

# Notice de Configuration du Serveur Debian 12

---

## 1. Prérequis

- **Serveur de virtualisation** : 8GB RAM, BIOS OVMF UEFI, Carte graphique VirtioGPU.
  - **Disques** :
    - **Disque 1 (50 GB)** : Partitionné en EXT4 (sauf partition "boot EFI" et swap).
    - **Disque 2 (30 GB)** : Données variables (**/var**).
    - **Disque 3 (30 GB)** : Données utilisateurs (**/home**).
- 

## 2. Plan de Partitionnement

- **Disque 1 (50 GB)** :
    - **boot EFI** : 500 MB
    - Racine (/) : 40 GB
    - Swap : 6 GB
    - **/tmp** : 4 GB
  - **Disque 2 (30 GB)** : **/var**
  - **Disque 3 (30 GB)** : **/home**
- 

## 3. Configuration des Comptes Utilisateurs

- **Service "Administration Informatique"** : 5 premiers utilisateurs.
  - **Service "Administratif"** : 5 suivants.
  - **Autres utilisateurs** : Répartis équitablement.
  - **Mots de passe** : Générés avec **pwgen** ou **crunch** (8 caractères alphanumériques).
- 

## 4. Services à Installer et Configurer

### 4.0 Accès SSH

- **Activation SSH** :

```
ip domain-name example.com
crypto key generate rsa
line vty 0 4
transport input ssh
login local
```

- Clé SSH

```
ssh-keygen -t rsa -b 4096
```

- Désactivation SSHv1 : *Modifier /etc/ssh/sshd\_config :*

```
Protocol 2
```

- Algorithmes recommandés par l'ANSSI:

```
Ciphers aes256-ctr,aes192-ctr,aes128-ctr
MACs hmac-sha2-512,hmac-sha2-256
KexAlgorithms ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-
hellman-group-exchange-sha256
```

---

## 4.1 Service DNS (Cache) avec Bind9

- Installation :

```
sudo apt update
sudo apt install bind9
```

- Modifiez le fichier /etc/bind/named.conf.options.
- Activez l'option de cache :

```
options {
    directory "/var/cache/bind";
    recursion yes;
    allow-recursion { any; };
    forwarders {
        8.8.8.8; // Serveurs DNS publics (Google DNS)
        8.8.4.4;
    };
    dnssec-validation auto;
};
```

- Démarrer et activer :

```
sudo systemctl enable bind9
sudo systemctl start bind9
```

---

## 4.2 Configuration du service SMB pour partager des fichiers

- Installer Samba :

```
sudo apt update
sudo apt install samba -y
```

- Créer les dossiers partagés :

1. Créez les dossiers qui seront partagés via SMB :

```
sudo mkdir -p /srv/samba/lecture
sudo mkdir -p /srv/samba/auth
```

2. Définition des permissions sur les répertoires :

- Lecture seule pour tous les utilisateurs (lecture des fichiers) :

```
sudo chmod 755 /srv/samba/public
sudo chown root:root /srv/samba/public
```

Cela permet à tout utilisateur d'accéder en lecture au dossier, sans possibilité de modification.

- Lecture/Écriture pour les groupes Administratif et Administration Informatique pour le dépôt de fichiers :

```
sudo chmod 775 /srv/samba/auth
sudo chown root:Administratif /srv/samba/auth
sudo chown root:AdministrationInformatique /srv/samba/auth
```

Ici, les groupes "Administratif" et "Administration Informatique" auront des droits de lecture et d'écriture dans le dossier de dépôt.

- Configurer Samba :

1. Modifiez le fichier Samba :

```
sudo nano /etc/samba/smb.conf
```

Ajoutez les sections suivantes à la fin du fichier :

```
[PartagePublic]
path = /srv/samba/public
```

```
browseable = yes
read only = yes
guest ok = yes
valid users = @Utilisateurs

[PartageAuthentifie]
path = /srv/samba/auth
browseable = yes
read only = no
valid users = @Administratif, @AdministrationInformatique
```

- [PartagePublic] permet un accès anonyme en lecture.
- [PartageAuthentifie] nécessite une authentification pour accéder au contenu.

2. Redémarrez Samba :

```
sudo systemctl restart smbd
```

---

## 4.3 Installation et Configuration de Nginx (Serveur Web)

1. Installation de Nginx :

```
sudo apt install nginx
```

2. Démarrer et activer le service Nginx pour qu'il se lance automatiquement au démarrage :

```
sudo systemctl start nginx
sudo systemctl enable nginx
```

3. Placer les fichiers de votre site web dans le répertoire /var/www/html : Si vous avez déjà votre site web prêt, vous pouvez copier les fichiers dans ce répertoire. Par exemple :

```
sudo cp -r /chemin/vers/votre/site/* /var/www/html/
```

- Modifier les permissions du répertoire pour que Nginx puisse lire les fichiers :

```
sudo chown -R www-data:www-data /var/www/html
sudo chmod -R 755 /var/www/html
```

Vérifier votre site web : Accédez à `http://localhost` ou `http://` dans votre navigateur pour voir votre site web. Si vous avez un fichier `index.html`, il sera affiché par défaut.

## 4.4 Installation et Configuration du Serveur TFTP

### 1. Installation de TFTP :

```
sudo apt install tftpd-hpa
```

### 2. Configurer TFTP : Éditez le fichier de configuration de TFTP pour spécifier le répertoire où seront stockés les fichiers (configuration des routeurs et commutateurs, IOS, etc.) 🤖

```
sudo nano /etc/default/tftpd-hpa
```

Modifiez-le pour qu'il ressemble à ceci :

```
TFTP_DIRECTORY="/var/lib/tftpboot"  
TFTP_ADDRESS="0.0.0.0:69"  
TFTP_OPTIONS="--secure"
```

- **TFTP\_DIRECTORY="/var/lib/tftpboot"** : c'est le répertoire où les fichiers seront stockés et servis.
- **TFTP\_ADDRESS="0.0.0.0:69"** : permet à TFTP d'écouter sur toutes les interfaces réseau sur le port 69.
- **TFTP\_OPTIONS="--secure"** : s'assure que le serveur TFTP fonctionne dans un mode sécurisé.

### 3. Créer le répertoire TFTP où vous allez stocker les fichiers (configurations et IOS) :

```
sudo mkdir /var/lib/tftp  
sudo chmod -R 777 /var/lib/tftp
```

Le répertoire `/var/lib/tftpboot` est maintenant prêt à accueillir vos fichiers.

### 4. Redémarrer le service TFTP pour appliquer les modifications :

```
sudo systemctl restart tftpd-hpa  
sudo systemctl enable tftpd-hpa
```

Transférer les fichiers de configuration des commutateurs/routeurs et les IOS dans le répertoire `/var/lib/tftpboot` : Vous pouvez déplacer vos fichiers dans le répertoire TFTP. Par exemple :

```
sudo cp /chemin/vers/configuration_router.txt /var/lib/tftp/  
sudo cp /chemin/vers/ios.bin /var/lib/tftp/
```

