DOCUMENTATION TECHNIQUE

Installation et configuration de Fail2Ban

Ayoub MOURKIA

Introduction

Fail2Ban est une solution de cybersécurité open source permettant de protéger un serveur contre les attaques par force brute en analysant les journaux système et applicatifs.

Contrairement à CrowdSec, Fail2Ban repose sur une logique simple basée sur des règles locales appelées "jails" associées à des filtres de logs.

Lorsqu'un comportement suspect est détecté (ex : multiples échecs SSH), Fail2Ban bannit automatiquement l'adresse IP via le pare-feu.

- Détection basée sur les logs
- Blocage automatique des IP malveillantes
- Support de nombreux services : SSH, Nginx, Apache, FTP
- Configuration locale simple et efficace

Architecture de Fail2Ban

- Filters : règles regex pour détecter les attaques
- Jails : configuration liant service + filtre + action
- Actions : bannissement via iptables ou nftables
- Backend : système de lecture des logs (systemd ou fichiers)

Prérequis

- Serveur Linux Debian ou Ubuntu à jour
- Accès root ou sudo
- Connexion Internet
- Logs disponibles (auth.log, nginx, etc.)
- Pare-feu actif (iptables ou nftables)

Recommandations avant installation

- Mettre à jour le système
- Identifier les services à protéger
- Vérifier les chemins de logs
- Ajouter une IP d'administration en allowlist

```
sudo apt update && sudo apt full-upgrade -y
sudo reboot
```

Installation de Fail2Ban

sudo apt install fail2ban -y

sudo systemctl enable fail2ban
sudo systemctl start fail2ban

sudo systemctl status fail2ban

Fichiers importants

/etc/fail2ban/jail.conf -> configuration par défaut
/etc/fail2ban/jail.local -> configuration personnalisée
/etc/fail2ban/filter.d/ -> filtres
/var/log/fail2ban.log -> logs

Configuration principale

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

Configuration SSH complète

[DEFAULT]
bantime = 3600
findtime = 600
maxretry = 5
ignoreip = 127.0.0.1/8 VOTRE_IP
backend = systemd

[sshd]
enabled = true
port = ssh
filter = sshd
logpath = /var/log/auth.log
maxretry = 5

Redémarrage

sudo systemctl restart fail2ban

Vérification

sudo fail2ban-client status
sudo fail2ban-client status sshd

Tests de fonctionnement

Tester brute force SSH :

ssh test@IP_SERVEUR (mauvais mot de passe plusieurs fois)

Puis vérifier :
sudo fail2ban-client status sshd

Gestion des décisions

sudo fail2ban-client set sshd banip 1.2.3.4
sudo fail2ban-client set sshd unbanip 1.2.3.4

Lister les IP bannies :
sudo fail2ban-client status sshd

Protection services web

Exemple Nginx :

[nginx-http-auth]
enabled = true
port = http,https
logpath = /var/log/nginx/error.log

Allowlist / Whitelist

Dans jail.local :

ignoreip = 127.0.0.1/8 192.168.1.0/24 VOTRE_IP

Logs et supervision

tail -f /var/log/fail2ban.log
journalctl -u fail2ban

Maintenance

sudo apt update
sudo apt upgrade -y
sudo systemctl restart fail2ban

Dépannage

Fail2Ban ne démarre pas :
sudo systemctl status fail2ban
sudo journalctl -u fail2ban -xe

Aucun bannissement :
• Vérifier logpath
• Vérifier service actif
• Vérifier backend utilisé

Faux positifs :
• Ajuster maxretry
• Ajouter allowlist

Conclusion

Fail2Ban est une solution simple, rapide et efficace pour protéger un serveur Linux contre les attaques par force brute.

Il est recommandé de l'utiliser en complément d'autres solutions comme CrowdSec pour une sécurité renforcée.