Déploiement d'un serveur virtuel Debian 12

Période : 10/2025 – 11/2025	**Entreprise :** SNCF — DSI	**Candidat :** Soul Florian

Objectif de la mission

Déployer et configurer un serveur virtuel Debian 12 hébergeant un service web Apache2 avec accès SSH sécurisé et pare-feu UFW, dans un environnement de formation.

Compétences BTS SIO mobilisées

Gérer le patrimoine informatique	Mettre à disposition des utilisateurs un service informatique
Vérifier la continuité d'un service informatique	

1. Présentation de la mission

Dans le cadre des travaux pratiques de formation, déploiement d'un serveur virtuel sous Debian 12 hébergeant un service web Apache avec accès SSH sécurisé. L'objectif est de maîtriser l'installation d'un système GNU/Linux en environnement virtualisé et la mise en production d'un service réseau.

2. Environnement technique

Composant	Valeur
Hyperviseur	VirtualBox 7.x ou VMware Workstation
OS invité	Debian 12 (Bookworm) — 64 bits
RAM allouée	2 Go minimum
Disque	20 Go (dynamique)
Réseau	Mode pont (Bridged) ou NAT avec redirection de ports
Services installés	OpenSSH-server, Apache2, UFW

3. Étapes de réalisation

3.1 — Installation de Debian 12

- Télécharger l'ISO Debian 12 netinst depuis debian.org
- Créer une VM avec les ressources ci-dessus
- Lors de l'installation : choisir une installation minimale sans bureau graphique

- Créer un utilisateur standard (ex : florian) et définir un mot de passe root fort
- Sélectionner uniquement : « Serveur SSH » et « Utilitaires usuels du système »

3.2 — Configuration post-installation

```
# Mise à jour du système apt update && apt upgrade -y # Installation des outils de base apt install -y curl wget vim net-tools ufw
```

3.3 — Sécurisation SSH

Modifier la configuration SSH pour renforcer la sécurité :

```
# Éditer /etc/ssh/sshd_config Port 2222 # Changer le port par défaut PermitRootLogin no # Interdire la connexion root PasswordAuthentication yes # (à passer en no si clé configurée) MaxAuthTries 3 # Redémarrer le service systemctl restart sshd
```

■ *Générer une paire de clés SSH côté client avec ssh-keygen puis copier la clé publique avec ssh-copy-id pour renforcer l'authentification.*

3.4 — Installation et configuration d'Apache2

```
apt install -y apache2 systemctl enable apache2 systemctl start apache2 # Créer une page d'accueil personnalisée echo '<h1>Serveur SNCF - DSI</h1>' > /var/www/html/index.html
```

3.5 — Configuration du pare-feu UFW

```
ufw allow 2222/tcp # SSH ufw allow 80/tcp # HTTP ufw allow 443/tcp # HTTPS ufw enable ufw status verbose
```

4. Tests de validation

Test	Commande	Résultat attendu
Connectivité SSH	ssh -p 2222 florian@IP_SERVEUR	Connexion établie
Service Apache actif	systemctl status apache2	active (running)
Page web accessible	curl http://IP_SERVEUR	HTML affiché
Pare-feu actif	ufw status	Status: active, ports ouverts
Ping serveur	ping -c 4 IP_SERVEUR	0% packet loss

5. Supervision basique

```
# Vérifier les logs Apache en temps réel tail -f /var/log/apache2/access.log # Vérifier l'utilisation des ressources htop df -h free -m
```

6. Livrables attendus

- Capture d'écran de la connexion SSH réussie
- Capture de la page web accessible depuis un navigateur
- Capture du statut des services (systemctl status apache2, sshd)
- Capture de ufw status avec les règles actives