

# Mise en place d'une administration sécurisée et d'un accès SSH avec authentification par clé sur Linux

## Tableau de matières

<b>Cahier des Charges</b>	<b>2</b>
1. Contexte et Problématique	2
2. Expression des Besoin	2
3. Description de l'existant	2
4. Analyse des Choix Techniques	3
5. Offre du Marché	3
<b>Mise en œuvre</b>	<b>4</b>
1. Matériel et logiciels utilisés	4
2. Étapes de mise en œuvre	4
Étape 1 : Création de l'utilisateur administrateur	4
Étape 2 : Définir un mot de passe sécurisé	4
Étape 3 : Création du groupe IT	5
Étape 4 : Ajouter l'utilisateur aux groupes sudo et it-team	5
Étape 5 : Test des droits sudo	5
Étape 6 : Installation du serveur SSH	5
Étape 7 : Génération de la clé SSH sur le client	6
Étape 8 : Transfert de la clé publique sur Debian	7
Étape 9 : Sécurisation du service SSH	7
Étape 10 : Configuration du firewall	7
Étape 11 : Test de connexion	8

# Cahier des Charges

## 1. Contexte et Problématique

Dans le cadre d'un environnement professionnel, l'entreprise doit garantir la continuité de service et la sécurité de son système d'information. Les administrateurs doivent pouvoir gérer les serveurs de manière fiable et sécurisée, tout en contrôlant les accès et les privilèges. L'objectif est donc de mettre en place une administration sécurisée des utilisateurs et un accès distant via SSH, authentifié par clé.

## 2. Expression des Besoin

- Créer un utilisateur administrateur pour la gestion du serveur.
- Créer un groupe pour l'équipe IT et assigner l'utilisateur à ce groupe avec les droits sudo nécessaires.
- Limiter les commandes disponibles via sudo pour renforcer la sécurité.
- Permettre un accès distant au serveur Debian via SSH avec authentification par clé.
- Sécuriser le service SSH (changement de port, désactivation du mot de passe, interdiction de root).
- Configurer un pare-feu pour autoriser uniquement les accès nécessaires.
- Tester et valider toutes les configurations.

## 3. Description de l'existant

- Le serveur utilise une distribution Linux Debian 12.
- Aucun utilisateur administrateur spécifique n'est créé pour l'équipe IT.
- Aucun groupe dédié n'est présent.
- Le service SSH est installé par défaut mais non configuré pour une authentification sécurisée par clé.
- Le pare-feu n'est pas configuré ou actif.
- Les accès distants se font uniquement par mot de passe standard.

## 4. Analyse des Choix Techniques

- **Gestion des utilisateurs et groupes Linux** : solution native, simple et fiable pour organiser les droits d'accès.
- **OpenSSH-server** : service standard pour les connexions SSH sur Linux.
- **Authentification par clé RSA 4096 bits** : méthode sécurisée, évite les risques liés aux mots de passe.
- **Changement de port SSH et désactivation de root** : réduisent les risques d'attaques automatisées.
- **UFW** : firewall simple et efficace pour filtrer les accès par port.

Ces choix répondent à la problématique en renforçant la sécurité tout en garantissant une administration distante efficace.

## 5. Offre du Marché

Plusieurs solutions d'accès distant existent :

- **Gestion native Linux** : utilisateurs et groupes intégrés, solution open-source et stable.
- **SSH / OpenSSH** : standard open-source, largement documenté et compatible avec Linux/Windows/Mac.
- **Solutions propriétaires d'accès distant** (TeamViewer, AnyDesk) : plus lourdes et non adaptées à la gestion serveur en entreprise.
- **VPN + SSH** : solution possible mais plus complexe

OpenSSH reste la solution la plus adaptée pour un projet SISR, car :

- numérique open-source,
- léger et rapide,
- compatible Linux/Windows/Mac,
- très sécurisé avec clé publique,
- facile à mettre en œuvre et documenté.

# Mise en œuvre

## 1. Matériel et logiciels utilisés

- Serveur Debian 12 (ou autre version utilisée)
- Client Windows pour la génération de clé
- OpenSSH-server (serveur SSH)
- UFW (pare-feu)

## 2. Étapes de mise en œuvre

### Étape 1 : Création de l'utilisateur administrateur

**sudo adduser adminuser --disabled-password --gecos ""**

```
azaanzer@debian-server:~$ id adminuser
uid=1001(adminuser) gid=1001(adminuser) groupes=1001(adminuser),27(sudo),100(users),1002(it-team)
azaanzer@debian-server:~$ groups adminuser
adminuser : adminuser sudo users it-team
azaanzer@debian-server:~$ _
```

### Étape 2 : Définir un mot de passe sécurisé

**sudo passwd adminuser**

```
adminuser@debian-server:~$ sudo passwd adminuser
Nouveau mot de passe :
```

### Étape 3 : Création du groupe IT

**sudo addgroup it-team**

```
azaanzer@debian-server:~$ sudo addgroup it-team
addgroup : Le groupe « it-team » existe déjà.
azaanzer@debian-server:~$
```

### Étape 4 : Ajouter l'utilisateur aux groupes sudo et it-team

**sudo usermod -aG it-team,sudo adminuser**

```
azaanzer@debian-server:~$ sudo usermod -aG it-team,sudo adminuser
[sudo] Mot de passe de azaanzer :
azaanzer@debian-server:~$ _
```

commande exécutée sans erreur.

## Étape 5 : Test des droits sudo

**su - adminuser**

**sudo whoami**

```
azaanzer@debian-server:~$ su - adminuser
Mot de passe :
adminuser@debian-server:~$ sudo whoami
root
adminuser@debian-server:~$
```

## Étape 6 : Installation du serveur SSH

- Commandes exécutées :
- **sudo apt update**
- **sudo apt install openssh-server -y**
- **sudo systemctl enable --now ssh**

```
azaanzer@debian-server:~$ sudo systemctl status ssh
• ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)
  Active: active (running) since Wed 2025-11-19 18:20:41 CET; 6min ago
    Docs: man:sshd(8)
           man:sshd_config(5)
  Main PID: 4651 (sshd)
    Tasks: 1 (limit: 4606)
  Memory: 1.4M
    CPU: 17ms
  CGroup: /system.slice/ssh.service
          └─4651 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

nov. 19 18:20:41 debian-server systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
nov. 19 18:20:41 debian-server sshd[4651]: Server listening on 0.0.0.0 port 22.
nov. 19 18:20:41 debian-server systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
nov. 19 18:20:41 debian-server sshd[4651]: Server listening on :: port 22.
• azaanzer@debian-server:~$
```

## Étape 7 : Génération de la clé SSH sur le client

- Commande PowerShell :

`ssh-keygen -t rsa -b 4096`

```
C:\Users\azaan>ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\azaan/.ssh/id_rsa):
C:\Users\azaan/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\azaan/.ssh/id_rsa
Your public key has been saved in C:\Users\azaan/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:K6dj4oVMdSik4B+R3inzACz3DzR0ivAbZA2vQVF12bA azaan@Ordi-D-Azaan
The key's randomart image is:
+---[RSA 4096]-----+
| o  +*+.. o+      |
| +=+*=*  o...     |
| oB=B+.+ E        |
| . =B== .         |
| .+*o  S          |
| .o o. .          |
|   o o o          |
|   ..o+           |
|   ..o..          |
+-----[SHA256]-----+
```

## Étape 8 : Transfert de la clé publique sur Debian

Copier la clé privée dans `/home/adminuser/.ssh/authorized_keys` avec les permissions correctes.

```
GNU nano 7.2 /home/adminuser/.ssh/authorized_keys
SHA256:K6dj4oVMdSik4B+R3inzACz3DzR0ivAbZA2vQVF12bA azaan@Ordi-D-Azaan
```

```
chmod 700 /home/adminuser/.ssh
chmod 600 /home/adminuser/.ssh/authorized_keys
chown -R adminuser:adminuser /home/adminuser/.ssh
```

## Étape 9 : Sécurisation du service SSH

- Édition du fichier `/etc/ssh/sshd_config` :
- `Port 2222`
- `PermitRootLogin no`
- `PasswordAuthentication yes`

```
GNU nano 7.2 /etc/ssh/sshd_config
Include /etc/ssh/sshd_config.d/*.conf
Port 22
PermitRootLogin no
PasswordAuthentication yes
```

## Étape 10 : Configuration du firewall

- `sudo ufw allow 2222/tcp`
- `sudo ufw reload`
- `sudo ufw status`

```
azaanzer@debian-server:~$ sudo systemctl restart ssh
azaanzer@debian-server:~$ sudo ufw allow 2222/tcp
Rules updated
Rules updated (v6)
azaanzer@debian-server:~$ sudo ufw reload
Firewall not enabled (skipping reload)
azaanzer@debian-server:~$ sudo ufw status
Status: inactive
azaanzer@debian-server:~$ sudo ufw status
Status: inactive
azaanzer@debian-server:~$ sudo ufw enable
Firewall is active and enabled on system startup
azaanzer@debian-server:~$ sudo ufw status
Status: active
```

To	Action	From
--	-----	----
2222/tcp	ALLOW	Anywhere
2222/tcp (v6)	ALLOW	Anywhere (v6)

## Étape 11 : Test de connexion

- Depuis Debian :

```
ssh -p 2222 adminuser@127.0.0.1
```

```
aasaanzer@debian-server:~$ ssh -p 2222 adminuser@127.0.0.1
adminuser@127.0.0.1's password:
Linux debian-server 6.1.0-40-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.153-1 (2025-09-20) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 19 19:44:45 2025 from 127.0.0.1
adminuser@debian-server:~$ _
```

- Depuis Windows :

```
ssh -p 2222 adminuser@192.168.1.10
```

```
PS C:\Users\aasaan> ssh -p 2222 adminuser@192.168.1.10
adminuser@192.168.1.10's password:
Linux debian-server 6.1.0-40-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.153-1 (2025-09-20) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 19 19:47:34 2025 from 192.168.1.10
adminuser@debian-server:~$
```