



24/06/2017

---

Fail2ban permet le blocage des adresses IP tentant une intrusion sur votre serveur. Pour cela, il s'appuie sur le parefeu Netfilter ou sur TCP Wrapper.

Il détecte les tentatives d'intrusion en parcourant les journaux du système.

## 1. Installation

Fail2ban est disponible dans le dépôt EPEL.

```
yum install fail2ban
```

Le fichier **/etc/fail2ban/jail.conf** est fourni par le paquet **fail2ban**.

Il ne faut le modifier directement, sous peine de voir ses changements perdus lors de la prochaine mise à jour du paquet. Au lieu de cela, il faut créer un fichier **/etc/fail2ban/jail.local**, et y placer sa configuration personnalisée.

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

## 2. Configuration

Options générales (section DEFAULT) :

- **bantime** : le temps en secondes durant lequel l'adresse IP ayant tentée une intrusion sera bannie ;
- **findtime** : la plage de temps en secondes utilisée pour analyser les logs. Plus cette plage est grande, plus l'analyse est longue ;
- **maxretry** : le nombre d'échecs de connexion tolérés avant le bannissement.

Il est primordiale d'ajuster au mieux ces trois valeurs. Un attaquant, avec la configuration par défaut, peut faire 5 tentatives toutes les 10 minutes, sans être banni. Cette valeur peut paraître ridiculement petite, mais elle représente par jour  $5 \times 6 \times 24 = 720$  tentatives, et 262 800 par an. Elle est encore à multiplier par le nombre de postes participant à l'attaque.



Même avec un système comme fail2ban, un poste n'est pas à l'abri des attaques. Fail2ban n'empêchera pas un attaquant de prendre possession de votre serveur, mais le retardera. Il est important de respecter les règles de bases de la sécurité informatique : changement fréquent de mot de passe, complexité, etc.

```
[root]# vim /etc/fail2ban/jail.local
[DEFAULT]
bantime  = 3600
findtime = 600
maxretry = 5

[ssh-iptables]
enabled  = true
filter   = sshd
action   = iptables[name=SSH, port=ssh, protocol=tcp] sendmail-whois[name=SSH,
dest=root, sender=fail2ban@formatux.fr]
logpath  = /var/log/secure
maxretry = 5
```

- section ssh-iptables :
  - enabled : active la règle
  - filter : fichier de log à analyser. Un chemin complet ou un raccourci (comme c'est le cas ici)
  - action : que dois faire fail2ban en cas d'échec de connexion ?
    - iptables : activer une règle dans le parefeu,
    - sendmail-whois : envoyer un mail de rapport.

### 3. Lancement du service

- Fail2ban s'appuyant sur le firewall netfilter pour bannir les adresses IP tentant une intrusion, il faut s'assurer que celui-ci soit démarré :

```
service iptables status
service ip6tables status
```

- Si le parefeu n'est pas actif sur le système :

```
chkconfig iptables on
chkconfig ip6tables on
service iptables start
service ip6tables start
```

- Démarrer le service fail2ban :

```
chkconfig fail2ban on
service fail2ban start
```

## 4. Vérification du service

La commande **fail2ban-client status** permet d'obtenir des informations sur les services surveillés ainsi que le nombre de règles iptables mises en place :

```
[root]# fail2ban-client status
Status
|- Number of jail:  2
`- Jail list:        mysqld-iptables, ssh-iptables
```

IPtables doit également renvoyer des informations concernant l'utilisation d'une chaîne fail2ban-SSH :

```
[root]# iptables --list

Chain INPUT (policy ACCEPT)
target     prot opt source                destination           tcp dpt:ssh
fail2ban-SSH tcp  --  anywhere              anywhere              state RELATED,ESTABLISHED
ACCEPT     all  --  anywhere              anywhere
...

Chain fail2ban-SSH (1 references)
target     prot opt source                destination
RETURN     all  --  anywhere              anywhere
```

## 5. Interface graphique

**Fail2Web** est une interface graphique web pour Fail2Ban, qui communique via **Fail2Rest** (service REST).



Attention à la configuration du serveur Fail2Rest, qui par défaut ne propose pas d'authentification, il faudra le faire via l'authentification apache.