LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

PROJET: GSB

LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Installation d'un serveur web

Sommaire

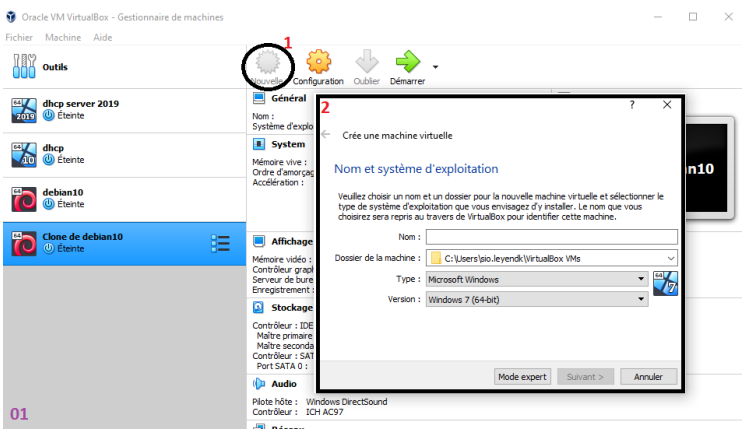
LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

1. *Création et initialisation de la machine virtuelle*

Lancement de VirtualBox :

Ouvrez VirtualBox, l'application de virtualisation, sur votre ordinateur.

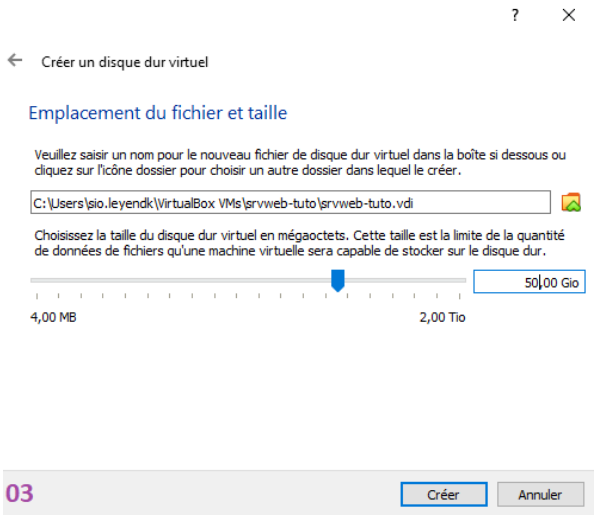
Création de la machine virtuelle (VM) :



Cliquez sur le bouton "Nouvelle" pour créer une nouvelle VM.
Donnez-lui un nom, par exemple "T-Debian 10-Graphique", et sélectionnez le type de système d'exploitation "Linux" et la version Debian (64 bits).

Configuration de la VM :

Dans la fenêtre de configuration de la VM, attribuez la quantité de mémoire RAM souhaitée à la VM .
Créez un nouveau disque dur virtuel en sélectionnant l'option "Créer un disque dur virtuel maintenant"



. Choisissez le type de fichier de disque dur virtuel "VDI (VirtualBox Disk Image)" et laissez les autres options par défaut. Sélectionnez ensuite l'option "Dynamiquement alloué" pour la taille du disque et attribuez la taille souhaitée (par exemple, 50 Go).

Placement de la VM dans un groupe :
Dans la fenêtre principale de VirtualBox, cliquez avec le bouton droit de la souris sur la VM et sélectionnez "Mettre dans un groupe". Créez un nouveau groupe appelé Machines Types et placez la VM dans ce groupe.

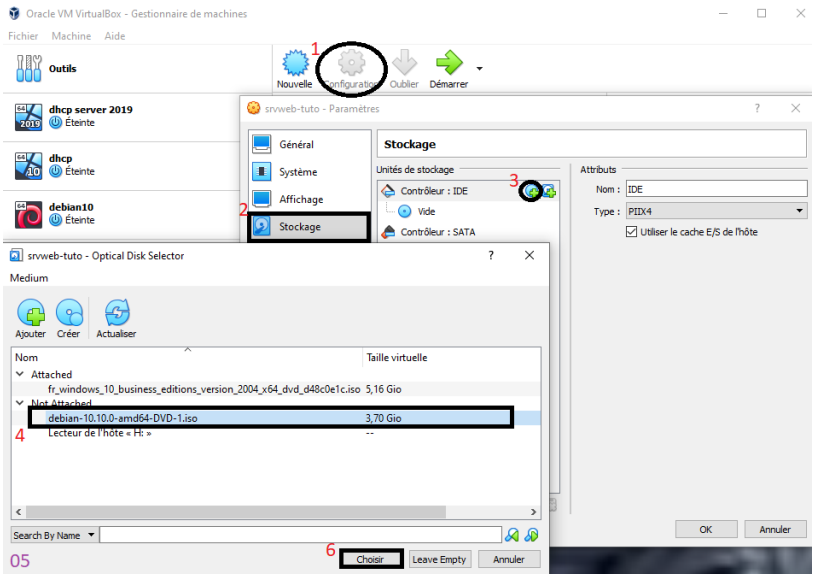
LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Clonage de la VM :

Cliquez avec le bouton droit de la souris sur la VM et sélectionnez "Cloner". Choisissez l'option "Machine actuelle" et cliquez sur "Suivant". Créez un nouveau groupe appelé TC4-TC6 et placez la VM clonée dans ce groupe.

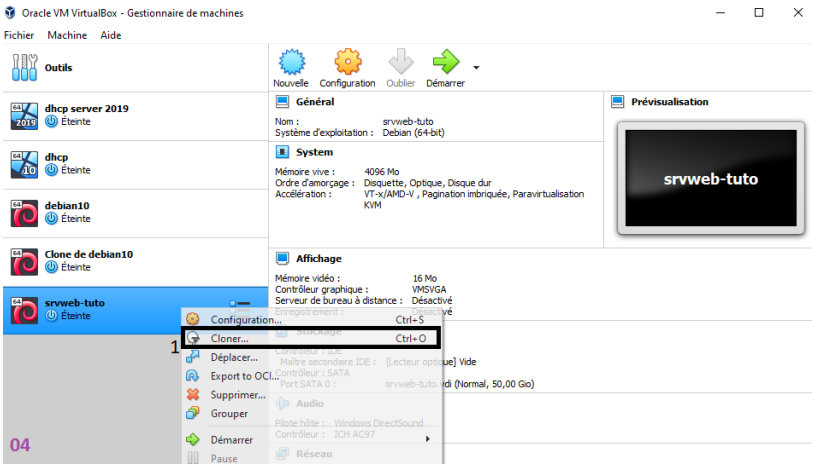
Installation de Debian 10-Buster :

Dans le cadre de l'installation de Debian 10, la première étape consiste à accéder à la configuration de la machine virtuelle. Pour ce faire, il est nécessaire de se rendre dans la sous-catégorie "Stockage" et de cliquer sur l'icône représentant un disque compact. Ensuite, vous devrez sélectionner l'image ISO de Debian 10, disponible gratuitement sur Internet, et cliquer sur "Choisir".



La seconde étape consiste à définir le mode d'accès réseau en choisissant l'option "Accès par pont". Pour cela, il convient de se rendre dans l'onglet "Réseau", puis de cliquer sur le menu déroulant correspondant au mode d'accès réseau et de sélectionner l'option "Accès par pont". Une fois cette étape effectuée, il suffit de cliquer sur "OK" pour valider les modifications.

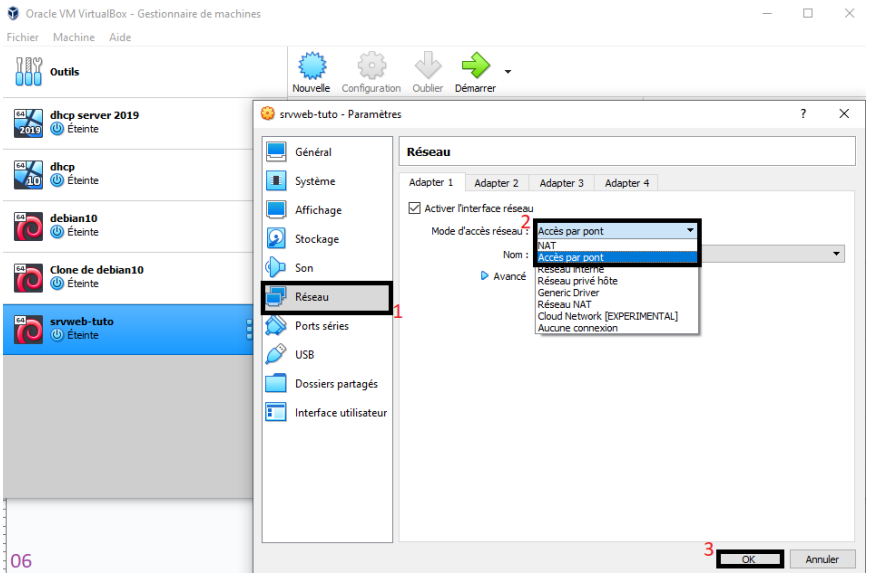
Enfin, il ne reste plus qu'à lancer la machine virtuelle en cliquant sur le bouton approprié. Une fois la VM lancée, vous devrez suivre les étapes pour finaliser l'installation de Debian 10. Dans un souci de simplicité, nous vous recommandons d'opter pour l'installation en mode graphique.



LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Configuration du mode réseau :

Dans la fenêtre de configuration de la VM clonée, accédez à l'onglet "Réseau". Sélectionnez "Accès par pont" dans le menu déroulant "Attaché à". Cette configuration permettra à la VM d'avoir sa propre adresse IP sur le réseau.



Création du fichier apt.conf :

Une fois que nous sommes sur notre machine virtuelle, nous nous connectons en tant qu'utilisateur que nous avons défini précédemment lors de l'installation de Debian 10. Pour ce faire, nous appuyons sur la touche "Windows" et tapons "Terminal" dans la barre de recherche, puis nous ouvrons l'application Terminal.

Une fois le Terminal ouvert, nous saisissons les lignes de code suivantes pour nous connecter en tant que superutilisateur (root) :

```
su root
```

Mot de passe défini par l'utilisateur qui sera demandé après tentative de connexion en root :

```
motDePasse
```

Déplacement dans l'arborescence jusqu'à "/etc/apt" :

```
cd /etc/apt
```

Création et édition d'un fichier nommé "apt.conf" en mode graphique :

```
gedit apt.conf
```

En mode core :

```
nano apt.conf
```

Une fois le fichier ouvert, il va falloir réécrire ces 4 lignes. **Attention** aux fautes :

```
Acquire::http::proxy "http://login:motdepasse@172.16.0.2:3128/";
Acquire::https::proxy "https://login:motdepasse@172.16.0.2:3128/";
Acquire::ftp::proxy "ftp://login:motdepasse@172.16.0.2:3128/";
Acquire::socks::proxy "socks://login:motdepasse@172.16.0.2:3128/";
```

Les quatre lignes suivantes permettent d'établir une connexion à Internet en utilisant le proxy. Il est nécessaire de modifier le nom d'utilisateur et le mot de passe par ceux de la machine hôte qui autorisent la connexion au proxy. De plus, l'adresse IP à la fin de la ligne doit être adaptée en fonction de votre réseau. Il est à noter que cette étape n'est pas obligatoire si vous ne souhaitez pas que votre machine accède à Internet. Cependant, si vous souhaitez y accéder, cette étape devient alors indispensable.

Création du groupe et des utilisateurs:

Sélectionnez la VM clonée dans la liste des machines virtuelles et cliquez sur le bouton "Démarrer". L'installation de Debian 10 graphique devrait se lancer. Ce qui va nous permettre de créer des utilisateurs.
Lors de l'installation de Debian 10 graphique, suivez les instructions à l'écran pour créer trois utilisateurs : doumbib,leyendk et lapezef. Définissez un mot de passe sécurisé pour chaque utilisateur.

Une fois l'installation terminée et après vous être connecté à la VM avec un utilisateur, ouvrez un terminal et exécutez la commande suivante pour ajouter l'utilisateur choisit au groupe "sudo" :

sudo usermod -aG sudo doumbib

Utilisation du compte sudo :
Pour utiliser le compte "doumbib" avec les droits sudo, utilisez la commande suivante pour basculer sur cet utilisateur dans le terminal :

su - doumbib

LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Configuration du réseau :

En mode graphique :

Pour cette étape, la procédure est simple. Accédez aux paramètres réseau de la machine, que vous pouvez trouver dans les paramètres généraux. Ensuite, renseignez l'adresse IP de la machine, son masque, la passerelle et enfin les serveurs DNS.

Dans notre cas, que ce soit en mode core ou graphique, nous avons utilisé les paramètres suivants :
- Adresse IP : 172.16.57.5
- Masque : 255.255.0.0
- Passerelle : 172.16.0.2
- Serveur DNS : 172.16.224.40

En mode core :

Ici la manipulation est un peu plus complexe.
vous allez devoir accéder au fichiers interfaces comme ceci :

```
nano /etc/network/interfaces
```

Et oui, gedit n'existe pas en mode core car il n'y a aucune interface graphique, il nous faut donc utiliser nano qui fonctionne de manière similaire.

le fichier devrait à peu près ressembler à ça :

```
# The loopback network interface
allow-hotplug lo
iface lo inet loopback

 # The primary network interface
auto xxxxx
iface xxxxx inet dhcp
```

les "xxxx" pouvant être eth0 ou enp4s0 par exemple.

Modifier le fichier pour qu'il ressemble à ceci :

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
```

```
auto xxxxx
iface xxxxx inet static
address 172.16.57.5
netmask 255.255.0.0
gateway 172.16.0.2
```

les informations en orange étant celles à adapter selon votre configuration réseau. Enregistrer le fichier en faisant CTRL+X puis exécutez ces lignes de codes afin d'ouvrir le fichier "resolv.conf" :

```
nano /etc/resolv.conf
```

mettez l'ip de votre box dans "nameserver" puis enregistrez et fermez le fichier. (il est possible qu'il n'y ait besoin d'aucune modification à apporter)

enfin relancez la configuration réseau :

```
/etc/init.d/networking restart
```

Mettre à jour les miroirs de téléchargement :

Modifiez le fichier "/etc/apt/sources.list" pour mettre à jour les miroirs de téléchargement avec ceux du site :

```
# Debian Buster, dépôt principal
deb http://deb.debian.org/debian/ buster main
deb-src http://deb.debian.org/debian/ buster main

# Debian Buster, mises à jour de sécurité
deb http://deb.debian.org/debian-security/ buster/updates main
deb-src http://deb.debian.org/debian-security/ buster/updates main

# Debian Buster, mises à jour "volatiles"
deb http://deb.debian.org/debian/ buster-updates main
deb-src http://deb.debian.org/debian/ buster-updates main
```

L'étape de la modification des sources est délicate, car il peut être nécessaire d'apporter quelques ajustements pour assurer le bon fonctionnement du système. Notamment, dans certains cas, il peut être nécessaire de remplacer "buster/updates" par "buster-updates" pour les mises à jour de sécurité. Les miroirs de téléchargement doivent être inscrits dans un fichier appelé sources.list.

```
gedit /etc/apt/sources.list
```

en mode core:

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

Enfin, mettez les à jours à l'aide de la commande

```
apt-get update
```

Installer Apache2 :

toujours sur le terminal de la machine, exécutez la commande suivante:

```
apt-get install apache2
```

Si vous rencontrez une erreur lors de l'installation, assurez-vous tout d'abord que vous avez une connexion internet fonctionnelle depuis la machine virtuelle. Vérifiez également les fichiers apt.conf et sources.list, car souvent des fautes de frappe, des erreurs de syntaxe ou des bugs dans ces fichiers peuvent entraîner des problèmes lors de l'installation d'Apache et de la poursuite du TP. Ces deux étapes précédentes sont cruciales pour assurer le bon déroulement du processus.

Si la commande s'est exécutée correctement sans aucune erreur (vous devriez voir le message "fait" s'afficher), il est maintenant nécessaire de revenir sur la machine hôte afin de modifier les serveurs DNS. Dans notre cas, nous devons configurer les serveurs DNS pour qu'ils correspondent à ceux du réseau du lycée, à savoir : [insérer les adresses IP des serveurs DNS du réseau du lycée].

```
primaire: 172.16.224.40
secondaire: 172.16.224.10
```

Enfin, si le proxy du lycée est configuré, il va nous falloir ajouter des exceptions au sein des paramètres du proxy. sur windows ceux-ci se trouvent ici:

```
Paramètres -> Réseau et internet -> Proxy -> Configuration
manuelle du proxy
```

dans les exceptions, pour le lycée entrez:

```
172.16.*;*.sio.local
```

Configuration d'Apache2 :

Pour configurer Apache2 et avoir accès au site web , vous devez utiliser les hôtes virtuels.

Mémo :

- apache2.conf : il contient la configuration générale d'Apache.
- ports.conf : il indique les ports utilisés par Apache.

- mods-enabled/*.conf : il contient les configurations des différents modules activés dans Apache.
- sites-enabled/*.conf : il contient les configurations des virtual hosts (hôtes virtuels), qui permettent de gérer plusieurs sites web distincts sur le même serveur. Chaque fichier de configuration correspond à un virtual host spécifique.

Modifiez le fichier de configuration d'Apache2 en utilisant la commande suivante :

```
gedit /etc/apache2/sites-available/NomDuSite.VotreNom.conf
```

en core:

```
nano /etc/apache2/sites-available/NomDuSite.VotreNom.conf
```

Ajoutez la configuration suivante dans le fichier :

```
<VirtualHost *:80>
ServerAdmin contact@grafikart.fr

# Domaines gérés par ce virtualhost
ServerName NomDuSite.VotreNom.sio.local
ServerAlias *. NomDuSite.VotreNom.sio.local

# Racine Web – Chemin d'accès aux fichiers du site web
DocumentRoot /var/www/html/NomDuSite

# Règles spécifiques s'appliquant à ce dossier
<Directory /var/www/html/NomDuSite>
# Des options : ici suivre les liens symboliques
Options FollowSymLinks

# Autoriser l'override, autrement dit les .htaccess
AllowOverride All
</Directory>

# Où placer les logs pour cette hôte
ErrorLog /var/www/html/NomDuSite/error.log
CustomLog /var/www/html/NomDuSite/access.log combined
</VirtualHost>
```

voici les VirtualHost pour un site en https pour les plus curieux:

```
<VirtualHost *:443>
ServerAdmin contact@grafikart.fr

# Domaines gérés par ce virtualhost
ServerName NomDuSite.VotreNom.sio.local
ServerAlias *. NomDuSite.VotreNom.sio.local

# Racine Web – Chemin d'accès aux fichiers du site web
DocumentRoot /var/www/html/NomDuSite

# Règles spécifiques s'appliquant à ce dossier
<Directory /var/www/html/NomDuSite>

# Des options : ici suivre les liens symboliques
Options FollowSymLinks

# Autoriser l'override, autrement dit les .htaccess
AllowOverride All
</Directory>

# Où placer les logs pour cette hôte
ErrorLog /var/www/html/NomDuSite/error.log
CustomLog /var/www/html/NomDuSite/access.log combined

SSLEngine on
SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt
SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key
</VirtualHost>
```

Mémo :

Les hôtes virtuels sont une fonctionnalité d'Apache qui permet de gérer plusieurs sites web en même temps. Si on examine le contenu du dossier "sites-enabled",on remarque qu'il n'y a qu'une seule configuration, qui est en réalité un lien symbolique vers le dossier "sites-available". En général, pour faciliter l'activation et la désactivation rapide des configurations, on place nos fichiers de configuration dans le dossier "sites-available" et on crée un lien symbolique depuis le dossier "sites-enabled" pour activer une configuration spécifique. Ainsi, en créant ce lien symbolique, on active l'hôte virtuel correspondant. Cette approche permet de gérer facilement les différentes configurations des hôtes virtuels sans avoir à modifier directement les fichiers dans le dossier "sites-enabled".

Si les dossier se trouve autre part que dans /var/www/html il faut activer l'option FollowSymLinks (L'option "FollowSymLinks" est une directive de configuration utilisée par d'Apache pour spécifier le comportement de suivi des liens symboliques (symbolic links). Un lien symbolique est un type spécial de fichier qui agit comme un raccourci ou une référence vers un autre fichier ou dossier.)

ln -s /home/dev/www /var/www/html/NomDuSite

Pour placer notre configuration dans le dossier sites-enabled on fait le lien symboliques avec la commande :
sudo a2ensite NomDuSite.VotreNom.conf

Ensuite on recharge la configuration d'Apache avec la commande :
sudo apache2ctl configtest ou sudo service apache2 reload

Pour accéder au site on ouvre un navigateur sur la machine hôte et on rentre l'adresse de notre site : http://nomdusite.votrenom.sio.local/ ou avec l'adresse IP ??? du site

Dans tous les cas, veuillez noter l'élément marqué en orange. Il s'agit de l'adresse qui vous permettra d'accéder à votre site web. Dans notre cas, nous avons utilisé cette configuration spécifique, cependant, il est important de l'adapter en fonction de l'adresse IP de votre serveur (que vous avez configurée précédemment) ou de l'adresse attribuée dans le DNS.

Ensuite, concentrons-nous sur l'élément marqué en vert. Peu importe son nom, c'est l'emplacement où notre site web sera hébergé. Cependant, pour le moment, aucun site web n'existe à cet endroit. Nous allons donc le créer. Pour ce faire, quittez le fichier de configuration mentionné précédemment et rendez-vous à cet endroit précis.

```
cd /var/www/html
```

Si vous exécutez la commande

```
ls
```

vous verrez qu'il n'y a rien. Par soucis d'organisation, nous allons y créer un dossier qui contiendra toutes les pages de notre site. pour cela :

```
mkdir NomDuSite
```

le nom devra correspondre à celui que l'on aura inscrit en vert au sein du fichier conf à savoir les virtual hosts.

il va ensuite falloir nous faire un lien symbolique entre les dossiers “sites-available” et “sites-enabled”, pour cela exécutez ceci:

```
a2ensite NomDuSite.VotreNom.conf
```

Cela va nous permettre tout simplement d’activer la configuration.
En dernier lieu, le moment tant redouté causant souvent de nombreux problèmes.. le test de la configuration suivi du rechargement de celle-ci.
Test de la configuration:

```
apache2ctl configtest
```

Rechargement de celle-ci

```
service apache2 reload
```

Si ces deux commandes ne vous renvoient aucune erreur, félicitations ! vous pouvez passez à la suite sans problème.. Dans ce cas, revérifiez votre fichier .conf et parfois même plus loin, les miroirs de téléchargement, le apt.conf ou même la configuration réseau, le problème se situe très souvent dans ces trois endroits.

Mise en place du FTP :

À présent, afin de permettre à nos utilisateurs de modifier les fichiers et de mettre à jour le site internet, nous devons mettre en place un protocole de transfert de fichiers FTP (File Transfer Protocol). Pour ce faire, nous devons installer le paquet requis.

```
apt-get install -y proftpd
```

un fois le paquet bien installé, nous allons créer notre propre fichier de configuration afin de mettre en place certaines règles au sein de ce ftp:

```
gedit /etc/proftpd/conf.d/ftp-perso.conf
```

en core:

```
nano /etc/proftpd/conf.d/ftp-perso.conf
```

Voici ce qu’il va falloir écrire au sein de ce fichier, voici en orange ce que vous pourrez modifiez:

```
# Nom du serveur (identique à celui définit dans /etc/hosts)
ServerName "NomDuSite.VotreNom.sio.local"

# Message de connexion
DisplayLogin "Bienvenue sur le FTP de nom"

# Désactiver IPv6
UseIPv6 off
```

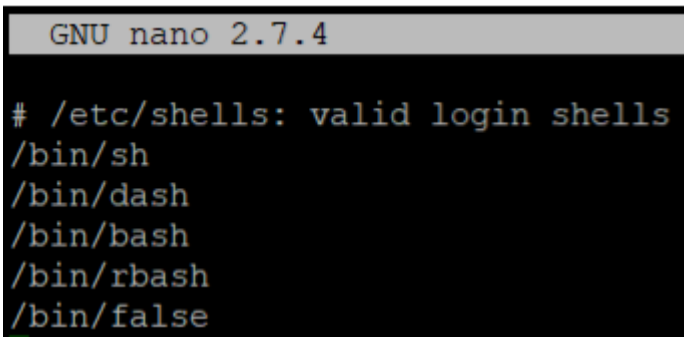
```
# Chaque utilisateur accède seulement au dossier du site (pour les
membres du groupe ftp2100)
DefaultRoot /var/www/html/NomDuSite ftp2100

# Port (défaut = 21)
Port 2100

# Refuser la connexion super-utilisateur "root"
RootLogin off

# Nombre de clients FTP max.
MaxClients 2

# Autoriser la connexion seulement aux membres du groupe "ftp2100"
grâce à la directive DenyGroup.
# En précisant "!" on refuse tout sauf le groupe "ftp2100"
<Limit LOGIN>
DenyGroup !ftp2100
</Limit>
```

une fois votre configuration décidé, plus qu'à recharger la configuration du paquet proftpd:

```
systemctl reload proftpd
```

Afin d'empêcher l'accès au ftp vis le shell, vous pouvez exécutez cette commande qui pointera vers un faux shell, un binaire qui n'existe pas:

```
echo "/bin/false" >> /etc/shells
```

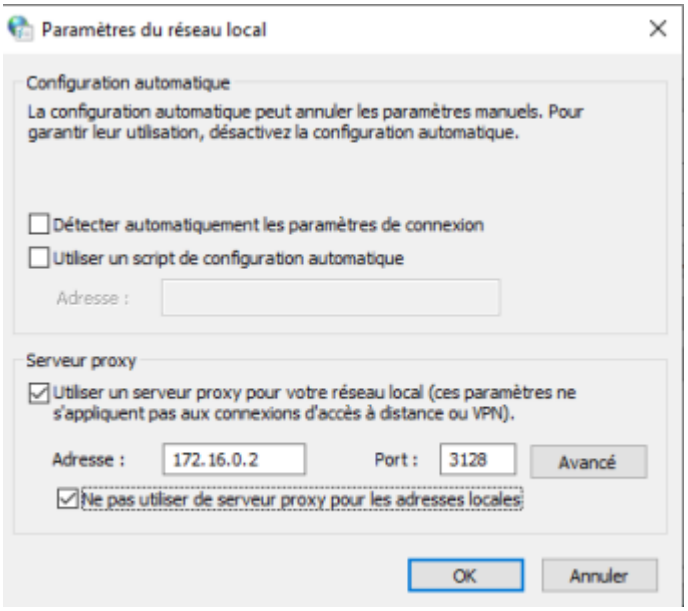


Pour tester la connexion à notre serveur FTP on à utiliser FileZilla

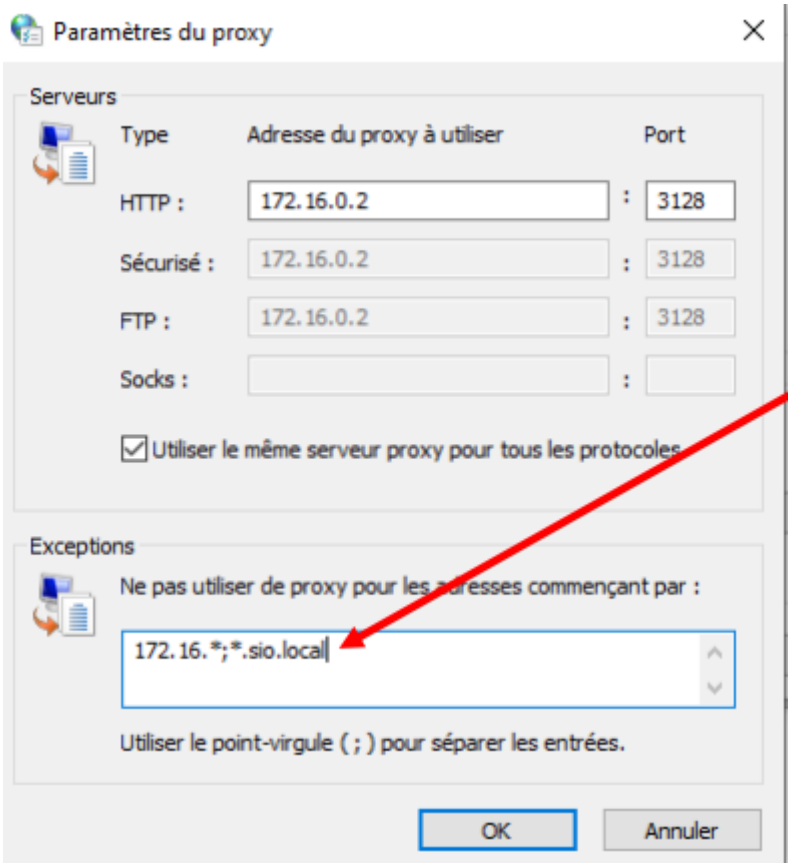
LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Modifier les options internet sur la machine hôte :

Connexions → Paramètres réseaux



Serveur proxy → Avancé



LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Cryptage des échanges entre le serveur web et le client – HTTPS

On installe le protocole SSL sur notre serveur en utilisant la commande : Apt-get install openssl puis on crée un certificat SSL et une clé privée :
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt

X.509 : est une norme d'infrastructure de clé publique utilisée par les protocoles SSL (Secure Sockets Layer) et TLS (Transport Layer Security) pour la gestion des clés et des certificats.

-nodes : que nous utilisons avec OpenSSL indique que nous ne voulons pas sécuriser notre certificat avec une phrase de passe. Dans ce contexte, une phrase de passe empêcherait Apache de lire automatiquement le fichier lors du démarrage du serveur. En désactivant cette option, nous permettons à Apache de lire le certificat sans intervention de l'utilisateur à chaque redémarrage.

-days 365 : fixe la durée de validité du certificat à un an. Cette option est importante car de nombreux navigateurs modernes refusent les certificats dont la durée de validité dépasse un an.

-newkey rsa:2048 : indique à OpenSSL de générer à la fois un nouveau certificat et une nouvelle clé. Étant donné que nous n'avons pas encore créé la clé nécessaire pour signer le certificat, nous devons la générer en même temps que le certificat. La partie "rsa:2048" spécifie la génération d'une clé RSA de 2048 bits.

-keyout et -out : indiquent respectivement à OpenSSL où placer le fichier de clé privée générée et où placer le certificat que nous sommes en train de créer. Ces options permettent de spécifier les emplacements de sauvegarde appropriés pour les fichiers générés par OpenSSL.

Ces fichiers créés sont placés dans /etc/ssl et on met à jour la configuration d'apache avec systemctl restart apache2 pour utiliser le nouveau certificat et la nouvelle clé.

Pour utiliser un certificat SSL on active d'abord mod_ssl qui est un module d'extension d'Apache qui permet de fournir des fonctionnalités de sécurisation des communications via SSL (Secure Socket Layer). Il permet de chiffrer les données échangées entre le serveur Apache et les clients, garantissant ainsi la confidentialité et l'intégrité des informations transmises.

Pour l'activer on utilise la commande : a2enmod ssl
et on redémarre Apache pour activer ce module

On met ensuite à jour notre configuration apache pour pouvoir utiliser notre certificat et notre clé auto-signée. On place des nouveaux fichiers de configuration dans : /etc/apache2/sites-available/ → ils seront donc chargés la prochaine fois que Apache est redémarrer.

On peut soit créer un nouveau fichier de configuration ou modifier celui déjà existant ce qu'on a décidé de faire donc au lieu de tout réécrire on a modifier quelques lignes.

Pour modifier le fichier on utilise cette commande :
gedit /etc/apache2/sites-available/NomDuSite.VotreNom.conf

et on rajoute ces lignes dans ce fichier :

```
<VirtualHost *:443>
ServerName your_domain_or_ip
ServerAlias *. NomDuSite.VotreNom.sio.local
DocumentRoot /var/www/your_domain_or_ip
SSLEngine on
SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt
SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key
</VirtualHost>
```

Ces lignes définissent un hôte virtuel (VirtualHost) pour le port 443, qui est généralement utilisé pour les connexions HTTPS sécurisées. Voici une explication ligne par ligne :

1. `<VirtualHost *:443>` : Cette balise indique le début de la configuration d'un hôte virtuel pour le port 443. L'astérisque (*) signifie que l'hôte virtuel est accessible via toutes les adresses IP du serveur.

2. `ServerName your_domain_or_ip` : Cette ligne spécifie le nom de domaine ou l'adresse IP associée à cet hôte virtuel. Vous devez remplacer "your_domain_or_ip" par le nom de domaine réel ou l'adresse IP du serveur.

3. `ServerAlias *.NomDuSite.VotreNom.sio.local` : Cette directive permet de spécifier des alias supplémentaires pour cet hôte virtuel. Ici, l'alias est défini pour tous les sous-domaines du domaine spécifié. Vous devrez remplacer "NomDuSite" et "VotreNom" par les valeurs appropriées.

4. `DocumentRoot /var/www/your_domain_or_ip` : Cette ligne indique le répertoire racine des fichiers pour cet hôte virtuel. Les fichiers correspondants à ce domaine seront servis à partir de ce répertoire. Vous devez remplacer "your_domain_or_ip" par le nom de domaine ou l'adresse IP correspondante.

5. `SSLEngine on` : Cette directive active le moteur SSL/TLS pour cet hôte virtuel. Elle permet d'utiliser le chiffrement SSL/TLS pour sécuriser les connexions.

6. `SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt` : Cette ligne spécifie le chemin du fichier de certificat SSL/TLS utilisé pour sécuriser les connexions. Dans cet exemple, le certificat utilisé est "/etc/ssl/certs/apache-selfsigned.crt". Vous devrez remplacer ce chemin par le chemin réel de votre certificat.

7. `SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key` : Cette directive indique le chemin du fichier de clé privée correspondant au certificat SSL/TLS. Le fichier de clé privée associé au certificat doit être spécifié ici. Dans cet exemple, le fichier de clé privée est "/etc/ssl/private/apache-selfsigned.key". Vous devrez utiliser le chemin approprié pour votre clé privée.

8. `</VirtualHost>` : Cette balise marque la fin de la configuration de l'hôte virtuel pour le port 443.

En résumé, ces lignes définissent un hôte virtuel pour le port 443 avec un nom de domaine ou une adresse IP spécifiée, un répertoire racine des fichiers, et des paramètres de sécurité SSL/TLS tels que le certificat et la clé privée à utiliser pour les connexions sécurisées.

Avec la configuration actuelle le site répond qu'au demandes HTTPS avec le port 443

On réouvre le fichier de configuration pour palier à ce problème :

/etc/apache2/sites-available/NomDuSite.VotreNom.conf

On crée un nouveau bloc VirtualHost pour faire correspondre les demandes du port 80 on utilise la directive ServerName pour lier à nouveau le nom de domaine. Puis on utilise Redirect pour faire correspondre les requêtes et les envoyer au VirtualHost SSl :

<VirtualHost *:80>
ServerName your_domain_or_ip
Redirect / https://your_domain_or_ip/
</VirtualHost>

Pour appliquer cette nouvelle configuration on redémare Apache :

sudo apachectl configtest ou sudo systemctl reload apache2

Et on test en utilisant le HTTP:// devant l'adresse et on regarde si ça nous redirige automatiquement sur le HTTPS://

Installer MySql

On installe le paquet mysql-server avec la commande : apt-get install mysql-server
Et on sécurise la base de données avec la commande : mysql_secure_installation → La commande mysql_secure_installation est utilisée pour améliorer la sécurité d'une installation de MySQL. Elle permet d'exécuter une série d'opérations pour configurer certains paramètres de sécurité de base, notamment :

- Demande de mot de passe root : La commande mysql_secure_installation peut vous demander d'entrer le mot de passe root actuel pour MySQL. Cela garantit que vous avez les droits nécessaires pour effectuer les modifications de sécurité.

- Suppression des utilisateurs anonymes : Par défaut, MySQL peut autoriser des connexions anonymes sans mot de passe. mysql_secure_installation vous permet de supprimer ces utilisateurs anonymes pour renforcer la sécurité.

- Désactivation de la connexion à distance pour l'utilisateur root : Par mesure de sécurité, mysql_secure_installation peut vous permettre de désactiver la possibilité de vous connecter à distance en tant qu'utilisateur root. Cela réduit les risques d'accès non autorisés à votre base de données.

- Suppression de la base de données de test : MySQL installé une base de données de test par défaut, qui peut être utilisée à des fins de test et de démonstration. Cependant, elle peut présenter des vulnérabilités de sécurité. mysql_secure_installation vous donne la possibilité de supprimer cette base de données.

- Rechargement des privilèges : Après avoir effectué les modifications de sécurité, mysql_secure_installation recharge les privilèges pour s'assurer que les modifications prennent effet immédiatement.

- En résumé, la commande mysql_secure_installation est utilisée pour effectuer une configuration de base de sécurité pour MySQL, notamment en supprimant les utilisateurs anonymes, en désactivant la connexion à distance pour l'utilisateur root et en supprimant la base de données de test. Cela aide à renforcer la sécurité de votre installation de MySQL.

On visualise ensuite la base de données : mysql -u root -p

LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Installer PhpMyAdmin

Phpmyadmin va nous permettre de gérer les base de données avec une interface web ce qui simplifie la tâche

Tout d'abord on installe les paquets : apt-get install phpmydamin → On sélectionne le serveur web Apache2 et on accepte l'aide à la configuration.
Pour la configuration graphique on utilise ces commandes :

- nano /etc/apache2/apache2.conf :

  Cette ligne de code ouvre le fichier de configuration principal d'Apache, appelé "apache2.conf", en utilisant l'éditeur de texte "nano". Ce fichier contient la configuration générale d'Apache, et en l'ouvrant, vous pouvez modifier différents paramètres pour personnaliser le comportement du serveur.

- Include /etc/phpmyadmin/apache.conf :

  Cette ligne de code ajoute un fichier de configuration supplémentaire spécifique à phpMyAdmin à la configuration d'Apache. L'inclusion de ce fichier permet à Apache de prendre en compte les paramètres spécifiques à phpMyAdmin, ce qui est nécessaire pour le bon fonctionnement de l'interface web de gestion de la base de données MySQL.

- service apache2 restart :

  Cette ligne de code redémarre le service Apache. Lorsque vous apportez des modifications à la configuration d'Apache, vous devez redémarrer le service pour que les modifications prennent effet. Cette commande arrête le service Apache en cours d'exécution, puis le redémarre avec la nouvelle configuration. Cela permet de s'assurer que les modifications apportées dans les fichiers de configuration prennent effet immédiatement.

LEYENDECKER Killian
DOUMBIA Bamody
LAPEZE Foucault

Pour se connecter à la base de données avec phpmyadmin on doit d'abord créer un utilisateur avec les droits administrateur :
mysql -u root –p → permet d'ouvrir une session de connexion avec le serveur MySQL en tant qu'utilisateur "root" et de demander le mot de passe correspondant. Une fois connecté, vous pouvez exécuter des commandes SQL pour interagir avec la base de données.

[mot de passe] ← définir un mot de passe
MariaDB [(none)]> CREATE USER 'my_user'@'localhost' IDENTIFIED BY 'my_password';
MariaDB [(none)]> GRANT ALL PRIVILEGES ON * . * TO 'my_user'@'localhost';
MariaDB [(none)]> GRANT GRANT OPTION ON * . * TO 'my_user'@'localhost';
MariaDB [(none)]> FLUSH PRIVILEGES;