

Sécuriser son serveur Proxmox et sa machine Debian (règles de base)



SOMMAIRE

1. SECURISER L'HYPERVISEUR PROXMOX

- Création d'un utilisateur PVE et gestion des permissions
- Désactivation de l'accès SSH root
- Modification du port SSH par défaut
- Installation de Fail2Ban sur PromoxVE 8.1

2. SECURISER LA MACHINE DEBIAN

- Désactivation de l'accès SSH root
- Installation « sudo » et ajout d'un utilisateur au groupe « sudo »
- Installation de Fail2Ban sur Debian 12
- Modification d'un mot de passe utilisateur Debian

© tutos-info.fr - 03/2024



DIFFICULTE



UTILISATION COMMERCIALE INTERDITE

1 – SECURISER SON SERVEUR PROXMOX

Il est primordial de bien sécuriser l'hyperviseur lorsque ce dernier est hébergé chez un fournisseur externe tel que OVH ou Scaleway dont on ne sait rien du réseau. Votre fournisseur peut proposer un firewall et/ou un système anti-DDOS mais cela ne vous protège que de l'extérieur pas de l'intérieur du réseau qui comporte une multitude de serveurs d'autres clients.

La logique pour un réseau en entreprise ou chez soit pour un « home lab » (auto-hébergement) sera la même.

Quel que soit l'hyperviseur, il faut que celui-ci soit sécurisé. S'il n'est pas sécurisé, toutes vos machines virtuelles ou conteneurs sont potentiellement compromis.

1^{ère} étape : création d'un utilisateur « PVE » (vivement conseillé)

L'authentification « **PAM** » correspond à l'authentification standard liée aux utilisateurs Linux. L'authentification « **PVE** » correspond aux utilisateurs créés dans l'environnement Proxmox.

Lors de l'installation de l'hyperviseur Proxmox, un utilisateur « root (PAM) » a été créé par défaut. Il est cependant fortement déconseillé de l'utiliser pour se connecter à l'hyperviseur. L'idéal étant de créer un utilisateur « PVE » auquel on accordera des droits suffisants pour administrer l'hyperviseur.

Pour créer l'utilisateur « PVE », on procédera ainsi :

- Connectez-vous en tant que « root » sur le royaume « PAM »
- Cliquez sur « **Centre de données** » et, dans le volet de droite, sur « **Utilisateurs** »
- Cliquez « **Ajouter** »
- Créez l'utilisateur PVE avec un mot de passe suffisamment fort

Une fois l'utilisateur créé, il convient de lui affecter des « permissions » de la manière suivante :

- Cliquez sur « **Centre de données** » et, dans le volet de droite, sur « **Permissions** »
- Cliquez « **Ajouter** » et « **Permissions de l'utilisateur** »
- Sélectionnez le « **chemin d'accès** » (la ressource pour laquelle vous souhaitez accorder des permissions)
- Sélectionnez l'utilisateur concerné
- Sélectionnez le « **rôle** » que vous voulez attribuer à cet utilisateur PVE :

Proxmox propose plusieurs rôles avec des privilèges bien définis.

Prenez connaissance des privilèges et faites bien attention au moment du choix !

Rôle ↑	Privilèges
Administrator	Datastore.Allocate, Datastore.AllocateSpace, Datastore.AllocateTemplate, Datastore.Audit, Group.Allocate, Mapping.Audit, Mapping.Modify, Mapping.Use, Permissions.Modify, Pool.Allocate, Pool.Audit, Realm.Allocate, Realm.AllocateUser, SDN.Allocate, SDN.Audit, SDN.Use, Sys.AccessNetwork, Sys.Audit, Sys.Console, Sys.Incoming, Sys.Modify, Sys.PowerMgmt, Sys.Syslog, User.Modify, VM.Allocate, VM.Audit, VM.Backup, VM.Clone, VM.Config.CDRom, VM.Config.CPU, VM.Config.Cloudinit, VM.Config.Disk, VM.Config.HWType, VM.Config.Memory, VM.Config.Network, VM.Config.Options, VM.Console, VM.Migrate, VM.Monitor, VM.PowerMgmt, VM.Snapshot, VM.Snapshot.Rollback
NoAccess	

PVEAdmin	Datastore.Allocate, Datastore.AllocateSpace, Datastore.AllocateTemplate, Datastore.Audit, Group.Allocate, Mapping.Audit, Mapping.Use, Pool.Allocate, Pool.Audit, Realm.AllocateUser, SDN.Allocate, SDN.Audit, SDN.Use, Sys.Audit, Sys.Console, Sys.Syslog, User.Modify, VM.Allocate, VM.Audit, VM.Backup, VM.Clone, VM.Config.CDRom, VM.Config.CPU, VM.Config.Cloudinit, VM.Config.Disk, VM.Config.HWType, VM.Config.Memory, VM.Config.Network, VM.Config.Options, VM.Console, VM.Migrate, VM.Monitor, VM.PowerMgmt, VM.Snapshot, VM.Snapshot.Rollback
PVEAuditor	Datastore.Audit, Mapping.Audit, Pool.Audit, SDN.Audit, Sys.Audit, VM.Audit

PVEDatastoreAdmin	Datastore.Allocate, Datastore.AllocateSpace, Datastore.AllocateTemplate, Datastore.Audit
PVEDatastoreUser	Datastore.AllocateSpace, Datastore.Audit
PVEMappingAdmin	Mapping.Audit, Mapping.Modify, Mapping.Use
PVEMappingUser	Mapping.Audit, Mapping.Use
PVEPoolAdmin	Pool.Allocate, Pool.Audit
PVEPoolUser	Pool.Audit
PVESDNAdmin	SDN.Allocate, SDN.Audit, SDN.Use
PVESDNUser	SDN.Audit, SDN.Use
PVESysAdmin	Sys.Audit, Sys.Console, Sys.Syslog

PVETemplateUser	VM.Audit, VM.Clone
PVEUserAdmin	Group.Allocate, Realm.AllocateUser, User.Modify
PVEVMAAdmin	VM.Allocate, VM.Audit, VM.Backup, VM.Clone, VM.Config.CDRom, VM.Config.CPU, VM.Config.Cloudinit, VM.Config.Disk, VM.Config.HWType, VM.Config.Memory, VM.Config.Network, VM.Config.Options, VM.Console, VM.Migrate, VM.Monitor, VM.PowerMgmt, VM.Snapshot, VM.Snapshot.Rollback
PVEVMUser	VM.Audit, VM.Backup, VM.Config.CDRom, VM.Config.Cloudinit, VM.Console, VM.PowerMgmt

Vous pouvez, également, accorder des permissions au niveau des ressources matérielles de votre hyperviseur. Par exemple, si vous cliquez sur le disque « Local » (volet de gauche du centre de données), vous avez la possibilité d'accorder des permissions d'accès à cette ressource (volet de droite).

Il est possible d'aller assez loin dans la configuration des permissions PVE mais, de nos jours, il est important de consacrer du temps à la configuration de ces permissions.

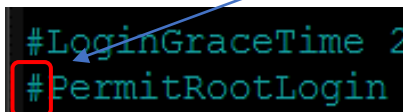
2^{ème} étape : désactivation de l'accès « root » en SSH sur le serveur

Il est fortement recommandé de DESACTIVER l'accès « root » en SSH sur votre hyperviseur !

- Cliquez sur le nom de nœud Proxmox
- Cliquez, dans le volet de droite, sur « Shell »
- Ouvrez le fichier « `sshd_config` » avec la commande :

```
nano /etc/ssh/sshd_config
```

- Assurez-vous que l'accès SSH est bien désactivé **en vérifiant que la ligne** « PermitRootLogin prohibit-password » du bloc « **Authentification** » est bien commentée (#) :



```
#LoginGraceTime 2
#PermitRootLogin
```

- Relancez le service SSH, si vous avez fait des modifications, avec la commande :

systemctl restart ssh

3^{ème} étape : modification du port SSH par défaut du serveur

L'une des premières actions à effectuer sur votre serveur est la configuration du port d'écoute du service SSH.

Par défaut, celui-ci est défini sur le **port 22**. Les tentatives de hack du serveur par des robots vont cibler ce port en priorité.

La modification de ce paramètre, au profit d'un port différent, est une mesure simple pour renforcer la protection de votre serveur contre les attaques automatisées.

Pour cela, effectuez les manipulations suivantes :

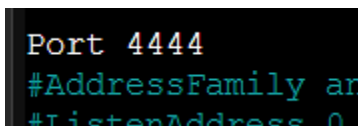
- Cliquez sur le nom de nœud Proxmox
- Cliquez, dans le volet de droite, sur « **Shell** »
- Ouvrez le fichier « **sshd_config** » avec la commande :

nano /etc/ssh/sshd_config

Modifiez le port « 22 » par défaut par un port inutilisé (**au minimum « 1024 »**) en ouvrant le fichier « sshd_config » et en modifiant le port par défaut (en haut du fichier) :

nano /etc/ssh/sshd_config

- Décommentez le port par défaut « 22 » et indiquez le numéro de port souhaité (minimum 1024) :



```
Port 4444
#AddressFamily an
#ListenAddress 0
```

- Quittez en sauvegardant les modifications faites dans le fichier « sshd_config »
- Relancez le service SSH avec la commande :

systemctl restart ssh

Attention, la connexion en SSH requiert maintenant le port spécifique défini dans le fichier « sshd_config ». Pour vous connecter avec un utilisateur en SSH, vous devrez saisir :

ssh user@ip_serveur -p 4444

ou

ssh user@domaine -p 4444

Remarque :

Sélectionnez un port disponible parmi les plus de 65 000 ports existants. Attention cependant de ne pas utiliser un port inférieur à 1024 car les ports inférieurs sont souvent utilisés pour différents services courants.

4^{ème} étape : installation de Fail2ban sur l'hyperviseur Proxmox

Fail2ban est un framework de prévention contre les intrusions, écrit en Python. Pour l'installer depuis le shell de l'hyperviseur, effectuez les manipulations suivantes :

1 – Installer Fail2ban sur l'hyperviseur (depuis le shell)

```
apt update
apt upgrade -y
apt install fail2ban -y
```

2 – Copier le fichier modèle "jail.conf" en "jail.local"

```
cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
```

3 – Éditer le fichier "jail.local" et ajouter les éléments donnés ci-dessous :

```
nano /etc/fail2ban/jail.local
```

Éléments à ajouter dans le fichier "jail.local", puis quitter en sauvegardant les modifications :

[proxmox]

```
enabled = true
port = https,http,8006
filter = proxmox
logpath = journal
backend = systemd
maxretry = 3
findtime = 2d
bantime = 1h
```

[sshd]

```
enabled = true
port = ssh
filter = sshd
logpath = journal
backend = systemd
maxretry = 2
findtime = 300
banaction = iptables-allports
bantime = 86400
ignoreip = 127.0.0.1
```

4 – Configurer le filtre en éditant le fichier "proxmox.conf" et en ajoutant les éléments donnés :

```
nano /etc/fail2ban/filter.d/proxmox.conf
```

Éléments à ajouter au fichier :

[Definition]

```
failregex = pvedaemon\[.*authentication failure; rhost=<HOST> user=.* msg=.*
ignoreregex =
```

5 – Redémarrer Fail2ban et vérifier le statut

```
systemctl restart fail2ban
systemctl status fail2ban
```

COMMANDES UTILES FAIL2BAN

Bannir une IP

```
fail2ban-client set [nom du jail] banip [IP à bannir]
```

Enlever le ban d'une IP

```
fail2ban-client set [nom du jail] unbanip [IP concerné]
```

Lister les règles

```
fail2ban-client status
```

Status

```
| - Number of jail:  1
` - Jail list:  sshd
```

Afficher les détails d'une règle

```
fail2ban-client status sshd
```

Status for the jail: sshd

```
| - Filter
| | - Currently failed: 0
| | - Total failed:  5
| ` - File list:  /var/log/auth.log
` - Actions
  | - Currently banned: 1
  | - Total banned:  1
  ` - Banned IP list: 192.168.1.21
```

Lister les tentatives de connexion

```
tail /var/log/auth.log
tail -f /var/log/auth.log
```

Si nécessaire créer le fichier auth.log avec droits 640 :

```
touch /var/log/auth.log
chmod 640 /var/log/auth.log
```

Si les adresses IPv6 ne sont pas gérées, la désactivation se fait au niveau du groupe **[Définitions]** du fichier « fail2ban.conf » :

```
nano /etc/fail2ban/fail2ban.conf
```

* Décommenter "allowipv6"

* Ajouter le paramètre "no"

```
# Option: allowipv6
# Notes.: Allows IPv6 interface:
#         Default: auto
# Values: [ auto yes (on, true, 1) no (off, false, 0) ] Default: auto
allowipv6 = no
```

Redémarrer Fail2ban et vérifier le statut (statut « active » sans erreur)

```
systemctl restart fail2ban
systemctl status fail2ban
```

```
● fail2ban.service - Fail2Ban Service
   Loaded: loaded (/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
   Active: active (running) since Mon 2024-03-25 20:05:40 CET; 21h ago
     Docs: man:fail2ban(1)
    Main PID: 1194316 (fail2ban-server)
      Tasks: 7 (limit: 76819)
     Memory: 65.5M
        CPU: 1min 30.503s
    CGroup: /system.slice/fail2ban.service
            └─1194316 /usr/bin/python3 /usr/bin/fail2ban-server -xf start
```

2 – SECURISER SA MACHINE DEBIAN 12 (règles de base)

Comme pour l'hyperviseur Proxmox, on peut modifier le port SSH et désactiver l'accès SSH pour le « root ».

1^{ère} étape : désactivation de l'accès SSH pour le root et modification du port SSH par défaut

Connectez-vous en tant que « root » sur la machine Debian et éditez le fichier « sshd_config » avec la commande :

```
nano /etc/ssh/sshd_config
```

- Modifiez le port SSH par défaut (utilisez un port disponible >1024)
- Assurez-vous que le « **PermitRootLogin** » est bien commenté (#)

```
Port 6666
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#loginGraceTime 2m
#PermitRootLogin prohibit-password
```


- Quittez et sauvegardez le fichier « sshd_config »
- Relancez le service SSH avec la commande :

systemctl restart ssh

2^{ème} étape : installation de « sudo » et ajout d'un utilisateur au groupe « sudo »

Afficher les utilisateurs du système :

cat /etc/group

```
ssl-cert:x:112:  
atedi:x:1000:  
systemd-coredump:x:999:  
mysql:x:113:  
sgx:x:114:
```

Afficher les groupes auxquels appartient un utilisateur :

groups nom_user

```
root@debian-atedi:~# groups atedi  
atedi : atedi cdrom floppy sudo audio dip video plugdev netdev
```

Installer « sudo » :

apt install sudo -y

Affecter un utilisateur au groupe « sudo » :

usermod -aG sudo nom_user

Dorénavant, l'utilisateur « sudoer » pourra exécuter des commandes avec des privilèges « root ». Il suffira d'ajouter « sudo » devant la commande (le mot de passe de l'utilisateur sudo sera à saisir une fois).

3^{ème} étape : installation de Fail2ban sur Debian 12

Fail2ban est un framework de prévention contre les intrusions, écrit en Python. Pour l'installer depuis le shell de la machine Debian 12, effectuez les manipulations suivantes :

1 – Installer Fail2ban sur Debian 12 (depuis la console)

apt update

apt upgrade -y

apt install fail2ban -y

2 – Copier le fichier modèle "jail.conf" en "jail.local"

cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local

3 – Éditer le fichier "jail.local" et ajouter les éléments donnés ci-dessous :

nano /etc/fail2ban/jail.local

Éléments à ajouter dans le fichier "jail.local", puis quitter en sauvegardant les modifications :

[sshd]

```
enabled = true
port = ssh
filter = sshd
logpath = journal
backend = systemd
maxretry = 2
findtime = 300
banaction = iptables-allports
bantime = 86400
ignoreip = 127.0.0.1
```

4 – Redémarrer Fail2ban et vérifier le statut

```
systemctl restart fail2ban
systemctl status fail2ban
```

COMMANDES UTILES FAIL2BAN

Bannir une IP

```
fail2ban-client set [nom du jail] banip [IP à bannir]
```

Enlever le ban d'une IP

```
fail2ban-client set [nom du jail] unbanip [IP concerné]
```

Lister les règles

```
fail2ban-client status
```

Status

```
| - Number of jail: 1
` - Jail list: sshd
```

Afficher les détails d'une règle

```
fail2ban-client status sshd
```

Status for the jail: sshd

```
| - Filter
| | - Currently failed: 0
| | - Total failed: 5
| ` - File list: /var/log/auth.log
` - Actions
| - Currently banned: 1
| - Total banned: 1
` - Banned IP list: 192.168.1.21
```

Lister les tentatives de connexion

```
tail /var/log/auth.log
```

```
tail -f /var/log/auth.log
```

Si nécessaire créer le fichier auth.log avec droits 640 :

```
touch /var/log/auth.log
```

```
chmod 640 /var/log/auth.log
```

Si les adresses IPv6 ne sont pas gérées, la désactivation se fait au niveau du groupe [Définitions] du fichier « fail2ban.conf » :

```
nano /etc/fail2ban/fail2ban.conf
```

* Décommenter "allowipv6"

* Ajouter le paramètre "no"

```
# Option: allowipv6
# Notes.: Allows IPv6 interface:
#         Default: auto
# Values: [ auto yes (on, true, 1) no (off, false, 0) ] Default: auto
allowipv6 = no
```

Redémarrer Fail2ban et vérifier le statut (statut « active » sans erreur)

```
systemctl restart fail2ban
```

```
systemctl status fail2ban
```

```
● fail2ban.service - Fail2Ban Service
   Loaded: loaded (/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
   Active: active (running) since Mon 2024-03-25 20:05:40 CET; 21h ago
     Docs: man:fail2ban(1)
  Main PID: 1194316 (fail2ban-server)
    Tasks: 7 (limit: 76819)
   Memory: 65.5M
      CPU: 1min 30.503s
   CGroup: /system.slice/fail2ban.service
           └─1194316 /usr/bin/python3 /usr/bin/fail2ban-server -xf start
```

4^{ème} étape : modification du mot de passe d'un compte utilisateur Debian

Pensez à sécuriser vos mots de passe (12 caractères au minimum avec des caractères alphanumériques, des symboles, des majuscules).

- Saisissez (en tant que « root » ou utilisateur « sudo ») la commande suivante :

```
sudo passwd nom_user
```

IMPORTANT – PRENEZ LE TEMPS D’AFFINER VOS REGLES DE SECURITE

Ces règles sont des bases à appliquer sur tous systèmes exposés au web. Ne négligez pas ces manipulations au risque de voir votre serveur et vos machines internes corrompues !