération a réussi. Normalement la prise en compte par le serveur est immédiate et aucun redémarrage n'est nécessaire. Le serveur envoie les logs produits au mécanisme de gestion des logs système. Il s'agit de **rsyslog** pour les versions récentes.

2.3.2 Et du côté de rsyslog

Crée un fichier de configuration dans le dossier `/etc/rsyslog.d/` en le nommant `10-slapd.conf` par exemple. Le chiffre dans le nom permet de classer les fichiers dans ce dossier. Ce fichier contient l'unique ligne :

```
local4.*    /var/log/slapd.log;slapdtmpl
```

`slapdtmpl` est un nom **au choix** qui désigne un format de présentation du contenu de ce fichier de log. Il s'agit donc de créer ce format, cela se fait dans le fichier de configuration de rsyslog :

```
vim /etc/rsyslog.conf
```

Dans l'éditeur, ajoute ce qui suit, en dessous de la première ligne à partir du début qui commence par `$template` (s'il n'y en a pas, mets le template avant le `$IncludeConfig /etc/rsyslog.d/*.conf`) :

```
$template slapdtmpl,"[%$DAY%-%$MONTH%-%$YEAR% %timegenerated
    :12:19:date-rfc3339%] %app-name% %syslogseverity-text% %msg
    %\n"
```

Redémarre *rsyslog* pour qu'il prenne en compte la nouvelle configuration.

```
/etc/init.d/rsyslog restart
```

2.3.3 Pour tester la journalisation

Pour vérifier si la journalisation fonctionne bien, il suffit de lancer une requête et de consulter le contenu du fichier `/var/log/slapd.log`.

```
ldapsearch -Y EXTERNAL -H ldapi:/// -b dc=telecom-etude,dc=com
less /var/log/slapd.log
```

2.4 Les droits des fichiers : un sujet épineux

Lors de l'installation du serveur LDAP, tu as utilisé des commandes sudo ou tu as utilisé le compte *root*. Cela signifie que les fichiers de l'annuaire auront *root* comme propriétaire, ce qui est problématique. Nous allons donc rectifier tout cela en modifiant le propriétaire des données. Pour ce faire, tape

```
chown -R openldap:openldap /var/lib/ldap
```

où openldap est le compte de service créé automatiquement lors de l'installation du serveur.

3 Utilisation

Pour démarrer ou arrêter le service *slapd* :

```
/etc/init.d/slapd start
/etc/init.d/slapd stop
```

Une petite explication des fichiers :
— Les fichiers de configuration sont dans le répertoire /etc/ldap/.
— Les fichiers contenant les données de l'annuaire sont dans le répertoire /var/lib/ldap.
Concernant la partie réseau :
— LDAP utilise le port 389/tcp pour les connexions non sécurisées.
— LDAP utilise le port 636/tcp pour les connexions sécurisées.

3.1 Exporter et importer un annuaire LDAP

3.1.1 Sauvegarder

En utilisant slapcat, on peut exporter l'annuaire après avoir arrêté le service LDAP.

```
/etc/init.d/slapd stop
slapcat -b "dc=telecom-etude,dc=com" -l export.ldif
/etc/init.d/slapd start
```

3.1.2 Restaurer

Si tu as peur de faire de mauvaises opérations sur l'annuaire ou si tu as tout simplement un mauvais pressentiment, tu peux faire une archive en utilisant la commande suivante :

```
tar -czf var_lib_ldap.tar.gz /var/lib/ldap
```

Il faut arrêter le service LDAP, ajouter le contenu du fichier LDIF, réindexer et redémarrer le service LDAP :

```
/etc/init.d/slapd stop
rm -f /var/lib/ldap/*  # A executer pour partir d'un annuaire
    vide, sinon à ignorer
slapadd -b "dc=telecom-etude,dc=com" -l export.ldif
slapindex
/etc/init.d/slapd start
```

Si certaines entrées sont déjà présentes, des erreurs vont s'afficher, mais elles ne prêtent pas à conséquence.

Important ! Si le démarrage échoue, c'est sûrement qu'il y a un problème de droits. Dans ce cas, applique les instructions du paragraphe « Les droits des fichiers : un sujet épineux ».

4 Installation de la connexion sécurisée

Tout d'abord, si ce n'est pas déjà fait, crée un certificat pour le domaine de `ldap.telecom-etude.com`.

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d ldap.telecom-etude.com -d ldap.telecom-
    etude.fr
/etc/init.d/apache2 start
```

Il faudra que l'utilisateur `openldap`, celui avec lequel tourne le serveur LDAP, puisse accéder aux certificats. Pour cela, la manière la plus facile de procéder est de rajouter `openldap` dans le groupe `root` et de modifier les permissions des répertoires contenant les certificats.

```
adduser openldap root
cd /etc/letsencrypt
chmod 750 archive live
```

A présent, configurons le serveur LDAP pour qu'il ait connaissance du certificat. Pour cela, crée un fichier de configuration `tls-config.ldif`...

```
dn: cn=config
changeType: modify
add: olcTLSCACertificateFile
olcTLSCACertificateFile: /etc/letsencrypt/live/ldap.telecom-
    etude.com/fullchain.pem
-
add: olcTLSCertificateFile
olcTLSCertificateFile: /etc/letsencrypt/live/ldap.telecom-etude.
    com/cert.pem
-
add: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/letsencrypt/live/ldap.telecom-
    etude.com/privkey.pem
```

```
-
add: olcTLSVerifyClient
olcTLSVerifyClient: never
```

... et importe-le dans le serveur.

```
ldapmodify -QY EXTERNAL -H ldapi:// -f tls-config.ldif
```

Pour terminer, configure LDAP pour qu'il utilise le protocole LDAPS en modifiant le paramètre `SLAPD\_SERVICES` du fichier `/etc/default/slapd`.

```
SLAPD_SERVICES="ldap:/// ldaps:/// ldapi:///"
```

N'oublie pas de redémarrer le service.

```
/etc/init.d/slapd restart
```

5 Installation d'un navigateur LDAP graphique

Avant 2018, Le Pôle Informatique utilisait *phpLDAPadmin* comme navigateur LDAP. Le problème est que ce logiciel n'est plus entretenu depuis un certain temps et qu'il présente des failles de sécurité lors du passage à Debian 9. C'est pour cette raison que nous utilisons à présent *LDAP Account Manager*, qui est le successeur de *phpLDAPadmin*.

Pour installer *LDAP Account Manager* (*LAM*), il te suffit de taper

```
apt-get install php-zip php-xml ldap-account-manager
```

Après, va à la page `http://telecom-etude.com/lam` avec un ordinateur équipé d'un système d'exploitation graphique. A ce niveau-là, tu devrais avoir une page d'authentification de *LAM*.

5.1 Création du certificat SSL

Si ce n'est pas déjà fait, crée un certificat pour le domaine de `ldap.telecom-etude.com`.

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d ldap.telecom-etude.com -d ldap.telecom-
    etude.fr
/etc/init.d/apache2 start
```

5.2 Configuration Apache2

Nous allons à présent changer la configuration de *Apache2*. Lors de l'installation de *LAM*, *Apache2* a été configuré automatiquement de manière à ce que

le navigateur LDAP soit accessible à l'url http://telecom-etude.com/lam. Nous souhaitons modifier la configuration de telle sorte que le navigateur soit accessible avec l'url http://ldap.telecom-etude.com/.

Lors de son installation, *LAM* a installé le fichier /etc/apache2/conf-available/ldap-account-manager.conf. A la place, nous allons configurer un hôte virtuel. Pour cela, nous allons copier le fichier de configuration et l'adapter à nos besoins.

```
cd /etc/apache2/
cp conf-available/ldap-account-manager.conf sites-available/
vim sites-available/ldap-account-manager.conf
```

Modifie le fichier /etc/apache2/sites-available/ldap-account-manager.conf de la manière suivante :

— Modifie la ligne `Alias /lam /usr/share/ldap-account-manager` en `DocumentRoot /usr/share/ldap-account-manager`...
— ... et encadre tout le contenu du fichier par le code suivant (« ... » représente le contenu actuel).

```
<VirtualHost *:80>
    ServerName ldap.telecom-etude.com
    ServerAlias ldap.telecom-etude.fr
    ServerAdmin admin@telecom-etude.com

    RewriteEngine on
    RewriteRule ^(.*) https://%{SERVER_NAME}$1 [R,L]
</VirtualHost>

<VirtualHost *:443>
    ServerName ldap.telecom-etude.com
    ServerAlias ldap.telecom-etude.fr
    ServerAdmin admin@telecom-etude.com

    Include /etc/letsencrypt/options-ssl-apache.conf
    SSLCertificateFile /etc/letsencrypt/live/ldap.telecom-
        etude.com/fullchain.pem
    SSLCertificateKeyFile /etc/letsencrypt/live/ldap.
        telecom-etude.com/privkey.pem

    ...
</VirtualHost>
```

Il ne te reste plus qu'à rendre opérationnelles tes modifications.

```
a2disconf ldap-account-manager
a2enmod rewrite ssl # Si nécessaire
a2ensite ldap-account-manager
/etc/init.d/apache2 reload
```

5.3 Configuration des paramètres généraux

Appuie sur le bouton « LAM Configuration » en haut à droite. Clique sur « Edit general settings » et renseigne le *default master password* « lam ». A présent, configure l'outil à l'aide des tableaux suivants :

Security settings

Session timeout	30 (minutes)
Allowed hosts	∅
Encrypt session	Yes
Certificate SSL	*Upload le fichier* `cert.pem`

Password policies

Minimum password length	8
Minimum lowercase characters	1
Minimum uppercase characters	1
Minimum numeric characters	1
Minimum symbolic characters	0
Minimum characters classes	3
Number of roles that must match	All
Password must not contain user name	Unselected
Password must not contain part of user/first/last name	Unselected

Logging

Log level	Warning
Log destination	System logging
PHP error reporting	Default

Change master password

Renseigne le *mot de passe usuel*.

5.4 Configuration d'un profil Telecom Etude

Appuie sur le bouton « LAM Configuration » en haut à droite. Clique sur « Edit serveur profile » et ensuite sur « Manage server profile ».

Dans la section « Add profile », renseigne :

Profile name	TE
Profile password	*Mot de passe usuel*
Template	Unix

Pour procéder à l'ajout, *LAM* va te demander le mot de passe principal. Maintenant, ce n'est plus « lam », mais c'est ce que tu as renseigné dans les paramètres généraux.

Il ne te reste plus qu'à configurer le profil...

5.4.1 General settings

Server settings

Server address	ldap ://localhost :389
Activate TLS	no
Tree suffix	dc=telecom-etude,dc=com
LDAP search limit	10 000
Display name	Telecom Etude LDAP

Language settings

Default language	Français (France)
Time zone	Europe / Paris

Lamdaemon settings

Laisse les valeurs par défaut (du moins, dans un premier temps, *cf* la section spécifique à *lamdaemon*).

Tool settings

Laisse les valeurs par défaut.

Security settings

Login method	Fixed list
List of valid users	cn=admin,dc=telecom-etude,dc=com

5.4.2 Account types

Active les types de compte suivants : *Users*, *Groups*, *Mail aliases* et *Hosts*. Configure-les ensuite de la manière suivante :

Users accounts (e.g. Unix, Samba and Kolab)

LDAP suffix	ou=People,dc=telecom-etude,dc=com
List attributes	#uid ;#givenName ;#sn ;#uidNumber ;#gidNumber

Groups accounts (e.g. Unix and Samba)

LDAP suffix	ou=Group,dc=telecom-etude,dc=com
List attributes	#cn ;#gidNumber ;#memberUID ;#description

Mail aliases (e.g. NIS mail aliases)

LDAP suffix	ou=Aliases,dc=telecom-etude,dc=com
List attributes	#cn ;#rfc822MailMember

Hosts accounts (e.g. Samba)

LDAP suffix	ou=Hosts,dc=telecom-etude,dc=com
List attributes	#cn ;#description ;#uidNumber ;#gidNumber

5.4.3 Modules

Sélectionne les modules suivants pour chaque type de compte :

User	Personal (inetOrgPerson), Unix (posixAccount), Shadow (shadowAccount), Samba3 (sambaSamAccount)
Groups	Unix (posixGroup)
Mail aliases	Mail aliases (nisMailAlias)
Hosts	Account (account), Samba3 (sambaSamAccount), Unix (posixAccount)

5.5 Configuration du démon *lamdaemon*

Attention, cette partie n'a jamais été réalisée sur le serveur de Telecom Etude. Elle a néanmoins été testée sur des serveurs personnels d'anciens membres du Pôle Informatique.

La gestion actuelle des utilisateurs via *LAM* au sein de Telecom Etude est assez complexe, principalement à cause de la structure complexe de l'annuaire LDAP (bien qu'il a été simplifié au début de l'année 2019). Le Pôle Informatique n'utilise, tout au plus, que la vue arborescente de *LAM*, se privant ainsi de l'interface de base que le logiciel propose pour gérer les utilisateurs.

Dans l'interface de base de *LDAP Account Manager* (celle qu'on voit quand on se connecte), il est possible de gérer directement les informations des différents utilisateurs au sein de l'annuaire. Mais *LAM* propose des fonctionnalités supplémentaires pour gérer les utilisateurs, comme par exemple, la possibilité de gérer leur répertoire personnel. Néanmoins, ces fonctionnalités nécessite de mettre en place le démon *lamdaemon*. C'est pourquoi, dans cette section, je vais vous présenter ce démon.

Tout d'abord, crée un nouvel utilisateur dans l'annuaire LDAP, l'utilisateur « lamdaemon ». Voici l'entrée qu'il doit avoir dans l'annuaire. Tu peux utiliser la vue arborescente de *LAM*.

```
RDN: cn=lamdaemon,dc=aprilas,dc=fr
gidNumber: 1 (daemon)
homeDirectory: /
```

```
objectClass: inetOrgPerson, top, posixAccount, shadowAccount
sn: lamdaemon
iuid: lamdaemon
uidNumber: 1001
userPassword: mot-de-passe-usuel
```

Et maintenant, installe le démon.

```
apt-get install ldap-account-manager-lamdaemon
```

Va dans la configuration du profil du serveur (page d'accueil de *LAM*, « Configuration de LAM », « modifier les profils »). Dans la section « Paramètres de lamdaemon », complète le formulaire avec les informations suivantes.

Server	localhost
Path to external script	`/usr/share/ldap-account-manager/lib/lamdaemon.pl`
User name	lamdaemon

Puis, dans un terminal du serveur, édite le fichier `/etc/sudoers.d`

```
visudo -f /etc/sudoers.d/10-lam
```

et donne à `lamdaemon` l'autorisation d'exécuter son programme dédié.

```
lamdaemon ALL=(ALL) NOPASSWD: /usr/share/ldap-account-manager/
    lib/lamdaemon.pl *
```

Et pour finir, exécute manuellement le script une première fois.

```
su lamdaemon
sudo /usr/share/ldap-account-manager/lib/lamdaemon.pl
```

C'est bon ! `lamdaemon` est opérationnel ! Si tu veux en être convaincu, tu peux te connecter sur *LAM* et aller dans la section des tests qui propose quelques tests pour `lamdaemon`.

A présent, quand tu crées un nouvel utilisateur via l'onglet « Users », après avoir renseigné toutes les informations de base, tu peux cocher la case « Create home directory » dans la section « Unix ». Et une fois un utilisateur créé, tu peux retourner dans ses paramètres et supprimer son répertoire personnel si tu changes d'avis.

Neuvième partie
Authentification et droits

Ressources :

— https://doc.ubuntu-fr.org/ldap_client
— https://docs.oracle.com/cd/E19253-01/816-5174/nsswitch.conf-4/index.html
— https://www.ssi.gouv.fr/uploads/2015/10/NP_Linux_Configuration.pdf

1 Authentification

1.1 Préliminaire

`nscd` est un démon qui fournit un cache pour les requêtes aux services des noms les plus courants.

En plus de `nscd`, installe `ssh` si ce n'est pas encore fait.

```
apt-get install nscd
```

1.2 Configuration de nsswitch.conf

Installe le paquet `libnss-ldap` et suis le tableau ci-dessous pour la configuration :

URI du serveur LDAP	`ldapi:///127.0.0.1`
DN de la base de recherche	`dc=telecom-etude,dc=com`
Version LDAP à utiliser	3
Compte LDAP pour le superutilisateur	`cn=admin,dc=telecom-etude,dc=com`
Mot de passe root	*Mot de passe usuel*
Rendre local	Oui (?)
Est-ce que l'annuaire LDAP nécessite un mot de passe ?	Non
Compte LDAP pour l'administration	`cn=admin,dc=telecom-etude,dc=com`
Mot de passe du compte précédent	*Mot de passe usuel*

Édite le fichier `/etc/nsswitch.conf` :

```
vim /etc/nsswitch.conf
```

... et remplace l'option « compat ») par les options « files ldap » pour les bases de données *passwd*, *group* et *shadow*.

```
passwd :         files ldap
group :          files ldap
shadow :         files ldap
```

Puis redémarre le démon `nscd` :

```
/etc/init.d/nscd restart
```

Tu peux tester la configuration avec les commandes plus bas (remplace `someldapuser` par un nom d'utilisateur et `someldapgroup` par un nom de groupe présent dans ton serveur LDAP) :

```
getent passwd someldapuser
getent group someldapgroup
```

1.3 Configuration de PAM

Normalement, en installant `libnss-ldap`, tu as installé le paquet `libpam-ldap` et tu l'as déjà configuré. Si ce n'est pas le cas, installe-le et utilise le tableau de la partie précédente pour la configuration.

La configuration de PAM est divisée en 4 fichiers qui sont dans le dossier `/etc/pam.d/` : `common-account`, `common-auth`, `common-password` et `common-session`. Si tu veux en savoir plus sur PAM, tu peux lire sur ce sujet le document qui est sur le *Wiki SOS*.

Vérifie que, dans ces quatre fichiers, il y a une ligne avec `pam-ldap.so`. Si c'est le cas, ne touche à rien dans ce fichier, pauvre fou ! Lors de mes premiers tests, j'ai modifié ces fichiers en suivant les instructions d'un site. Mais il devait y avoir un problème dans ma configuration. Résultat : il m'était impossible de me connecter sur l'ordinateur (via la page d'accueil, via *ssh*, via *su*), même en tant que *super-utilisateur*. Je rappelle que la PAM sert à l'authentification de nombreux modules. Pour information, j'ai réussi à éviter la réinstallation du système en rétablissant le contenu des fichiers de configuration de PAM en me connectant en *root* sans mot de passe, grâce à une faille de sécurité qui est présente sur de nombreux linux s'ils ne sont pas protégés. Il suffit, au redémarrage de l'ordinateur, de modifier les commandes de démarrage de Grub2.

2 Distribution des droits avec *sudo*

2.1 Préliminaire

Le paquet *sudo* permet à un administrateur système d'accorder à certains utilisateurs (ou groupes d'utilisateurs) la possibilité de lancer une commande en tant qu'administrateur, ou comme autre utilisateur, tout en conservant une trace des commandes saisies et des arguments.

2.2 Configuration du *sudoers*

La configuration de l'usage de *sudo* se fait à partir du fichier `/etc/sudoers`. Il ne faut pas faire de bêtises avec ce fichier, car cela peut devenir rapidement une faille de sécurité. Si tu es motivé, je te conseille de lire les Recommandations de configuration d'un système GNU/Linux de l'ANSSI.

En réalité, on n'a pas besoin de grand chose. Il suffit juste d'autoriser le Pôle Informatique à utiliser sans limitation le sudo (parce qu'on a confiance en ce pôle).

Il suffit donc d'ajouter la ligne

```
%poleinfo ALL=(ALL:ALL) ALL
```

en dessous de la ligne

```
%sudo ALL=(ALL:ALL) ALL
```

On pourrait aussi ajouter un fichier dans /etc/sudoers.d/ au lieu de modifier le fichier de base /etc/sudoers, mais je pense que cela n'en vaut pas la peine pour une ligne.

Par contre, si, pour une application quelconque, tu dois rajouter des règles pour sudo, privilégie la création d'un fichier.

Si cela t'intéresse, voici la syntaxe d'une ligne du fichier /etc/sudoers :

```
identifiant ALL = (user) /chemin/complet/commande,/chemin/
    complet/autrecommande
%groupe ALL = (user) /chemin/complet/commande,!/chemin/complet/
    autrecommande
```

— identifiant représente un identifiant utilisateur du système Debian ;
— %groupe désigne un groupe d'utilisateurs du système Debian. Le nom du groupe doit donc être précédé d'un symbole de pourcentage (%) ;
— ALL désigne la ou les machines dans lesquelles les commandes suivantes sont autorisées ou refusées pour cet utilisateur ou ce groupe d'utilisateurs. Le mot-clé ALL désigne l'ensemble des machines de ton parc informatique. Dans le cadre d'une utilisation à domicile, laisser ALL n'est pas un inconvénient. Dans un grand parc d'entreprise, de meilleures stratégies sont à prévoir ;
— user (entre parenthèses) désigne l'utilisateur dont on prend les droits (peut valoir ALL pour tous) ;
— commande et autrecommande représentent des commandes pouvant être exécutées par l'utilisateur ou le groupe d'utilisateurs désigné en début de ligne ;
— les commandes précédées d'un point d'exclamation (**!**) sont refusées, alors que celles sans point d'exclamation sont autorisées. L'utilisation des points d'exclamation peut facilement induire une faille de sécurité comme expliqué par l'ANSSI dans ses recommandations ;
— les commandes multiples sont séparées par une virgule, sans espace ;
— les commandes doivent être entrées de manière exacte. Pour cette raison, préfère saisir des chemins absolus vers des commandes plutôt que des chemins relatifs (par exemple, /usr/sbin/update-manager plutôt que update-manager). Pour connaître le chemin absolu d'une commande ou d'un utilitaire, saisis dans un terminal which commande, ou whereis commande en remplaçant commande par la commande en question.

Dixième partie
Serveur Mail

Ressources :
— http://wiki.gandi.net/fr/.../tutorials/debian/mail-server-ldap
— http://www.postfix.org/postconf.5.html
— https://www.binarytides.com/linux-mail-command-examples/

1 Installation d'un serveur de redirection de mails

Pour installer le serveur mail, installe le paquet Postfix (et son extension pour l'utiliser avec ldap)...

```
apt-get install postfix postfix-ldap
```

en utilisant les options suivantes :

Type de configuration	Site internet
Nom de courrier	telecom-etude.com

Procédons à la configuration de Postfix ! Pour cela, édite le fichier `/etc/postfix/main.cf` et fais-le correspondre à la configuration suivante :

```
# See /usr/share/postfix/main.cf.dist for a commented, more
    complete version

smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
# appending .domain is the MUA's job.
append_dot_mydomain = no
compatibility_level = 2

# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/
    smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/
    smtp_scache

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc
    package for
# information on enabling SSL in the smtp client.

### Global properties of the server
# Debian specific:  Specifying a file name will cause the first
```

```
# line of that file to be used as the name.  The Debian default
# is /etc/mailname.

# Format of $myhostname: je.$mydomain
myhostname = je.telecom-etude.com
myorigin = /etc/mailname
mydestination = je, je.enst.fr, telecom-etude.fr, $myhostname,
    $mydomain, localhost.$mydomain, localhost
mynetworks = 137.194.26.0/23 127.0.0.0/8 [::ffff:127.0.0.0]/104
    [::1]/128
# 137.194.26.0/23 is the network with je.enst.fr
    (137.192.26.162)
inet_interfaces = all

### Mailbox properties
# Email to unix account will be stored in $home_mailbox
home_mailbox = Maildir/
mailbox_size_limit = 0

### Recipient alias and redirection
alias_maps = ldap:/etc/postfix/ldap-aliases.cf,hash:/etc/aliases
alias_database = hash:/etc/aliases
relayhost = smtp2.enst.fr
recipient_delimiter = +


### Filter option
# No filter for the moment
# Maybe later ?
```

puis crée le fichier `/etc/postfix/ldap-aliases.cf` contenant

```
server_host = 127.0.0.1
search_base = dc=telecom-etude,dc=com
query_filter = (&(mail=%s@telecom-etude.com)(|(objectClass=
    courierMailMemberAlias)(objectClass=courierMailAlias)(
    objectClass=courierGroupAlias)))
result_attribute = maildrop, memberUid
special_result_attribute = member
```

Quelques explications sur cette configuration :

— lorsqu'un utilisateur envoie un email avec son compte unix, son adresse email sera `$login@$myorigin`. Par exemple, puisque `$myorigin` vaut `telecom-etude.com` (si tu as suivi les étapes, le fichier `/etc/mailname` contient `telecom-etude.com`), quand *root* va envoyer un email, ce sera avec `root@telecom-etude.com`.

— le fichier `/etc/postfix/ldap-aliases.cf` te permet d'indiquer à Postfix d'aller vérifier que l'email de destination ne soit pas un alias dans l'annuaire LDAP. Si c'est le cas où c'en est un, l'email de destination est remplacé par les adresses qui ont le bon alias. Par exemple, quand le serveur va recevoir un email pour `info@telecom-etude.com`, il va regarder dans LDAP et

va trouver la liste des emails des membres du Pôle Informatique. Il va donc envoyer l'email à toutes ces personnes **au lieu** de l'envoyer à un utilisateur ayant réellement `info@telecom-etude.com` sur le système ;
— dans le cas où le destinataire de l'email traité n'est pas dans le domaine de Telecom Etude, le message est réexpédié via le serveur `smtp2.enst.fr` ;
— lorsqu'un email est à destination de Telecom Etude et n'est pas un alias, cela signifie qu'il est sous le format `$username@$domain` et il sera expédié à l'utilisateur du serveur avec le `username` correspondant. Les emails sont stockés sous forme de fichiers dans le dossier indiqué par la variable `$home_mailbox` (chemin relatif à partir du dossier `home` de l'utilisateur). Dans la configuration actuelle, on peut voir que les emails vont être stockés dans le sous-dossier `Maildir` du dossier `home` de l'utilisateur (le sous-dossier sera créé s'il n'existe pas).

La configuration de Postfix est terminée. Il ne te reste plus qu'à vérifier que tu n'as fait aucune erreur (la commande suivante ne doit renvoyer aucun retour) :

```
postfix check
```

Si tout est bon, tu peux recharger Postfix afin qu'il prenne en charge ta nouvelle configuration :

```
postfix reload
postfix/postfix-script: refreshing the Postfix mail system
```

Il ne te reste plus qu'à faire une ouverture de port au niveau du routeur de la JE pour rediriger les requêtes SMTP vers le serveur (port 25).

À présent, tu peux installer un petit utilitaire `mail` qui te permettra d'envoyer facilement des mails en ligne de commande.

```
apt-get install mailutils
```

Pour tester

Tu peux faire des tests (si ces tests ne fonctionnent pas, n'hésite pas à aller voir les logs `/var/log/` pour identifier la raison) :

— Envoi simple de mail sans passer par LDAP :

```
echo "Coucou !" | mail myself@enst.fr
```

— Envoi simple de mail en utilisant un alias mail LDAP :

```
echo "Coucou !" | mail myself@telecom-etude.com
```

— Redirection des emails externes en utilisant un alias mail LDAP : envoie un email à une adresse @telecom-etude.com avec ta messagerie mail.

2 Installation d'un serveur mail complet

Attention, cette installation n'a jamais été réalisée sur le serveur de Telecom Etude. Elle a néanmoins été testée sur des serveurs personnels d'anciens membres du Pôle Informatique.

Ressources :
— https://www.tecmint.com/install-postfix-mail-server-with-webmail-...
— https://www.laintimes.com/installer-...-un-serveur-mail
— https://philippe.scoffoni.net/comment-auto-config...-thunderbird

2.1 Configuration minimale du DNS

Crée une entrée MX sur le serveur DNS. Cela va indiquer au monde entier que les emails envoyés à ce domaine seront pris en charge par `mail.telecom-etude.com`.

```
3600 IN MX 1 mail.telecom-etude.com.
```

Puis, redirige ce sous-domaine vers ton serveur.

```
mail 3600 IN A 137.194.26.162
```

2.2 Préparation de la machine

Tout d'abord, tu vas préparer la machine pour accueillir le serveur mail. Commence par installer ces packages qui seront utiles pour l'administration système :

```
apt-get install curl net-tools bash-completion wget lsof
```

Ouvre le fichier `/etc/host.conf/` et ajoute la ligne suivante au début du fichier afin que la résolution DNS procède d'abord en lisant le fichier `/etc/hosts`.

```
order hosts,bind
multi on
```

Configure ta machine *FQDN* (*fully qualified domain name*), et ajoute le nom de domaine et ton système *FQDN* à `/etc/hosts/`. Utilise l'adresse publique du serveur. Et à la fin, redémarre-le.

```
hostnamectl set-hostname je.telecom-etude.com
echo "137.194.26.162 je je.telecom-etude.com" >> /etc/hosts
init 6
```

Après le redémarrage, vérifie que le `hostname` a été correctement configuré.

```
hostname # Must return "je.telecom-etude.com"
hostname -s # Must return "je"
hostname -f # Must return "telecom-etude.com"
hostname -A # Must return "telecom-etude.com"
cat /etc/hostname # Must return "je.telecom-etude.com"
```

2.3 Installation du serveur mail Postfix

Pour installer le serveur mail, installe le paquet Postfix...

```
apt-get install postfix
```

en utilisant les options suivantes :

Type de configuration	Site internet
Nom de courrier	telecom-etude.com

2.4 Configuration du serveur mail Postfix

Procédons à la configuration de *Postfix* ! Pour cela, édite le fichier `/etc/postfix/main.cf` et fais-le correspondre à la configuration suivante :

```
# See /usr/share/postfix/main.cf.dist for a commented, more
    complete version

smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
biff = no
# appending .domain is the MUA's job.
append_dot_mydomain = no
compatibility_level = 2

# TLS parameters
smtpd_tls_cert_file=/etc/letsencrypt/live/telecom-etude.com/
    fullchain.pem
smtpd_tls_key_file=/etc/letsencrypt/live/telecom-etude.com/
    fullchain.pem
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/
    smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/
    smtp_scache

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc
    package for
# information on enabling SSL in the smtp client.

### Global properties of the server

smtpd_relay_restrictions = permit_mynetworks
    permit_sasl_authenticated defer_unauth_destination

myorigin = $mydomain
# mydestination = $myhostname, $mydomain, localhost.$mydomain,
    localhost
```

```
mydestination = je.telecom-etude.com, telecom-etude.com,
    localhost.telecom-etude.com, je.telecom-etude.fr, telecom-
    etude.fr, localhost.telecom-etude.fr, localhost
mynetworks = 137.194.26.0/23 127.0.0.0/8 [::ffff:127.0.0.0]/104
    [::1]/128
# 137.194.26.0/23 is the network with je.enst.fr
    (137.192.26.162)
inet_interfaces = all
inet_protocols = all

### Mailbox properties
# Email to unix account will be stored in $home_mailbox
home_mailbox = Maildir/
mailbox_size_limit = 0

### Recipient alias and redirection
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
relayhost =
recipient_delimiter = +

### SMTP-Auth settings
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_local_domain = $myhostname
smtpd_recipient_restrictions = permit_mynetworks,
    permit_auth_destination,permit_sasl_authenticated,reject
```

Dans la configuration de *Postfix*, on a indiqué un certificat SSL. Si celui-ci n'est pas généré, alors tu peux le faire maintenant.

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d telecom-etude.com -d telecom-etude.fr
/etc/init.d/apache2 start
```

A présent, tu peux redémarrer *Postfix* pour appliquer les changements et t'assurer que le service est actif en vérifiant que le service de *Postfix* `master` écoute aux ports 25 (non-sécurisé par défaut) et 465 (sécurisé).

```
systemctl restart postfix
systemctl status postfix
netstat -tlpn
```

2.5 Test du serveur mail Postfix

Pour tester si *Postfix* peut procéder à des transferts de mails, installe le package `mailutils`.

```
apt-get install mailutils
```

Et teste l'envoi d'un email à `root`.

```
echo "mail body"| mail -s "test mail" root
ls /root/Maildir/new
```

Tu peux regarder de quelle manière le mail a été pris en charge par le service *Postfix* en lisant le fichier de log.

```
tailf /var/log/mail.log
```

2.6 Installation et configuration de Dovecot IMAP

Pour installer le serveur *Dovecot* et le paquet *Dovecot IMAP*, exécute la commande suivante :

```
apt install dovecot-core dovecot-imapd
```

Maintenant que *Dovecot* a été installé sur le système, ouvre le fichier de configuration `/etc/dovecot/dovecot.conf`, recherche et décommente la ligne suivante :

```
listen = *, ::
```

À présent, il y a une petite série de fichiers de configuration à éditer :

— Edite `/etc/dovecot/conf.d/10-auth.conf` et change les lignes suivantes :

```
disable_plaintext_auth = no
auth_mechanisms = plain login
```

— Edite `/etc/dovecot/conf.d/10-mail.conf` et ajoute la ligne suivante pour stocker les emails dans le répertoire `Maildir` :

```
mail_location = maildir:~/Maildir
```

— Edite `/etc/dovecot/conf.d/10-master.conf`. Recherche le bloc `smtp-auth` et effectue les changements suivants :

```
# Postfix smtp-auth
unix_listener /var/spool/postfix/private/auth {
  mode = 0666
  user = postfix
  group = postfix
 }
```

— Edite `/etc/dovecot/conf.d/10-ssl.conf` et modifie les paramètres suivants :

```
ssl = yes
ssl_cert = </etc/letsencrypt/live/telecom-etude.com/
    fullchain.pem
ssl_key = </etc/letsencrypt/live/telecom-etude.com/
    privkey.pem
```

Normalement, le certificat SSL pour `telecom-etude.com` existe déjà, puisque tu as dû le créer au minimum pour *Postfix*.

Après tous ces changements, redémarre le démon *Dovecot*, contrôle son statut et vérifie que *Dovecot* écoute aux ports 143 (non sécurisé) et 993 (sécurisé).

```
systemctl restart dovecot.service
systemctl status dovecot.service
netstat -tlpn
```

Vérifie si le serveur mail fonctionne correctement en envoyant un mail en te connectant au service SMTP via `netcat` (installe le paquet `netcat` si `nc` n'existe pas sur ton système).

```
nc localhost 25
ehlo localhost
mail from: root
rcpt to: je
data
subject: test
Mail body
.
quit
```

N'oublie pas de vérifier si le mail est bien arrivé dans la boîte mail.

```
ls /home/je/Maildir/new/
```

2.7 Installation d'un Webmail

On appelle « webmail » l'interface informatique permettant de lire, gérer et envoyer des emails depuis un navigateur Internet. Il en existe plein sur le Net, plus ou moins complets. Dans le cadre de ce guide, je vais te présenter l'installation de *Rainloop*, un logiciel qui peut fonctionner avec *Apache2* qui a déjà été installé sur le serveur.

Commence par télécharger *Rainloop*.

```
mkdir /var/www/rainloop
cd /var/www/rainloop
wget -qO- https://repository.rainloop.net/installer.php | php
```

Puis définis le bon propriétaire et les bonnes permissions.

```
cd /var/www/rainloop
find . -type d -exec chmod 755 {} \;
find . -type f -exec chmod 644 {} \;
chown -R www-data:www-data .
```

Mais, pour que *Rainloop* soit accessible sur un navigateur, il faut créer le *virtual host* correspondant dans *Apache2*. Pour cela, va dans `/etc/apache2/sites-available` et crée le fichier `rainloop.conf` avec le contenu suivant :

```
<VirtualHost *:80>
    ServerAdmin admin@telecom-etude.com
    ServerName webmail.telecom-etude.com
    ServerAlias webmail.telecom-etude.fr

    RewriteEngine On
    RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
</VirtualHost>

<VirtualHost *:443>
    ServerAdmin admin@telecom-etude.com
    ServerName webmail.telecom-etude.com
    ServerAlias webmail.telecom-etude.fr

    DocumentRoot /var/www/rainloop/
    ErrorLog /var/log/apache2/rainloop.log

    Include /etc/letsencrypt/options-ssl-apache.conf
    SSLCertificateFile    /etc/letsencrypt/live/webmail.telecom-
        etude.com/fullchain.pem
    SSLCertificateKeyFile  /etc/letsencrypt/live/webmail.telecom
        -etude.com/privkey.pem

    <Directory "`/var/www/rainloop/data/"'>
        Order allow,deny
        Deny from all
    </Directory>
</VirtualHost>
```

Il faut aussi créer le certificat SSL de `webmail.telecom-etude.com` si ce n'est pas déjà fait.

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d webmail.telecom-etude.com -d webmail.
    telecom-etude.fr
/etc/init.d/apache2 start
```

Il ne reste plus qu'à activer l'hôte virtuel.

```
a2ensite rainloop
systemctl reload apache2
```

Maintenant que *Rainloop Webmail* est installé et accessible, connecte-toi sur l'interface d'administration :

— Url : `https://webmail.telecom-etude.com/?admin`
— Identifiant : `admin`
— Mot de passe : `12345`

Commence tout d'abord par changer le mot de passe administrateur !

Ensuite, va dans « Domain » et clique sur « Add domain ». Remplis le formulaire avec les paramètres suivants :

Name	telecom-etude.com
Server IMAP	telecom-etude.com :993
Server IMAP - Secure	SSL/TLS
Use short login	Yes
Server SMTP	telecom-etude.com :465
Server SMTP - Secure	SSL/TLS
Use short login	Yes
Authentication	Yes

A présent, tu peux te connecter avec un compte normal.

— Url : `https://webmail.telecom-etude.com`
— Identifiant : `your-username@telecom-etude.com`
— Mot de passe : `your-password`

Tu peux maintenant essayer d'envoyer et de recevoir des emails via l'interface proposée.

2.8 Auto-configuration de comptes de messagerie

Certains logiciels de messagerie, comme *Thunderbird*, cherchent à déterminer automatiquement les paramètres d'un serveur mail, afin de simplifier la vie des utilisateurs de messagerie. Cette opération se réalise à partir de l'adresse email que tu as saisie pour ton nouveau compte de messagerie.

Pour *Thunderbird*, le mécanisme utilise plusieurs méthodes :

1. Une base de données ISPB gérée par Mozilla ;
2. Un fichier de configuration mis à disposition par l'hébergeur de messagerie ;
3. Un fichier de configuration sur le poste local ;
4. La devinette ; Thunderbird essaye de combiner smtp ou imap avec le nom de domaine de la messagerie ;
5. La configuration manuelle en dernier ressort.

Nous allons aborder la méthode numéro 2. Cette méthode se met en place à l'aide d'une entrée DNS et d'un fichier de configuration XML accessible via une URL spécifique.

Tout d'abord, crée un répertoire `autoconfig` dans `/var/www`. Crée un sous-répertoire `mail` avec un fichier de configuration qui devra s'appeler `config-v1.1.xml`.

```
mkdir -p /var/www/autoconfig/mail
vim /var/www/autoconfig/mail/config-v1.1.xml
```

Dans le fichier `config-v1.1.xml`, ajoute le contenu suivant :

```
<?xml version="1.0" encoding="UTF-8"?>
<clientConfig version="1.1">
        <emailProvider id="telecom-etude.com">
```

```
                <domain>telecom-etude.com</domain>
                <displayName>Services telecom-etude.com</
                    displayName>
                <displayShortName>telecom-etude.com</
                    displayShortName>
                <incomingServer type="imap">
                        <hostname>telecom-etude.com</hostname>
                        <port>993</port>
                        <socketType>SSL</socketType>
                        <authentication>password-cleartext</
                            authentication>
                        <username>%EMAILLOCALPART%</username>
                </incomingServer>
                <outgoingServer type="smtp">
                        <hostname>telecom-etude.com</hostname>
                        <port>465</port>
                        <socketType>SSL</socketType>
                        <authentication>password-cleartext</
                            authentication>
                        <username>%EMAILLOCALPART%</username>
                </outgoingServer>
        </emailProvider>
</clientConfig>
```

Maintenant, crée un nouveau *virtual host* dans *Apache2*. Pour cela, crée un fichier de configuration `/etc/apache2/sites-available/autoconfig.conf` avec le contenu suivant :

```
<VirtualHost *:80>
        ServerName autoconfig.telecom-etude.com
        ServerAdmin admin@telecom-etude.com

        DocumentRoot /var/www/autoconfig
</VirtualHost>
```

Il ne te reste plus qu'à activer ce *virtual host*.

```
a2ensite autoconfig
/etc/init.d/apache2 reload
```

Et pour finir, crée une entrée DNS pour `autoconfig.telecom-etude.com`.

```
autoconfig 3600 IN CNAME telecom-etude.com.
```

A présent, le fichier doit être accessible à l'URL : `http://autoconfig.telecom-etude.com/mail/config-v1.1.xml`.

Il ne te reste plus qu'à faire un test depuis *Thunderbird* pour vérifier que tout fonctionne bien. Lorsque tu ajoutes un nouveau compte, en indiquant ton adresse mail, *Thunderbird* devrait trouver automatiquement les bons paramètres, qui sont :

— Entrant : `telecom-etude.com:993` avec SSL/TLS et mot de passe normal
— Sortant : `telecom-etude.com:465` avec SSL/TLS et mot de passe normal

2.9 Configuration avancée pour éviter que les emails soient considérés comme des spams

Ressources :
— https://linuxfr.org/.../mails-consideres-comme-spam-par-gmail-config
— https://lea-linux.org/documentations/Installer_DKIM-SPF_sous_Postfix
— https://www.linode.com/.../postfix/configure-spf-and-dkim-in-postfix/

Si tu as configuré le serveur mail comme indiqué précédemment, il y a hélas de très fortes chances que les emails envoyés par `telecom-etude.com` soient considérés comme des spams par les autres messageries.

Une des raisons pour lesquelles un email peut être considéré comme un spam est que les messageries n'ont pas confiance dans le nom de domaine utilisé par l'email expéditeur. Il est facile de créer un petit script *Python* qui envoie un email avec une adresse email à laquelle on n'a pas accès. Par exemple, je pourrais très facilement envoyer un mail à `contact@telecom-etude.fr` de la part de `bill.gates@gmail.com`, alors que je n'ai bien évidemment pas accès à ce compte et que je ne sais même pas s'il existe en réalité. L'idée est donc de prouver au destinataire du mail que l'email a bien été envoyé par la personne qui a accès au compte de l'adresse mail en question.

Pour cela, nous allons utiliser deux méthodes : SPF et DKIM.

2.9.1 Sender Policy Framework (SPF)

> Le protocole *Simple Mail Transfer Protocol* (SMTP) utilisé pour le transfert du courrier électronique sur Internet ne prévoit pas de mécanisme de vérification de l'expéditeur, c'est-à-dire qu'il est facile d'envoyer un courrier avec une adresse d'expéditeur factice, voire usurpée. SPF vise à réduire les possibilités d'usurpation en publiant, dans le DNS, un enregistrement (de type *TXT*) indiquant quelles adresses IP sont autorisées ou pas à envoyer du courrier pour le domaine considéré.
>
> — *Wikipédia, l'encyclopédie libre*

L'idée est relativement simple. Lorsque le destinataire reçoit un email de la part de `thibauld.feneuil@telecom-etude.com`, il va demander au DNS qui gère `telecom-etude.com` l'enregistrement *SPF*. Cet enregistrement va lui dire que seule l'adresse IP `137.194.26.162` (par exemple) a le droit d'envoyer des emails avec ce nom de domaine. Alors, le destinataire va comparer cette adresse IP avec l'adresse IP du serveur qui a envoyé l'email reçu : si ce sont les mêmes, il garde le message, sinon il le considère comme un spam.

Bien entendu, dans cet exemple, j'ai dit que l'entrée SPF ne renvoie qu'une adresse IP, mais on peut retourner ce que l'on veut : une énumération d'IP, une plage d'IP, ...

Voici un exemple d'enregistrement SPF :

```
ietf.org. 720 IN TXT "v=spf1 ip4:12.22.58.0/24 ip6:2001:1890:123
    a::/56 ip4:64.170.98.0/24 ip6:2001:1890:126c::/56 ip4
    :4.31.198.32/27 ip6:2001:1900:3001:0011::0/64 ip4
    :209.208.19.192/27 ip6:2607:f170:8000:1500::0/64 ip4
    :72.167.123.204 -all"
```

Seuls les blocs d'adresses IPv4 et IPv6 indiqués sont habilités à envoyer du courrier avec un expéditeur du domaine `ietf.org`. Un serveur de courrier participant à SPF peut donc rejeter un mail provenant d'autres blocs d'adresses que ceux-ci.

Dans notre cas, l'enregistrement SPF à ajouter dans l'entrée DNS est simple. Il suffit d'indiquer l'adresse IP du serveur :

```
telecom-etude.com. 720 IN TXT "v=spf1 ip4:137.194.26.162 -all"
```

Le protocole *Sender Policy Framework* permet de réduire le spam, mais il n'est pas infaillible. Par exemple, lorsque le destinataire demande la liste des adresses IP autorisées au DNS, le pirate informatique pourrait intercepter la requête et lui retourner une fausse liste d'adresses IP.

C'est pourquoi on va rajouter une autre sécurité.

2.9.2 DomainKeys Identified Mail (DKIM)

> DKIM est une norme d'authentification fiable du nom de domaine de l'expéditeur d'un courrier électronique. Elle constitue une protection efficace contre le spam et l'hameçonnage.
>
> En effet, DKIM fonctionne par signature cryptographique du corps du message ou d'une partie de celui-ci et d'une partie de ses en-têtes. Une signature DKIM vérifie donc l'authenticité du domaine expéditeur et garantit l'intégrité du message. DKIM intervient au niveau de la couche application du modèle OSI, ainsi il constitue une double protection pour des protocoles de messagerie électronique tels que SMTP, IMAP et POP en plus de l'utilisation de ces protocoles en mode sécurisé (POPS, IMAPS).
>
> *— Wikipédia, l'encyclopédie libre*

DKIM fait appel à de la cryptographique asymétrique avec une clé publique et une clé privée. Pour simplifier, DKIM va hacher le message de l'email (avec une fonction de hachage cryptographique connue de tous). Avec la clé privée, l'expéditeur va chiffrer le *hash* et l'envoyer dans les entêtes du message. Le destinataire du message va, avec la clé publique qu'il aura récupérée grâce au serveur DNS, déchiffrer le *hash*. En parallèle, avec la même fonction de hachage, il va calculer le *hash*. Si ces deux valeurs correspondent, cela montre que le mail a été signé par le domaine indiqué, et n'a pas été altéré pendant le transit, car aucun pirate informatique ne peut déterminer le *hash* codé sans l'aide de la clé privée.

On va donc, dans la suite, générer une paire de clés (publique et privée), installer et configurer un service qui va signer les mails envoyés par Postfix à l'aide de la clé privée, et mettre la clé publique à disposition du monde.

Commence par installer les paquets suivants :

```
apt-get install opendkim opendkim-tools
```

Modifie le fichier de configuration d'OpenDKIM `/etc/opendkim.conf` afin qu'il ressemble à celui-ci :

```
# This is a basic configuration that can easily be adapted to
# suit a standard installation. For more advanced options, see
# opendkim.conf(5) and/or /usr/share/doc/opendkim/examples/
    opendkim.conf.sample.

# Log to syslog
Syslog          yes
# Required to use local socket with MTAs that access the socket
# as a non-privileged user (e.g. Postfix)
UMask           002
# OpenDKIM user
# Remember to add user postfix to group opendkim
UserID          opendkim

# Map domains in From addresses to keys used to sign messages
KeyTable        /etc/opendkim/key.table
SigningTable        refile:/etc/opendkim/signing.table

# Hosts to ignore when verifying signatures
ExternalIgnoreList  /etc/opendkim/trusted.hosts
InternalHosts       /etc/opendkim/trusted.hosts

# Commonly-used options;
# the commented-out versions show the defaults.
Canonicalization    relaxed/simple
Mode            sv
SubDomains      no
#ADSPAction     continue
AutoRestart     yes
AutoRestartRate     10/1M
Background      yes
DNSTimeout      5
SignatureAlgorithm  rsa-sha256

# Always oversign From (sign using actual From and a null From
# to prevent # malicious signatures header fields (From and/or
# others) between the signer and the verifier.  From is
# oversigned by default in the Debian package because it is
# often the identity key used by reputation systems and thus
# somewhat security sensitive.
OversignHeaders     From
```

```
# Define the location of the Socket and PID files
Socket                  local:/var/spool/postfix/opendkim/opendkim.
    sock
PidFile                 /var/run/opendkim/opendkim.pid
```

Assure-toi que les permissions de ce fichier sont correctement configurées.

```
chmod u=rw,go=r /etc/opendkim.conf
```

Crée un répertoire qui va contenir toutes les données d'OpenKIM.

```
mkdir /etc/opendkim
mkdir /etc/opendkim/keys
```

Crée le fichier `/etc/opendkim/signing.table`. Il a besoin d'avoir une ligne par nom de domaine que le service va prendre en charge. Chaque ligne doit ressembler à ceci :

```
*@example.com    example
```

Remplace `example.com` par le nom de domaine approprié et `example` par un diminutif pour le domaine. Le premier champ doit représenter les adresses mail du domaine (via un *pattern*). Le second champ est un nom qui permettra de faire le lien avec la table des clés, renseignant ainsi la clé à utiliser pour signer les emails envoyés à partir de ce domaine.

Dans le cadre de l'association, le contenu du fichier `/etc/opendkim/signing.table` devrait ressembler à

```
*@telecom-etude.com    te.com
*@telecom-etude.fr     te.fr
```

Et maintenant, crée la table des clés `/etc/opendkim/key.table`. Il doit y avoir une ligne par diminutif présent dans le fichier précédent avec le format suivant :

```
example     example.com:YYYYMM:/etc/opendkim/keys/example.private
```

en remplaçant `example` par le diminutif du domaine, `example.com` par le nom du domaine, `YYYYMM` par la date actuelle (cela sera ton sélecteur).

Le second champ est composé de trois sections séparées par des virgules.

— La première section est le nom de domaine avec lequel la clé privée est utilisée ;
— La deuxième section est un sélecteur pour désigner une clé publique spécifique dans les enregistrements du DNS ;
— La troisième section est le fichier qui contient la clé privée pour le domaine.

Dans le cadre de Telecom Etude, le fichier devrait être semblable à

```
te.com      telecom-etude.com:201908:/etc/opendkim/keys/telecom-
    etude.com.private
te.fr       telecom-etude.com:201908:/etc/opendkim/keys/telecom-
    etude.com.private
```

Crée le fichier des hôtes de confiance `/etc/opendkim/trusted.hosts`.

```
127.0.0.1
::1
localhost
je
je.telecom-etude.com
je.telecom-etude.fr
telecom-etude.com
telecom-etude.fr
```

Assure-toi que le propriétaire et les permissions de `/etc/opendkim` soient correctement configurés.

```
chown -R opendkim:opendkim /etc/opendkim
chmod -R go-rwx /etc/opendkim/keys
```

Et si tu générais la paire de clés publique-privée ? Utilise la commande suivante pour générer des paires de clés (il faut l'utiliser autant de fois qu'il y a de clés différentes dans `/etc/opendkim/key.table`).

```
cd /etc/opendkim/keys
opendkim-genkey -v -b 2048 -h rsa-sha256 -r -s 201908
    -d telecom-etude.com
mv 201908.private telecom-etude.com.private
mv 201908.txt telecom-etude.com.txt
```

Je te rappelle que `201908` *fait référence au sélecteur défini dans la table des clés.*

Assure-toi, à nouveau, que le propriétaire et les permissions de `/etc/opendkim` soient correctement configurés.

```
chown -R opendkim:opendkim /etc/opendkim
chmod -R go-rw /etc/opendkim/keys
```

Et maintenant, redémarre OpenDKIM et vérifie que le redémarrage s'est bien passé.

```
systemctl restart opendkim
systemctl status -l opendkim
```

A présent, nous allons configurer Postfix pour qu'il utilise OpenDKIM pour signer les emails envoyés.

Commence par créer le répertoire pour la *socket* d'OpenDKIm dans l'espace de travail de Postfix.

```
mkdir /var/spool/postfix/opendkim
chown opendkim:postfix /var/spool/postfix/opendkim
```

Ajoute l'utilisateur `postfix` dans le groupe `opendkim`...

```
adduser postfix opendkim
```

... et définis la bonne *socket* pour Postfix dans le fichier `/etc/default/opendkim`.

```
# Command-line options specified here will override the contents
# of /etc/opendkim.conf. See opendkim(8) for a complete list of
# options.
#DAEMON_OPTS=""
#
# Uncomment to specify an alternate socket
# Note that setting this will override any Socket value in
# opendkim.conf
SOCKET="local:/var/spool/postfix/opendkim/opendkim.sock"
#SOCKET="inet:54321" # listen on all interfaces on port 54321
#SOCKET="inet:12345@localhost" # listen on loopback on port
    12345
#SOCKET="inet:12345@192.0.2.1" # listen on 192.0.2.1 on port
    12345
```

Edite le fichier de configuration `/etc/postfix/main.cf` et ajoute une section pour activer la signature des emails par le démon d'OpenDKIM.

```
# Milter configuration
# OpenDKIM
milter_default_action = accept
# Postfix >= 2.6 milter_protocol = 6, Postfix <= 2.5
    milter_protocol = 2
milter_protocol = 6
smtpd_milters = local:opendkim/opendkim.sock
non_smtpd_milters = local:opendkim/opendkim.sock
```

L'endroit où tu places cette section n'a pas d'importance. Habituellement, elle est placée après l'entrée `smtpd_recipient_restrictions`. Tu peux remarquer que le chemin de la *socket* n'est pas le même que celui présent dans `/etc/defaults/opendkim`. C'est parce que le service Postfix est dans un *container chroot*, le chemin est donc relatif au répertoire racine de cet environnement isolé.

Redémarre le démon OpenDKIM, puis redémarre Postfix.

```
systemctl restart opendkim
systemctl restart postfix
```

Tous les emails envoyés par Postfix sont maintenant signés. Il ne reste plus qu'à renseigner la clé publique via une entrée DNS.

Tout d'abord, ouvre le fichier `/etc/opendkim/keys/telecom-etude.com.txt`.

```
201908._domainkey  IN  TXT ( "**v=DKIM1; h=rsa-sha256; k=rsa; s=
    email; "
    "p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAu5oIUrFDWZK"
    "7F4thFxpZa2or6jBEX3cSL6b2TJdPkO5iNn9vHNXhNX31nOefN8FksX94"
    "YbLJ8NHcFPbaZTW8R2HthYxRaCyqodxlLHibg8aHdfa+bxKeiI/xABRuA"
    "M0WG0JEDSyakMFqIO40ghj/h7DUc/4OXNdeQhrKDTlgf2bd+FjpJ3bNAF"
    "cMYa3Oeju33b2Tp+PdtqIwXRZksfuXh7m30kuyavp3Uaso145DRBaJZA5"
```

```
    "5lNxmHWMgMjO+YjNeuR6j4oQqyGwzPaVcSdOG8Js2mXt+J3Hr+nNmJGxZ"
    "UUW4Uw5ws08wT9opRgSpn+ThX2d1AgQePpGrWOamC3PdcwIDAQAB**");
    ----- DKIM key 201908 for telecom-etude.com
```

Sélectionne et copie tout le contenu des guillemets (sans les étoiles *) dans un fichier à part, et remplace `h=rsa-sha256` par `h=sha256`.

```
v=DKIM1; h=sha256; k=rsa; s=email; p=
  MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAu5oIUrFDWZK7F4thF
  xpZa2or6jBEX3cSL6b2TJdPkO5iNn9vHNXhNX31nOefN8FksX94YbLJ8NHcFP
  baZTW8R2HthYxRaCyqodxlLHibg8aHdfa+bxKeiI/xABRuAM0WG0JEDSyakMF
  qIO40ghj/h7DUc/4OXNdeQhrKDTlgf2bd+FjpJ3bNAFcMYa3Oeju33b2Tp+Pd
  tqIwXRZksfuXh7m30kuyavp3Uaso145DRBaJZA55lNxmHWMgMjO+YjNeuR6j4
  oQqyGwzPaVcSdOG8Js2mXt+J3Hr+nNmJGxZUUW4Uw5ws08wT9opRgSpn+ThX2
  d1AgQePpGrWOamC3PdcwIDAQAB
```

Copie cette valeur dans une entrée TXT du serveur DNS avec l'hôte `201908._domainkey` (certains serveurs DNS proposent de renseigner directement une entrée DKIM). *In fine*, la nouvelle entrée DNS est de la forme

```
201908._domainkey IN TXT ( "v=DKIM1; k=rsa; s=email; h=sha256;
   p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAu5oIUrFDWZK7F4
   thFxpZa2or6jBEX3cSL6b2TJdPkO5iNn9vHNXhNX31nOefN8FksX94YbLJ8N
   HcFPbaZTW8R2HthYxRaCyqodxlLHibg8aHdfa+bxKeiI/xABRuAM0WG0JEDS
   yakMFqIO40ghj/h7DUc/4OXNdeQhrKDTlgf2bd+FjpJ3bNAFcMYa3Oeju33b
   2Tp+PdtqIwXRZksfuXh7m30kuyavp3Uaso145DRBaJZA55lNxmHWMgMjO+Yj
   NeuR6j4oQqyGwzPaVc/SdOG8Js2mXt+J3Hr+nNmJGxZUUW4Uw5ws08wT9opR
   gSpn+ThX2d1AgQePpGrWOamC3PdcwIDAQAB" )
```

Répète l'opération pour chaque clé publique que tu as.

Si tu veux vérifier que la clé publique est bien disponible, utilise la commande suivante :

```
opendkim-testkey -d telecom-etude.com -s 201908 -vvv
```

Si tout va bien, le dernier message est « Key OK ». Juste avant, il est possible que tu voies un message « Key not secure ». Ceci est normal et ne signale pas une erreur. C'est juste que le domaine n'est pas configuré pour DNSSEC.

2.9.3 Author Domain Signing Practices (ADSP)

En l'état actuel de ton serveur, tes emails sont signés grâce à DKIM. Mais si un serveur distant recevait un mail avec ton nom de domaine mais sans signature DKIM, il ne saurait pas si c'est juste que ton serveur ne signe pas les mails ou si c'est un faux email. Pour corriger cela, il suffit de renseigner dans le serveur DNS que tous les emails qui sortiront du domaine seront signés. Ajoute l'entrée TXT suivante dans la zone DNS :

```
_adsp._domainkey IN TXT "dkim=all"
```

2.9.4 Domain Message Authentication, Reporting & Conformance (DMARC)

Une entrée DMARC dans le serveur DNS permet d'indiquer aux différents serveurs mail du monde ce que nous leur conseillons de faire avec les emails de notre domaine qui échouent face aux tests SPF et/ou DKIM.

Un enregistrement DMARC est une entrée TXT avec l'hôte `_dmarc`. Il y a de nombreuses valeurs possibles. Je vais en présenter deux.

Supposons que l'entrée DMARC ait pour valeur

```
v=DMARC1;p=quarantine;sp=quarantine;adkim=r;aspf=r
```

Cela signifierait que, si l'email échoue à SPF ou DKIM, il faut l'écarter des messages normaux. De plus, le serveur mail destinataire n'a pas besoin de signaler le problème au gestionnaire du domaine. Peu de serveurs demandent à être alertés s'il y a des emails qui ne passent pas les tests. Si tu veux mettre en place ce procédé, voici la valeur qu'aurait l'entrée DMARC :

```
v=DMARC1;p=quarantine;sp=quarantine;adkim=r;aspf=r;fo=1;rf=afrf;
    rua=mailto:user@example.com
```

Les serveurs mail signaleraient tous les emails suspects à `user@example.com`. N'hésite pas à chercher sur le Net les différentes possibilités qu'offrent les entrées DMARC.

Dans le cadre de la Junior-Entreprise, il ne devrait pas y avoir beaucoup d'emails suspects. Alors, autant être averti s'il y a en. Il te suffit donc de créer une entrée TXT dont l'hôte est `_dmarc` et dont la valeur est

```
v=DMARC1;p=quarantine;sp=quarantine;adkim=r;aspf=r;fo=1;rf=afrf;
    rua=mailto:admin@telecom-etude.com
```

2.9.5 Tests de la configuration

La configuration avancée est assez complexe à mettre en place. Il vaut mieux réaliser des tests de l'installation. Je te propose deux outils :

— Envoie un mail à `check-auth@verifier.port25.com` en utilisant une adresse de ton domaine. Tu recevras un rapport détaillé sur la configuration de ton serveur mail en réponse.
— Va sur le site `https://www.mail-tester.com`, prends l'email généré et envoie un email à cette adresse. Après l'envoi, tu pourras demander un rapport détaillé sur le site.

2.9.6 DKIM Key rotation

La raison pour laquelle le format `YYYMM` est le plus adapté pour le sélecteur DKIM est qu'il faudrait changer les paires de clés publiques-privées régulièrement. Il est recommandé de le faire une fois par mois. En tout cas, il vaut mieux

éviter de garder une paire plus de 6 mois. Pour le faire sans perturber les emails en transit, il faut créer les nouvelles clés en utilisant un nouveau sélecteur.

Voici la méthodologie :

1. Génère de nouvelles clés avec la date actuelle comme sélecteur, pas directement dans `/etc/opendkim/keys`
2. Rajoute le nouvel enregistrement DKIM dans le serveur DNS sans retirer l'ancien. Teste la nouvelle entrée avec `opendkim-testkey`. Il faut que le test fonctionne avant de continuer.

```
opendkim-testkey -d example.com -s YYYYMM -k example.
    private
```

3. Arrête Postfix et OpenDKIM avec `systemctl stop postfix opendkim` afin qu'ils ne traitent pas d'emails pendant le changement de clés.
4. Mets les clés dans `/etc/opendkim/keys` et configure correctement le propriétaire et les permissions.
5. Edite `/etc/opendkim/key.table` et change le vieux sélecteur `YYYYMM` par le nouveau.
6. Redémarre OpenDKIM et Postfix par

```
systemctl start opendkim
systemctl start postfix
```

   Vérifie que les deux services ont démarré sans problème.

7. Après une quinzaine de jours, tous les emails en transit devraient avoir été délivrés, et donc l'ancien enregistrement DKIM n'est plus nécessaire. Tu peux le supprimer. Ne t'inquiète pas si tu oublies de retirer l'ancienne clé, il n'y a pas de problème de sécurité. C'est juste pour avoir la zone DNS la plus propre possible.

Onzième partie
Données des utilisateurs

Normalement, lorsque tu as restauré l'annuaire LDAP, le système a récupéré de nombreux utilisateurs qui peuvent se connecter au serveur. Mais, pour le moment, ils n'ont pas de répertoires personnels. Pour un utilisateur dont le login est `username`, le répertoire personnel est accessible au chemin `/home/jetmenXX/username` avec `XX` pour la promo. Par exemple, mon login était `tfeneuil` et je faisais partie des Jetmen 2020, donc mon répertoire personnel était `/home/jetmen20/tfeneuil`.

1 Restituer les répertoires personnels

Supposons que les répertoires personnels soient sur l'ordinateur `192.168.1.250` dans le réseau interne de la Junior-Entreprise, sur le compte de `backup-server` au chemin `/path-to-backup/`.

Il faut d'abord les récupérer.

```
scp -rp backup-server@192.168.1.250:/path-to-backup/home/
    jetmenXX /home/
```

Malheureusement, à ce stade, tous les répertoires personnels appartiennent à `root`. Il faut donc rétablir les permissions.

```
chown -R je:jetmen /home/jetmenXX
```

Pour rétablir les permissions à des répertoires personnels à chacun de ses propriétaires, tu pourrais appliquer la commande précédente à chacun d'entre eux, mais ce serait fastidieux. A la place, tu vas donc créer un script *Bash* qui va rétablir automatiquement les permissions !

```
cd /home/
vim restore-owners
```

```
folders=`find $1 -maxdepth 1 -type d`
for folder in $folders; do
        user=`basename $folder`
        chown -R $user:jetmen $folder 2> /dev/null
done
```

Il ne te reste plus qu'à exécuter ce script.

```
bash restore-owners jetmenXX
```

Et c'est terminé pour la promo des Jetmen 20XX ! Il ne te reste plus qu'à refaire tout cela pour chaque promo que tu veux restaurer...

Douzième partie
Serveur Samba

Ressources :
— *Mettre en place un serveur Samba*, OpenClassrooms,
https://openclassrooms.com/courses/mettre-en-place-un-serveur-samba

1 Installation

Samba est le paquet correspondant au serveur Samba. *samba-common-bin* contient des utilitaires pour le serveur et les clients.

```
apt-get install samba
apt-get install samba-common-bin
```

2 Configuration du serveur

Commence par aller dans le dossier de configuration de Samba (`/etc/samba/`).

```
cd /etc/samba/
```

Le fichier de configuration est `/etc/samba/smb.conf`. Fais une sauvegarde de la version de base de ce fichier (cela peut toujours être utile dans le futur).

```
cp smb.conf smb.conf.old
```

Puis édite-le (crée-le) de manière à ce qu'il ait le contenu suivant :

```
[global]
        workgroup = JET
        server string = %L (Samba %v)
        passdb backend = ldapsam
        passwd program = /usr/bin/passwd %u
        passwd chat = *Enter\snew\sUNIX\spassword:* %n\n *Retype
            \snew\sUNIX\spassword:* %n\n .
        passwd chat debug = Yes
        username map = /etc/samba/smbusers
        unix password sync = Yes
        security = user
        log level = 3
        log file = /var/log/samba/log.%m
        name resolve order = hosts bcast
        time server = Yes
        printcap name = /etc/printcap
        add user script = /usr/sbin/ldapadduser %u jetmen
```

```
        delete user script = /usr/sbin/ldapdeleteuser %u
        add group script = /usr/sbin/ldapaddgroup %g
        delete group script = /usr/sbin/ldapdeletegroup %g
        add user to group script = /usr/sbin/ldapaddusertogroup
            %u %g
        add machine script = /usr/sbin/ldapaddmachine %u 100
        logon script = logon.cmd
        logon path = \\je\profile\%U
        logon drive = X:
        domain logons = Yes
        preferred master = Yes
        domain master = Yes
        use client driver = Yes

        ldap admin dn = cn=admin,dc=telecom-etude,dc=com
        ldap user suffix = ou=People
        ldap group suffix = ou=Group
        ldap idmap suffix = ou=Idmap
        ldap machine suffix = ou=People
        ldap passwd sync = yes
        ldap suffix = dc=telecom-etude,dc=com
        ldap ssl = no

        winbind enum users = Yes
        winbind enum groups = Yes
        wins support = yes

        map to guest = Never
        guest account = nobody
        guest ok = no

[homes]
        comment = Home
        valid users = %S
        read only = No
        browseable = No

[netlogon]
        comment = Network Logon Service
        path = /var/lib/samba/netlogon
        valid users = %U
        admin users = @poleinfo
        read only = No
        create mask = 0664
        directory mask = 0775

[profile]
        comment = User profiles
        path = /home/%u/NTProfile
        valid users = %U
```

```
        read only = No
        create mask = 0600
        directory mask = 0700
        browseable = No

[public]
        comment = Public
        path = /home/public
        valid users = @jetmen, Admin
        force group = jetmen
        read only = No
        create mask = 0664
        directory mask = 0775
```

Plusieurs remarques sur la configuration Samba :

— la configuration est divisée en sections. À chaque ligne avec le format

```
[Du texte...]
```

commence une nouvelle section, c'est-à-dire, commence la définition d'un nouveau partage. Tu peux constater la présence d'une section obligatoire « global » qui définit tous les paramètres généraux de Samba : le nom du groupe, les paramètres de stockage des entités dans LDAP, la journalisation, . . . Dans la section « profile », sont définis tous les paramètres liés aux profils itinérants des utilisateurs (quand ils se connectent sur un ordinateur du local). La section « public », qui permet de connecter un lecteur réseau, correspond au dossier *Public* du serveur ;

— dans le cas où tu veux modifier le fichier de configuration, fais attention si tu rajoutes le paramètre *netbios name*. Pour que les profils itinérants puissent fonctionner, il faut que le paramètre *netbios name* contienne l'*hostname* du serveur. Puisque, par défaut, Samba prend juste cette valeur, j'ai jugé qu'il était inutile de l'insérer dans la configuration (au contraire, cela peut inciter à modifier ce paramètre, faisant ainsi bogger les partages) ;

— l'imprimante de Telecom Etude n'est pas gérée par le serveur. Elle est branchée sur le réseau interne de Telecom Etude. Si elle était branchée au serveur (via un câble USB par exemple) **sans** être branchée sur le réseau, on aurait pu la partager à l'aide de Samba pour permettre aux ordinateurs du local de l'utiliser. Mais, dans notre cas, pour pouvoir l'utiliser sur ces ordinateurs, il suffit d'installer l'imprimante en cherchant celles installées sur le réseau. Cette configuration permet en plus aux jetmen qui utilisent le Wifi « Telecom Etude » d'imprimer.

A présent, donne le mot de passe de LDAP (*ie* le *mot de passe usuel*) à Samba pour qu'il puisse se connecter à l'annuaire :

```
smbpasswd -W
```

Pour finir, il ne te reste plus qu'à redémarrer Samba.

```
/etc/init.d/samba restart
```

Samba utilise certains ports pour communiquer avec les différentes machines connectées au serveur. Les ports qu'il utilise sont :

— 135 en TCP et UDP ;
— 137 en TCP et UDP ;
— 138 en UDP ;
— 139 en TCP ;
— 445 en TCP et UDP.

Par défaut, il n'y a pas de pare-feu sur Linux. Cependant, si tu veux en configurer un avec `iptables`, il te faudra ajouter certaines règles pour que Samba fonctionne correctement.

3 Préparation des ordinateurs du local qui travaille sous Windows

Les ordinateurs sous Windows vont fonctionner dans le domaine indiqué dans le paramètre *workgroup* (à vérifier, sinon, c'est via le LDAP). Pour cela, ils vont devoir trouver un contrôleur de domaine *Active Directory*, et Samba en intègre un (depuis la version 4) : il s'agit du service *smbd*.

Toutes ces consignes ont été testées sur des *Windows 7*. Pour les autres versions de Windows, il te faudra sûrement adapter un peu les consignes. *Dans la suite, je supposerait qu'on travaille sur un ordinateur avec Windows 7 où une session admin a déjà été créée (avec le mot de passe usuel).*

3.1 Ajouter l'ordinateur dans LDAP

Il faut qu'il y ait une entrée correspondant à la machine dans le LDAP pour que cela fonctionne (au niveau de `ou=Hosts,dc=telecom-etude,dc=com`). L'entrée doit avoir une forme semblable à

```
dn: uid=JE1-1$,ou=Hosts,dc=telecom-etude,dc=com
objectClass: account
objectClass: sambaSamAccount
objectClass: top
sambaAcctFlags: [W          ]
sambaSID: S-1-5-21-1668423846-3243855295-230216567-1239
uid: JE1-1$
structuralObjectClass: account
sambaNTPassword: 079456F7B0FA6D99F9A37709E503CF67
sambaPwdLastSet: 1524588919
```

3.2 Détecter le contrôleur de domaine Active Directory

Pour que l'ordinateur puisse détecter le contrôleur de domaine *Active Directory* mis en place par Samba dans le serveur, il faut un peu toucher aux registres. Pour y accéder, appuie sur *Windows*+R, renseigne « regedit » et valide. Normalement, le registre s'ouvre (avec une demande d'élévation de permissions si tu n'es pas dans la session *admin*).

Va dans

```
HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\
    LanmanWorkstation\\Paramaters
```

et modifie les registres de la manière suivante :

```
DomainCompatibilityMode = DWORD: 1
DNSNameResolutionRequired = DWORD: 0
```

Si tu ne vois pas le serveur dans « Réseau » de l'explorateur Windows (après avoir activé la découverte du réseau), modifie le champ « Type de nœud » qui apparaît avec la commande

```
ipconfig/all
```

dans une console. Pour modifier le type de nœud, va alors à nouveau dans les registres, puis dans le dossier

```
HKEY_LOCAL_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\NetBT\\
    Paramaters
```

et modifie les données de la manière suivante :

```
DhcpNodeType = REG_DWORD: 8
```

3.3 Configurer l'ordinateur pour utiliser le domaine

Il suffit d'aller dans « Panneau de configuration / Système / Modifier les paramètres / Nom de l'ordinateur / Modifier le domaine ». Il ne te reste plus qu'à renseigner le nom du domaine (*JET* si tu as suivi à la lettre ce guide). En validant, des identifiants vont t'être demandés pour l'opération de jonction dans le domaine : renseigne *root* et le *mot de passe usuel*.

3.4 Personnalisation avancée de l'ordinateur

— Pour modifier le fond d'écran de verrouillage, il te suffit d'aller dans les registres dans le dossier

```
HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows\\
    CurrentVersion\\Authentification\\LogonUI\\Background
```

et modifier les données de la manière suivante :

```
OEMBackground = REG_DWORD: 1
```

Il ne te reste plus qu'à ajouter l'image JPG de ton fond d'écran à l'emplacement (et nom) suivant :

```
C:\\Windows\\System32\\oobe\\info\\backgrounds\\
    backgroundDefault.jpg
```

— Tu peux également faire apparaître un message d'information qui s'affichera lorsqu'un utilisateur tente de se connecter sur l'ordinateur. Pour cela, va dans le registre

```
HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows\\
    CurrentVersion\\Policies\\System
```

mets le titre de ton message dans l'item *legalnoticecaption (REG_SZ)* et le contenu dans *legalnoticetext (REG_SZ)*.

Treizième partie
Les Wikis

1 Préliminaire

Pour pouvoir installer les différents Wikis, il faut d'abord que tu installes le serveur Web si ce n'est pas déjà fait. Ensuite, si ce n'est pas déjà fait, il faut que tu actives les modules `ssl`, `rewrite` et `headers` dans *Apache2*.

```
a2enmod ssl rewrite headers
/etc/init.d/apache2 reload
```

De plus, il faut que tu installes le paquet `php-ldap`.

```
apt-get install php-ldap
/etc/init.d/apache2 reload
```

2 Installation d'un Wiki vide

Tout d'abord, télécharge le Wiki vierge sur le site officiel de *DokuWiki*. Sauf changement, tu devrais obtenir une archive sous le format TGZ. Supposons dans la suite qu'elle soit nommée `dokuwiki.tgz`. Ensuite, il faut que tu l'envoies sur le serveur. Tu peux, pour cela, utiliser par exemple `scp` (pour que tu puisses le faire, il faut que tu sois sur un système Unix et que ton ordinateur soit dans un réseau qui connaît `je.enst.fr`).

```
scp dokuwiki.tgz mylogin@je.enst.fr:~/
```

Et maintenant, sur le serveur, extrais l'archive TGZ, place-la avec tous les sites (`/var/www/`) et donne-lui les bonnes permissions.

```
tar xzvf dokuwiki.tgz
cp -r dokuwiki /var/www/
chown -R www-data:www-data /var/www/dokuwiki
```

— Pour le Wiki SOS, le Wiki du Pôle Informatique :

```
mv /var/www/dokuwiki /var/www/sos
```

— Pour le Wiki TE, le Wiki de la gestion de l'association :

```
mv /var/www/dokuwiki /var/www/wiki
```

3 Configuration Apache2 d'un Wiki

3.1 Wiki SOS, le Wiki du Pôle Informatique

Crée un certificat pour le Wiki :

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d sos.telecom-etude.com -d sos.telecom-etude.
    fr
/etc/init.d/apache2 start
```

Puis, crée un hôte virtuel sur *Apache2* pour accéder au Wiki. Pour cela, crée un fichier `/etc/apache2/sites-available/sos.conf` :

```
vim /etc/apache2/sites-available/sos.conf
```

avec le contenu suivant :

```
<VirtualHost *:80>
    ServerAdmin admin@telecom-etude.com
    ServerName sos.telecom-etude.com
    ServerAlias sos.telecom-etude.fr

    RewriteEngine On
    RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
</VirtualHost>

<VirtualHost *:443>
    ServerAdmin admin@telecom-etude.com
    Header always set Strict-Transport-Security "max-age
        =31536000";
    ServerName sos.telecom-etude.com
    ServerAlias sos.telecom-etude.fr
    DocumentRoot /var/www/sos/

    ErrorLog /var/log/apache2/sos.log

    # Activation de la connexion securisee SSL
    Include /etc/letsencrypt/options-ssl-apache.conf
    SSLCertificateFile    /etc/letsencrypt/live/sos.telecom-
        etude.com/fullchain.pem
    SSLCertificateKeyFile  /etc/letsencrypt/live/sos.telecom-
        etude.com/privkey.pem

    ServerSignature On

    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
```

```
    <Directory /var/www/sos/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride all
        Order allow,deny
        allow from all
    </Directory>

</VirtualHost>
```

Rends opérationnel le nouvel hôte virtuel et recharge *Apache2*.

```
a2ensite sos
/etc/init.d/apache2 reload
```

A présent, le Wiki devrait être disponible à l'URL https://sos.telecom-etude.com... mais il n'est pas encore configuré.

3.2 Wiki TE, le Wiki de la gestion de l'association

Crée un certificat pour le Wiki :

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d wiki.telecom-etude.com -d wiki.telecom-
    etude.fr
/etc/init.d/apache2 start
```

Puis, crée un hôte virtuel sur *Apache2* pour accéder au Wiki. Pour cela, crée un fichier `/etc/apache2/sites-available/wiki.conf` :

```
vim /etc/apache2/sites-available/wiki.conf
```

avec le contenu suivant :

```
<VirtualHost *:80>
    ServerAdmin admin@telecom-etude.com
    ServerName wiki.telecom-etude.com
    ServerAlias wiki.telecom-etude.fr

    RewriteEngine On
    RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
</VirtualHost>

<VirtualHost *:443>
    ServerAdmin admin@telecom-etude.com
    Header always set Strict-Transport-Security "max-age
        =31536000";
    ServerName wiki.telecom-etude.com
    ServerAlias wiki.telecom-etude.fr
    DocumentRoot /var/www/wiki/
```

```
    ErrorLog /var/log/apache2/wiki.log

    # Activation de la connexion securisee SSL
    Include /etc/letsencrypt/options-ssl-apache.conf
    SSLCertificateFile    /etc/letsencrypt/live/wiki.telecom-
        etude.com/fullchain.pem
    SSLCertificateKeyFile  /etc/letsencrypt/live/wiki.telecom-
        etude.com/privkey.pem

    ServerSignature On

    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/wiki/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride all
        Order allow,deny
        allow from all
    </Directory>

</VirtualHost>
```

Rends opérationnel le nouvel hôte virtuel et recharge *Apache2*.

```
a2ensite wiki
/etc/init.d/apache2 reload
```

A présent, le Wiki devrait être disponible à l'URL https://wiki.telecom-etude.com... mais il n'est pas encore configuré.

4 Configuration complète d'un Wiki

4.1 Wiki SOS, le Wiki du Pôle Informatique

Va donc à l'URL https://sos.telecom-etude.com/install.php et rentre les informations suivantes :

Wiki name	Pôle Informatique
Enable ACL	Yes
Superuser	admin
Real name	Admin
Email	admin@telecom-etude.com
Password	*Mot de passe usuel*
Initial ACL policy	Closed Wiki
Allow users to register themselves	No
Licence	Do not show any licence information
Send anonymous usage data	No

Tu peux à présent te connecter au Wiki (`https://sos.telecom-etude.com/`) avec le compte administrateur que tu viens de configurer.

Va dans le panneau « Admin », clique sur « Configuration Settings », sélectionne « fr » pour le paramètre « Interface language » et change la description du Wiki en « Ce Wiki est uniquement dédié au Pôle Informatique de Telecom Etude. Il permet de centraliser tout le savoir des différentes générations du Pôle. ». Enregistre et recharge la page pour que le Wiki soit en français.

Nous allons installer toutes les extensions nécessaires au fonctionnement du Wiki. Pour cela, va dans le panneau « Administrer », clique sur « Gestionnaire d'extensions » et installe toutes les extensions suivantes si elles ne le sont pas déjà (utilise l'onglet « Rechercher et installer ») :

— *ACL Manager* ;
— *LDAP Auth Plugin* ;
— *Configuration Manager* ;
— *CSV Plugin* ;
— *Extension Manager* ;
— *folded plugin* ;
— *include plugin* ;
— *Info Plugin* ;
— *Move plugin* ;
— *Revert Manager* ;
— *safefnrecode plugin* ;
— *styling plugin* ;
— *User Manager* ;
— *Wrap Plugin*.

Puis active toutes ces extensions (la plupart le sont déjà...) et désactive toutes celles qui ne sont pas dans la liste et qui sont déjà installées !

Maintenant que tu as installé toutes les extensions indispensables, tu as en particulier installé *LDAP Auth Plugin* qui permet de se connecter au LDAP avec des comptes présents dans un annuaire LDAP. Tu peux donc aller dans le gestionnaire de configuration de l'interface d'administration pour la configurer.

Dans la section « Authldap », renseigne les informations suivantes :

Serveur LDAP	127.0.0.1
Port du serveur LDAP	389
Comptes utilisateur	`ou=People,dc=telecom-etude,dc=com`
Groupes d'utilisateurs	`ou=Group,dc=telecom-etude,dc=com`
Email	`admin@telecom-etude.com`
Filtre LDAP pour les comptes utilisateur	`(\&(uid=\%\{user\})(objectClass=inetOrgPerson))`
Filtre LDAP pour les groupes	`(\&(objectClass=posixGroup)(|(gidNumber=\%\{gid\})(memberUID=\%\{user\})))`
Version du protocole	3

Avant de rendre l'authentification LDAP opérationnelle, il te faut renseigner les utilisateurs qui auront le droit d'accéder au Wiki. Pour cela, rends-toi dans la

« Gestion de la liste des contrôles d'accès (ACL) » dans l'interface d'administration et définis les contrôles d'accès suivants :

Page	Utilisateur/Groupe	Autorisations
*	@ALL	Aucune
*	@poleinfo	Effacer
wiki :*	@ALL	Lecture

Il ne te reste plus qu'à activer l'authentification LDAP. Pour cela, dans les paramètres de configuration, va dans la section « Paramètres d'authentification » et change

Mécanisme d'authentification	authldap
Super-utilisateur	@poleinfo

Puis maintenant, il faudrait songer à importer les données d'une sauvegarde du Wiki SOS. Pour cela, il te suffit juste de remplacer le répertoire `sos/data` par celui de la sauvegarde. Et une fois cela fait, il faut rétablir le bon propriétaire.

```
chown -R www-data:www-data /var/www/sos/data
```

Simple suggestion : va dans les paramètres de style du thème dans l'interface d'administration et change la valeur de l'attribut « Largeur du site complet » de 75em en 85em.

4.2 Wiki TE, le Wiki de la gestion de l'association

Va donc à l'URL `https://wiki.telecom-etude.com/install.php` et rentre les informations suivantes :

Wiki name	 Gestion quotidienne
Enable ACL	Yes
Superuser	admin
Real name	Admin
Email	admin@telecom-etude.com
Password	*Mot de passe usuel*
Initial ACL policy	Closed Wiki
Allow users to register themselves	No
Licence	Do not show any licence information
Send anonymous usage data	No

Tu peux à présent te connecter au Wiki (`https://wiki.telecom-etude.com/`) avec le compte administrateur que tu viens de configurer.

Va dans le panneau « Admin », clique sur « Configuration Settings », sélectionne « fr » pour le paramètre « Interface language » et change la description du Wiki en « Le Wiki de Telecom Etude regroupant les précieuses informations de l'association. ». Enregistre et recharge la page pour que le Wiki soit en français.

Nous allons installer toutes les extensions nécessaires au fonctionnement du Wiki. Pour cela, va dans le panneau « Administrer », clique sur « Gestionnaire d'extensions » et installe toutes les extensions suivantes si elles ne le sont pas déjà (utilise l'onglet « Rechercher et installer »).

— *ACL Manager* ;
— *LDAP Auth Plugin* ;
— *Configuration Manager* ;
— *CSV Plugin* ;
— *Extension Manager* ;
— *folded plugin* ;
— *include plugin* ;
— *Info Plugin* ;
— *Move plugin* ;
— *Revert Manager* ;
— *safefnrecode plugin* ;
— *styling plugin* ;
— *User Manager* ;
— *Wrap Plugin*.

Puis active toutes ces extensions (la plupart le sont déjà...) et désactive toutes celles qui ne sont pas dans la liste et qui sont déjà installées !

Maintenant que tu as installé toutes les extensions indispensables, tu as en particulier installé *LDAP Auth Plugin* qui permet de se connecter au LDAP avec des comptes présents dans un annuaire LDAP. Tu peux donc aller dans le gestionnaire de configuration de l'interface d'administration pour la configurer.

Dans la section « Authldap », renseigne les informations suivantes :

Serveur LDAP	127.0.0.1
Port du serveur LDAP	389
Comptes utilisateur	`ou=People,dc=telecom-etude,dc=com`
Groupes d'utilisateurs	`ou=Group,dc=telecom-etude,dc=com`
Email	`admin@telecom-etude.com`
Filtre LDAP pour les comptes utilisateur	`(\&(uid=\%\{user\})(objectClass=inetOrgPerson))`
Filtre LDAP pour les groupes	`(\&(objectClass=posixGroup)(|(gidNumber=\%\{gid\})(memberUID=\%\{user\})))`
Version du protocole	3

Avant de rendre l'authentification LDAP opérationnelle, il te faut renseigner les utilisateurs qui auront le droit d'accéder au Wiki. Pour cela, rends-toi dans la « Gestion de la liste des contrôles d'accès (ACL) » dans l'interface d'administration et définis les contrôles d'accès suivants :

Page	**Utilisateur/Groupe**	**Autorisations**
*	@ALL	Aucune
*	@poleinfo	Effacer
*	@jetmen	Envoyer
wiki :*	@ALL	Lecture
guides :redaction_du_cc	@ALL	Lecture

Il ne te reste plus qu'à activer l'authentification LDAP. Pour cela, dans les paramètres de configuration, va dans la section « Paramètres d'authentification » et change

Mécanisme d'authentification	authldap
Super-utilisateur	@poleinfo

Puis maintenant, il faudrait songer à importer les données d'une sauvegarde du Wiki TE. Pour cela, il te suffit juste de remplacer le répertoire `sos/data` par celui de la sauvegarde. Et une fois cela fait, il faut rétablir le bon propriétaire.

```
chown -R www-data:www-data /var/www/sos/data
```

Simple suggestion : va dans les paramètres de style du thème dans l'interface d'administration et change la valeur de l'attribut « Largeur du site complet » de 75em en 85em.

Quatorzième partie

Intranet V2020

1 Installation des dépendances

1.1 Installation de l'environnement de l'intranet

Commence par installer *Python3* ainsi que l'outil pour créer des environnements virtuels.

```
apt-get install python3 python3-pip
pip3 install virtualenv
```

Puis crée le dossier de l'intranet...

```
mkdir /var/www/intranet
cd /var/www/intranet
```

... et initialise son environnement virtuel.

```
virtualenv -p python3 venv
```

1.2 Installation du module WSGI

L'Intranet est fait en Python avec le *framework* et nous utilisons *Apache2* comme serveur Web. Pour faire le lien, il faut installer le module WSGI. Le piège est de l'installer directement avec `apt-get`. Si tu fais cela, tu vas installer le module pour du Python2 et... cela va planter (#vécu) !

> Attention, il semblerait qu'installer le paquet `libapache2-mod-wsgi-py3` suffise pour répondre au problème. On ne m'a appris cela qu'après coup. Si tu utilises cette méthode (conseillée *a priori*), n'oublie pas quand même d'activer le module `wsgi` sur *Apache2* (*cf* la fin de cette partie).

Il faut donc compiler soi-même le module WSGI avec la bonne version de Python.

```
apt-get install apache2-dev
cd /var/www/intranet/
source venv/bin/activate
pip install mod_wsgi
deactivate
```

Normalement, le module WSGI est installé. Voici un test pour vérifier l'installation :

```
sudo -u www-data venv/bin/mod_wsgi-express start-server
```

Cette commande publie sur l'URL `http://localhost:8000` une page relativement simple.

Il te faut maintenant récupérer le module WSGI pour le « donner » à *Apache2*. Pour savoir où il se trouve, lance la commande suivante :

```
sudo -u www-data venv/bin/mod_wsgi-express module-config
```

Tu devrais y trouver l'emplacement du module au niveau du paramètre "LoadModule". Dans mon cas, il s'agit de

```
LoadModule wsgi_module "/var/www/intranet/venv/lib/python3.5/
    site-packages/mod_wsgi/server/mod_wsgi-py35.cpython-35m-
    x86_64-linux-gnu.so"
...
```

Il faut que tu le positionnes à l'emplacement indiqué dans le fichier `/etc/apache2/mod-availables/wsgi.load`.

```
#/etc/apache2/mod-availables/wsgi.load
LoadModule wsgi_module "/usr/lib/apache2/modules/mod_wsgi.so"
```

Dans mon cas, comme tu peux le voir, il s'agit de `/usr/lib/apache2/modules/mod_wsgi.so`. Donc tape simplement

```
cd /usr/lib/apache2/modules/
cp /var/www/intranet/venv/lib/python3.5/site-packages/mod_wsgi/
    server/mod_wsgi-py35.cpython-35m-x86_64-linux-gnu.so
    mod_wsgi.so
chmod 644 wsgi.so
```

Il ne te reste plus qu'à activer le module :

```
a2enmod wsgi
/etc/init.d/apache2 reload
```

2 Installation de l'intranet

2.1 Installation du code

Supposons que le code est sur un dépôt Git accessible à l'URL `https://git.telecom-etude.fr/pole-info/Intranet`. Télécharge-le et déplace-le dans le dossier `/var/www/intranet`.

```
git clone https://git.telecom-etude.fr/pole-info/Intranet
mv IntranetTE/* /var/www/intranet/
rm -r IntranetTE
chown -R www-data:www-data /var/www/intranet
```

Par la suite, toutes les commandes pour faire des opérations où il y a potentiellement des problèmes de droits devront être précédées par

```
sudo -u www-data ...
```

2.2 Installation des dépendances de l'intranet

Tape

```
source /var/www/intranet/venv/bin/activate
pip install -r requirements.txt
deactivate
```

S'il y a un problème pour installer `python-ldap`, installe les paquets suivants :

```
apt-get install libsasl2-dev python-dev libldap2-dev libssl-dev
```

2.3 Ajout d'une base de données MySQL

Dans la suite, on supposera que tu as suivi le protocole d'installation du serveur MySQL et de PHPmyAdmin.

2.3.1 Ajout d'un utilisateur MySQL

Connecte-toi en tant qu'administrateur (`admin@localhost`) sur PHPmyAdmin (`https://telecom-etude.com/phpmyadmin`).

- Crée une nouvelle base de données
  - Nom : intranet
  - Encodage : utf8mb4_general_ci
- Crée un nouvel utilisateur
  - Mot d'utilisateur : intranet
  - Mot de passe : k******3
  - Et laisse les valeurs par défaut pour le reste.
- Edite les privilèges de l'utilisateur `intranet`, sélectionne la base de données `intranet` et donne tous les droits possibles dessus.

2.3.2 Installation de MySQL pour Python

Tape

```
apt-get install python3-dev default-libmysqlclient-dev
source /var/www/intranet/venv/bin/activate
pip install mysqlclient
deactivate
```

Cette opération sera à ignorer quand le paquet `mysqlclient` sera dans le fichier `requirements.txt`.

2.4 Configuration de l'intranet

Crée un fichier pour stocker tous les identifiants de ta base de données :

```
mkdir /etc/intranet
cd /etc/intranet/
cat > db.conf << EOF
> [client]
> database = intranet
> user = intranet
> password = k******3
> default-character-set = utf8
EOF
chmod 640 db.conf
chown root:www-data db.conf
```

Modifie le fichier `/var/www/intranet/intranet/settings.py`.

```
SECRET_KEY = ...à regénérer...

DEBUG = False
ALLOWED_HOSTS = ['intranet.telecom-etude.com', 'intranet.telecom
    -etude.fr']

DATABASES = {
        'default': {
                'ENGINE': 'django.db.backends.mysql',
                'OPTIONS': {
                        'read_default_file': '/etc/intranet/db.
                            conf'
                },
                'CONN_MAX_AGE': None,
        }
}
```

Et, à présent, fais une migration initiale.

```
cd /var/www/intranet/
source venv/bin/activate
python manage.py makemigrations
python manage.py migrate
deactivate
```

2.5 Mise en place du design de l'administration

Tape

```
cd /var/www/intranet/
sudo -u www-data cp -r venv/lib/pythonX.X/site-packages/django/
    contrib/admin/static/admin static/
```

3 Configuration de Apache2

Le plus dur est fait ! Il ne te reste plus qu'à créer un petit hôte virtuel Apache qui utilise le module WSGI.

3.1 Génération du certificat

Comme d'habitude pour créer des certificats, tape

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d intranet.telecom-etude.com -d intranet.
    telecom-etude.fr
/etc/init.d/apache2 start
```

3.2 Création d'un hôte virtuel

Crée pour cela un fichier `intranet.conf`

```
vim /etc/apache2/sites-available/intranet.conf
```

dont le contenu est

```
<VirtualHost *:80>
        ServerAdmin admin@telecom-etude.com
        ServerName intranet.telecom-etude.fr
        ServerAlias intranet intranet.telecom-etude.com

        RewriteEngine on
        RewriteCond %{HTTPS} !=on
        RewriteRule ^(.*) https://%{SERVER_NAME}$1 [R,L]
</VirtualHost>

<VirtualHost *:443>
        ServerAdmin admin@telecom-etude.com
        ServerName intranet.telecom-etude.fr
        ServerAlias intranet intranet.telecom-etude.com

        Include /etc/letsencrypt/options-ssl-apache.conf
        SSLCertificateFile    /etc/letsencrypt/live/intranet.
            telecom-etude.com/fullchain.pem
        SSLCertificateKeyFile /etc/letsencrypt/live/intranet.
            telecom-etude.com/privkey.pem

        ErrorLog /var/log/apache2/intranet.log

        WSGIDaemonProcess intranet python-path=/var/www/intranet
             python-home=/var/www/intranet/venv
```

```
        WSGIProcessGroup intranet

        WSGIScriptAlias /update /var/www/intranet/intranet/wsgi.
            py process-group=intranet
        <Directory /var/www/intranet/>
                <Files wsgi.py>
                        Order deny,allow
                        Allow from all
                </Files>
        </Directory>
</VirtualHost>
```

Quinzième partie
Gitlab

Ressources :
— https://docs.docker.com/install/linux/docker-ce/debian/
— https://docs.gitlab.com/omnibus/docker/
— https://docs.gitlab.com/ee/raketasks/backup_restore.html

1 Installation de Docker CE

1.1 Suppression des éventuelles anciennes versions de Docker

Tape

```
apt-get remove docker docker-engine docker.io
```

1.2 Mise en place d'un *repository*

Lors de la configuration du gestionnaire de paquets *Apt-Get*, tu as modifié le fichier `/etc/apt/sources.list` pour y insérer une liste de *repositories*, c'est-à-dire une liste de sources sur le Net pour récupérer des paquets. Normalement, lors de la configuration, tu as mis les sources officielles de Debian Stretch. A présent, il va falloir y rajouter le *repository* des images Docker officielles.

```
apt-get update
apt-get install \
     apt-transport-https \
     ca-certificates \
     curl \
     gnupg2 \
     software-properties-common
```

```
curl -fsSL https://download.docker.com/linux/debian/gpg | apt-
    key add -
apt-key fingerprint 0EBFCD88
```

Lors de l'exécution de la dernière commande, une série d'informations va apparaître. Tu dois vérifier que l'empreinte digitale (*fingerprint*) correspond à la série de caractères hexadécimaux suivante (c'est une sécurité pour vérifier que tu as récupéré le bon fichier).

```
9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88
```

Et maintenant, installe le *repository* dans *Apt-Get*.

```
add-apt-repository \
   "deb [arch=amd64] https://download.docker.com/linux/debian \
   $(lsb_release -cs) \
   stable"
```

1.3 Installation de Docker CE

Tape

```
apt-get update
apt-get install docker-ce
```

Et maintenant, tu peux tester l'installation de Docker avec le traditionnel « Hello World ».

```
docker run hello-world
```

Lorsque tu vas lancer cette commande, l'image de « hello-world » va être téléchargée et lancée dans un *container*. Quand celui-ci s'exécute, il affiche un message d'information et s'arrête.

1.4 Quelques commandes utiles pour utiliser Docker

— Pour installer une image (après l'avoir téléchargée si nécessaire) et la démarrer :

```
docker run [options] image
```

— Pour redémarrer une image :

```
docker start image
```

— Pour arrêter une image :

```
docker stop image
```

— Pour supprimer l'installation d'une image :

```
docker rm image
```

2 Installation de Gitlab

2.1 Installation du *container* de Gitlab

Télécharge, installe puis démarre l'image Docker de Gitlab :

```
docker run --detach \
       --hostname localhost \
       --publish 5443:443 --publish 5080:80 --publish 5022:22 \
```

```
        --name gitlab \
        --restart always \
        --volume /srv/gitlab/config:/etc/gitlab \
        --volume /srv/gitlab/logs:/var/log/gitlab \
        --volume /srv/gitlab/data:/var/opt/gitlab \
        gitlab/gitlab-ce:latest
```

Avec cette configuration, le *container* de Gitlab écoutera aux ports 5080, 5443 et 5022. Néanmoins, après son installation dans le *container*, GitLab aura l'impression d'écouter aux ports 80, 443 et 22. Le paramètre `--restart always` permet de redémarrer le *container* de Gitlab lorsque le serveur redémarre.

2.2 Configuration de Apache2

A présent, il va falloir configurer Apache2 afin que toutes les requêtes pour `https://git.telecom-etude.com` arrivant aux ports 80 et 443 soient redirigées vers les ports 5080 et 5443. Gitlab pourra ensuite les exploiter comme il le faut. Ce principe de redirection correspond à un *proxy*.

2.2.1 Création des certificats pour l'url Gitlab

Tape

```
/etc/init.d/apache2 stop
/opt/letsencrypt/letsencrypt-auto --rsa-key-size 4096 certonly
    --standalone -d git.telecom-etude.com -d git.telecom-etude.
    fr
/etc/init.d/apache2 start
```

2.2.2 Configuration d'un proxy Apache2

Crée un fichier `/etc/apache2/sites-available/gitlab.conf`...

```
cd /etc/apache2/sites-available/
vim gitlab.conf
```

et remplis-le avec le contenu suivant :

```
<VirtualHost *:80>
        ServerAdmin admin@telecom-etude.com
        ServerName git.telecom-etude.com
        ServerAlias gitlab git.telecom-etude.fr
        ServerSignature Off

        RewriteEngine on
        RewriteCond %{HTTPS} !=on
        RewriteRule ^(.*) https://%{SERVER_NAME}$1 [R,L]
</VirtualHost>
```

```
<VirtualHost *:443>
        ServerAdmin admin@telecom-etude.com
        ServerName git.telecom-etude.com
        ServerAlias gitlab git.telecom-etude.fr

        Include /etc/letsencrypt/options-ssl-apache.conf
        SSLCertificateFile    /etc/letsencrypt/live/git.telecom-
            etude.com/fullchain.pem
        SSLCertificateKeyFile /etc/letsencrypt/live/git.telecom-
            etude.com/privkey.pem

        ErrorLog /var/log/apache2/gitlab.log

        ProxyPreserveHost On
        AllowEncodedSlashes NoDecode

        <Location />
                ProxyPass "http://localhost:5080/"
                Require all granted
                ProxyPassReverse "http://localhost:5080/"
                ProxyPassReverse "http://git.telecom-etude.com/"
        </Location>
</VirtualHost>
```

Il ne te reste plus qu'à activer cet hôte et à recharger *Apache2*.

```
a2ensite gitlab
a2enmod rewrite ssl proxy proxy_http
/etc/init.d/apache2 reload
```

2.3 Configuration de Gitlab

2.3.1 Initialisation de Gitlab

Avec un navigateur, va sur le site https://git.telecom-etude.fr/. Si une erreur 502 s'affiche, cela signifie que Gitlab est train de démarrer et il te suffit d'attendre quelques dizaines de secondes (en réactualisant la page).

Ensuite, le mot de passe *root* va t'être demandé. Renseigne-le mot de passe usuel.

2.3.2 Configuration de LDAP sur Gitlab

Edite le fichier de configuration de GitLab dans le *container*.

```
docker exec -it gitlab /bin/bash
vim /etc/gitlab/gitlab.rb
```

Modifie les paramètres suivants :

```
gitlab_rails['ldap_enabled'] = true
gitlab_rails['ldap_servers'] = YAML.load <<-EOS
        main:
                label: 'Telecom Etude LDAP'
                host: 'ldap.telecom-etude.com'
                port: 636
                uid: 'uid'
                bind_dn: 'cn=admin,dc=telecom-etude,dc=com'
                password: 'mot de passe usuel'
                encryption: 'simple_tls'
                verify_certificates: true
                active_directory: false
                allow_username_or_email: false
                lowercase_user: false
                block_auto_created_users: false
                base: 'ou=People,dc=telecom-etude,dc=com'
                user_filter: ''
EOS
```

Redémarre Gitlab.

```
gitlab-ctl reconfigure
gitlab-rake gitlab:ldap:check
```

La seconde commande vérifie la configuration et, si tout se passe bien, devrait afficher une liste de 100 membres de LDAP.

2.3.3 Configuration de Gitlab

Pour configurer l'interface en français, va à « Compte Administrator / Settings / Preferences / Localization / Language / Français ».

Dans le panneau d'administration, renseigne :

1. Paramètres > Général > Restrictions d'inscription
   — Désactivation de la possibilité de créer des comptes de l'extérieur :
   Sign-up enabled : OFF
2. Paramètres > Général > Contrôles de visibilité et d'accès
   — Default branch protection : Not protected
   — Désactivation de la possibilité de créer des projets publics ou semi-publics :
   Restricted visibility levels : Interne & Public
   — Pour activer l'accès par HTTPS (cela ne sert à rien d'activer l'accès par SSH car cela ne fonctionne pas si on n'est pas dans le réseau de Télécom ParisTech)
   Enabled Git access protocols : Only HTTP(S)
3. Paramètres > Général > Limitations du compte

— Par défaut, il faut que les personnes soient externes (cela empêche qu'ils créent des projets) :
New users set to external : YES
— La seule exception est pour les Jetmen (identifiables par leur adresse mail) :
Utilisateurs internes : `@telecom-etude\.com`

4. Paramètres > Intégration et livraison continues > Intégration et déploiement continus
— Default to Auto DevOps pipeline for all projects : NO

Pour personnaliser un peu l'interface, renseigne :

1. Apparence > Favicon : Sélectionne une icône de Telecom Etude
2. Apparence > Sign in/Sign up pages :
— Titre : Gitlab Telecom Etude
— Description :
Dépôt des études pour la Junior Entreprise de Télécom Paristech.
Site web : https://www.telecom-etude.fr/

Il va falloir modifier quelques paramètres du fichier de configuration de Gitlab.

```
docker exec -it gitlab /bin/bash
cd /etc/gitlab/
vim gitlab.rb
```

Modifie les paramètres suivants :

```
## Gitlab URL
external_url "https://git.telecom-etude.com"

...

nginx['listen_port'] = 80
nginx['listen_https'] = false
```

Il n'est pas nécessaire que GitLab écoute le port sécurisé 443. En effet, lors de la configuration de l'hôte virtuel dans *Apache2*, on a redirigé toutes les requêtes vers 5080 :80. Cela évite d'avoir à créer un certificat SSL pour le service GitLab à l'intérieur du docker.

```
gitlab-ctl reconfigure
```

Attention, il y a un bug dans la version actuelle. Alors que tu as configuré GitLab de manière à ce que les jetmen soient définis comme "internes", cela sera ignoré en raison d'un petit bug (https://gitlab.com/gitlab-org/gitlab-ce/issues/52940).

Il est possible de le régler. Pour cela, édite le fichier `/opt/gitlab/embeddled/service/gitlab-rails/app/services/users/build_service.rb` et déplace la condition `if user_default_internal_regex_enabled? ...` en suivant les instructions Git ci-dessous :

```
# /opt/gitlab/embeddled/service/gitlab-rails/app/services/users/
    build_service.rb
@@ -97,10 +97,6 @@ module Users
         if params[:reset_password]
           user_params.merge!(force_random_password: true,
               password_expires_at: nil)
         end
-
-        if user_default_internal_regex_enabled? && !user_params
    .key?(:external)
-          user_params[:external] = user_external?
-        end
       else
         allowed_signup_params = signup_params
         allowed_signup_params << :skip_confirmation if
             skip_authorization
@@ -111,6 +107,10 @@ module Users
         end
       end

+      if user_default_internal_regex_enabled? && !user_params.
    key?(:external)
+        user_params[:external] = user_external?
+      end
+
       user_params
     end
```

Seizième partie
Sauvegarde automatisée

La solution pour laquelle nous avons opté est une solution « faite-maison » qui utilise la commande `rsync`. L'idée est que le serveur principal va envoyer chaque jour les modifications des fichiers à l'ordinateur des *backups*.

1 Sur l'ordinateur des sauvegardes

Avant toute chose, il faut installer `rsync` pour que l'ordinateur puisse récupérer les sauvegardes envoyées par cette commande.

```
apt-get install rsync
```

Maintenant, tu peux créer un nouvel utilisateur qui stockera la sauvegarde.

```
adduser backup-server
```

Puis tu te connecteras avec son compte et tu créeras un dossier.

```
su backup-server
mkdir RSyncBackup
```

Dans la suite, je supposerai que l'adresse IP de l'ordinateur des sauvegardes est 192.168.1.250.

2 Sur le serveur

Avant toute chose, il faut installer `rsync` pour que le serveur puisse envoyer les sauvegardes.

```
apt-get install rsync
```

Commence par créer un utilisateur dédié.

```
adduser backup-server --disaled-login
cd /home/backup-server
```

Crée un dossier `scripts` et ajoute un fichier `launch`.

```
mkdir scripts
vim scripts/launch
```

Complète-le de la manière suivante :

```
#!/bin/bash

##### Configuration #####
DESTINATION='backup-server@192.168.1.250:/home/backup-server/
    RSyncBackup/'
EMAIL='info@telecom-etude.com'
SOURCES='/home /etc /var /usr /srv'
OPTIONS='--archive --verbose --compress --delete-after'
METHOD='--rsh ssh' #Please use an ssh-agent to login to the
    destination

# For emails
EMAIL_FAIL_BEGIN='Errors happened during the backup. Here is the
     log file:\n'
EMAIL_FAIL_END='\n\nTelecom Etude Backup Script'
EMAIL_SUCCESS_BEGIN="TE server has been saved in the backup
    server (${DESTINATION})."
EMAIL_SUCCESS_END='\n\nTelecom Etude Backup Script'
EMAIL_SUCCESS_SEND='no' # 'yes' or 'no'

##### Execute backup #####
LOG_DIR='/var/log/backups/'
mkdir -p ${LOG_DIR}
LOG_DIR="${LOG_DIR}paris-" # Prefix of log files

date >> ${LOG_DIR}history.log

/etc/init.d/slapd stop
/usr/sbin/slapcat -b "dc=telecom-etude,dc=com" -l /var/backups/
    slapd-database.ldif
/etc/init.d/slapd start

docker exec gitlab gitlab-rake gitlab:backup:create STRATEGY=
    copy

rsync $METHOD $OPTIONS $SOURCES "$DESTINATION" 1> ${LOG_DIR}
    last_backup.log 2> ${LOG_DIR}last_errors.log

##### To notify #####
if [ -s ${LOG_DIR}last_errors.log ]
then
        DATE=`date +%F`
        EMAIL_FILE="/tmp/backup_email_${DATE}"
        touch $EMAIL_FILE
        echo -e $EMAIL_FAIL_BEGIN >> $EMAIL_FILE
        cat ${LOG_DIR}last_errors.log    >> $EMAIL_FILE
        echo -e $EMAIL_FAIL_END   >> $EMAIL_FILE
        cat $EMAIL_FILE | mail -s '[IMPORTANT] Server Backup
            Failed !' $EMAIL
        rm $EMAIL_FILE
```

```
else
        if [ $EMAIL_SUCCESS_SEND != 'no' ]
        then
                echo -e $EMAIL_SUCCESS_BEGIN $EMAIL_SUCCESS_END
                    | mail -s 'Successful Server Backup' $EMAIL
        fi
fi
```

Adapte un peu les permissions.

```
chown backup-server:backup-server . -R
chmod 744 scripts/launch
```

A présent, il faut que l'utilisateur *root* puisse accéder au compte *Backup-Server* du serveur de Sophia. Pour cela, si ce n'est pas déjà fait, il faut générer une clé SSH pour le compte super-utilisateur, et l'envoyer sur le serveur distant.

```
su # Connexion au compte super-utilisateur
ssh-keygen # default file (enter), génération d'une clé SSH
ssh-copy-id backup-server@192.168.1.250 # Envoie de la clé sur
    le serveur de Backup
```

Tente de te connecter sur l'ordinateur des sauvegardes. Normalement, tu n'as pas besoin de mot de passe.

```
ssh backup-serveur@192.168.1.250
exit
```

2.1 Utilisation de *Cron*

On souhaite exécuter périodiquement le script de sauvegarde. Pour cela, exécute (en *root*)

```
crontab -e
```

Chaque entrée de la table (chaque ligne) correspond à une tâche à exécuter et doit respecter cette notation :

```
mm hh jj MMM JJJ tâche
```

- `mm` représente les minutes (de 0 à 59) ;
- `hh` représente l'heure (de 0 à 23) ;
- `jj` représente le numéro du jour du mois (de 1 à 31) ;
- `MMM` représente l'abréviation du nom du mois (jan, feb, ...) ou bien le numéro du mois (de 1 à 12) ;
- `JJJ` représente l'abréviation du nom du jour ou bien le numéro du jour dans la semaine :
  - 0 = dimanche
  - 1 = lundi
  - ...
  - 6 = samedi

— 7 = dimanche (représenté deux fois, puisque, selon les pays, le dimanche est considéré comme le premier ou le dernier jour de la semaine).

Pour chaque valeur numérique (mm, hh, jj, MMM, JJJ) les notations possibles sont :

— * : à chaque valeur (0, 1, 2, 3, 4,. . .)
— 5,12 : les valeurs 5 et 12
— 2-5 : les valeurs de 2 à 5 (2, 3, 4, 5)
— */3 : toutes les 3 valeurs (0, 3, 6, 9,. . .)
— 10-20/3 : toutes les 3 valeurs, entre la dixième et la vingtième (10, 13, 16, 19)

Si, sur la même ligne, le « numéro du jour du mois » et le « jour de la semaine » sont renseignés, alors *cron* exécutera la tâche quand l'un des champs correspond. Par exemple, la ligne suivante indique que la tâche doit être exécutée les vendredis ainsi que le 13 de chaque mois, à minuit :

```
0 0 13 * 5 tache
```

Revenons à notre sauvegarde automatisée ; ajoute à la fin la ligne suivante

```
0 3 * * * /home/backup-server/scripts/launch >> /var/log/backup/
    crontab.log
```

qui va exécuter la sauvegarde tous les jours à 3 heures du matin.

Dix-septième partie
Conclusion

C'est ainsi que se conclut le guide. Si tu l'as suivi en entier, tu es capable de monter un serveur de services relativement complet. Mais il reste encore du travail pour réaliser la maintenance :

— Au sein de l'association, *LDAP Account Manager* n'est pas suffisamment complet pour gérer les entrées dans l'annuaire LDAP actuel. Il faudrait, soit modifier la structure de l'annuaire, soit utiliser un autre outil. Sous mon mandat, nous avons développé une API Python répondant à ce besoin, il faudrait donc la réinstaller.
— Tu as certainement installé le gestionnaire des certificats SSL nommé *Let's Encrypt*. Cet outil produit des certificats qui ont une durée de vie de trois mois. Cela signifie qu'il faut les renouveler tous les trois mois (au moins). Cela peut être fait à la main, ou sinon tu peux automatiser le renouvellement en utilisant le *crontab*.
— Il faudrait également mettre à jour régulièrement les paquets via APT. Cette fois-ci, il faut faire attention avec le renouvellement automatique, car tu pourrais casser certains services qui ne seraient pas (encore) compatibles avec des paquets trop récents.
— Et plein d'autres choses...

Un gros service qui pourrait manquer dans ce guide est un cloud autohébergé. Une très bonne solution technique pour ce genre de service est Nextcloud. Cet outil est accompagné d'une large communauté et d'un large panel d'extensions. En particulier, il existe une extension qui permet de faire de l'édition collaborative (indispensable si on veut remplacer Google Drive). Nextcloud est principalement un service de cloud, mais il peut se diversifier (toujours grâce aux extensions), comme par exemple avec une messagerie instantanée. C'est (relativement) facilement configurable et c'est un réel plus pour un serveur. Je n'ai pas mis l'installation de ce service dans le guide, car il me fallait bien mettre un jour un point final à ce dernier. Peut-être qu'un futur membre du pôle voudra faire un **deuxième tome** ?