Protéger son serveur avec Fail2ban

>>> Fail2ban et Debian 8

Description :

Cet article explique comment configurer Fail2ban afin de sécuriser votre serveur. Fail2ban permet de sécuriser plusieurs services comme SSH, Apache, vsftp, proftpd, contre des attaques de types brute-force ou Deny-Of-Services.

Protéger son serveur avec Fail2ban

>>> Fail2ban et Debian 8

Sommaire :

I) Introduction

1) Présentation

Toutes les machines connectées à Internet sont exposées à des risques d'attaques. Fail2ban permet de protéger votre serveur contre des attaques basiques de type brute force et Deny-Of-Services (DOS).

2) Fonctionnement de Fail2ban

Fail2Ban est un logiciel permettant d'analyser des fichiers de logs et de déclencher des actions si une attaque est détectée. Fail2ban étant développé en langage Python, il est très modulaire au niveau des mécanismes de détections mais aussi sur les actions à mener.

Pour résumer, Fail2ban analyse les logs de votre serveur lorsqu'il détecte plusieurs tentatives de connexion échoué il va mettre en place des actions que vous aurez définit comme par exemple bloquer l'adresse IP.

Fail2Ban se base sur un système de prisons (jails) que l'on peut définir, activer ou désactiver dans un simple fichier de configuration (/etc/fail2ban/jail.conf).
Une prison (jail) est composée, entre autres, des éléments suivants :
Nom du fichier de log à analyser.
Filtre à appliquer sur ce fichier de log (la liste des filtres disponibles se trouve dans le répertoire /etc/fail2ban/filter.d). Il est bien sûr possible de créer ses propres filtres.
Paramètres permettant de définir si une action doit être déclenchée quand le filtre correspond ("match") : Nombre de "matchs" (maxretry), intervalle de temps correspondant (findtime)...
Action à mener si nécessaire. La liste des actions se trouve dans le répertoire /etc/fail2ban/action.d. Il est également possible de créer ses propres actions.

II) Installation

On installe le service :

```
aptitude install fail2ban
```

Voilà Fail2ban est installé.

III) Configuration

Dans le répertoire **"/etc/fail2ban/filter.d/"** vous pourrez trouver tous les tests que vous pouvez mettre en place. Bien sûr vous pouvez aussi créer vos propres fichiers "filter".

- Affichez le contenu du répertoire **"/etc/fail2ban/filter.d/"** :

```
ls -l /etc/fail2ban/filter.d/
```

- Vous devez obtenir ceci :

```
-rw-r--r-- 1 root root  442 mars  15  2014 3proxy.conf
-rw-r--r-- 1 root root 3233 mars  15  2014 apache-auth.conf
-rw-r--r-- 1 root root 2736 mars  15  2014 apache-badbots.conf
-rw-r--r-- 1 root root  818 mars  15  2014 apache-common.conf
-rw-r--r-- 1 root root  402 mars  15  2014 apache-modsecurity.conf
-rw-r--r-- 1 root root  596 mars  15  2014 apache-nohome.conf
-rw-r--r-- 1 root root  778 mars  15  2014 apache-noscript.conf
-rw-r--r-- 1 root root 2000 mars  15  2014 apache-overflows.conf
-rw-r--r-- 1 root root 1156 mars  15  2014 assp.conf
-rw-r--r-- 1 root root 2270 mars  15  2014 asterisk.conf
-rw-r--r-- 1 root root 1671 mars  15  2014 common.conf
-rw-r--r-- 1 root root  393 mars  15  2014 courierlogin.conf
-rw-r--r-- 1 root root  352 mars  15  2014 couriersmtp.conf
-rw-r--r-- 1 root root  418 mars  15  2014 cyrus-imap.conf
-rw-r--r-- 1 root root 1386 mars  15  2014 dovecot.conf
-rw-r--r-- 1 root root 1696 mars  15  2014 dropbear.conf
-rw-r--r-- 1 root root  767 mars  15  2014 ejabberd-auth.conf
-rw-r--r-- 1 root root  403 mars  15  2014 exim-common.conf
-rw-r--r-- 1 root root 1349 mars  15  2014 exim.conf
-rw-r--r-- 1 root root  983 mars  15  2014 exim-spam.conf
-rw-r--r-- 1 root root  942 mars  15  2014 freeswitch.conf
-rw-r--r-- 1 root root  223 mars  15  2014 groupoffice.conf
-rw-r--r-- 1 root root  322 mars  15  2014 gssftpd.conf
-rw-r--r-- 1 root root  404 mars  15  2014 horde.conf
-rw-r--r-- 1 root root  323 mars  15  2014 lighttpd-auth.conf
-rw-r--r-- 1 root root  886 mars  15  2014 mysqld-auth.conf
-rw-r--r-- 1 root root  400 mars  15  2014 nagios.conf
-rw-r--r-- 1 root root 1693 mars  15  2014 named-refused.conf
-rw-r--r-- 1 root root  422 mars  15  2014 nginx-http-auth.conf
-rw-r--r-- 1 root root  701 mars  15  2014 nsd.conf
-rw-r--r-- 1 root root  495 mars  15  2014 openwebmail.conf
-rw-r--r-- 1 root root  808 mars  15  2014 pam-generic.conf
-rw-r--r-- 1 root root  568 mars  15  2014 perdition.conf
-rw-r--r-- 1 root root  834 mars  15  2014 php-url-fopen.conf
-rw-r--r-- 1 root root  691 mars  15  2014 postfix.conf
-rw-r--r-- 1 root root  312 mars  15  2014 postfix-sasl.conf
-rw-r--r-- 1 root root 1054 mars  15  2014 proftpd.conf
-rw-r--r-- 1 root root 1725 mars  15  2014 pure-ftpd.conf
-rw-r--r-- 1 root root  795 mars  15  2014 qmail.conf
-rw-r--r-- 1 root root 1213 mars  15  2014 recidive.conf
-rw-r--r-- 1 root root  907 mars  15  2014 roundcube-auth.conf
-rw-r--r-- 1 root root  517 mars  15  2014 selinux-common.conf
-rw-r--r-- 1 root root  570 mars  15  2014 selinux-ssh.conf
```

```
-rw-r--r-- 1 root root  330 mars  15  2014 sendmail-auth.conf
-rw-r--r-- 1 root root 1665 mars  15  2014 sendmail-reject.conf
-rw-r--r-- 1 root root  371 mars  15  2014 sieve.conf
-rw-r--r-- 1 root root  472 mars  15  2014 sogo-auth.conf
-rw-r--r-- 1 root root 1093 mars  15  2014 solid-pop3d.conf
-rw-r--r-- 1 root root  193 mars  15  2014 squid.conf
-rw-r--r-- 1 root root 1816 mars  15  2014 sshd.conf
-rw-r--r-- 1 root root  697 mars  15  2014 sshd-ddos.conf
-rw-r--r-- 1 root root  645 mars  15  2014 suhosin.conf
-rw-r--r-- 1 root root  374 mars  15  2014 uwimap-auth.conf
-rw-r--r-- 1 root root  621 mars  15  2014 vsftpd.conf
-rw-r--r-- 1 root root  444 mars  15  2014 webmin-auth.conf
-rw-r--r-- 1 root root  514 mars  15  2014 wuftpd.conf
-rw-r--r-- 1 root root  503 mars  15  2014 xinetd-fail.conf
```

1) Protection de SSH

a) Explications

Le premier service que nous allons protéger avec Fail2ban et l'accès SSH de notre serveur. Nous allons configurer Fail2ban pour qu'il détecte 4 échecs à la suite de la saisie Login/password. Lors de la détection, il bloquera l'adresse IP de la machine cliente pendant 3 minutes.

Afin détecter les échecs d'authentification, Fail2ban va analyser le fichier **"/var/log/auth.log"** en utilisant le filtre **"/etc/fai2ban/filter.d/sshd.conf"**. Il appliquera l'action par défaut.

Information : La protection SSH est la seule protection activée par défaut après l'installation.

b) Configuration

- Pour mettre en place cette protection nous allons éditer le fichier **"/etc/fail2ban/jail.conf"**.

```
vim /etc/fail2ban/jail.conf
```

- Rechercher les paramètres **[ssh]**

```
[ssh]

enabled  = true
port     = ssh
filter   = sshd
logpath  = /var/log/auth.log
maxretry = 6
```

- Modifiez les paramètres comme ceci :

```
[ssh]

enabled  = true
port     = ssh
filter   = sshd
logpath  = /var/log/auth.log
maxretry = 4
bantime = 180
```

- On active aussi la détection d'attaque DOS sur le protocole SSH. Recherchez les paramètres **[ssh-ddos]**

```
[ssh-ddos]
```

```
enabled  = false
port     = ssh
filter   = sshd-ddos
logpath  = /var/log/auth.log
maxretry = 6
```

- Modifiez les paramètres.

```
[ssh-ddos]

enabled  = true
port     = ssh
filter   = sshd-ddos
logpath  = /var/log/auth.log
maxretry = 4
```

- On redémarre Fail2ban pour la prise en compte des paramètres.

```
service fail2ban restart
```

c) Tests

- Depuis un client réalisez 4 échecs d'authentifications. Puis affichez les règles IPtables.

```
iptables -L
```

- Vous devez obtenir ceci :

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
fail2ban-ssh-ddos  tcp  --  anywhere             anywhere             multiport dports ssh
fail2ban-ssh  tcp  --  anywhere             anywhere             multiport dports ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain fail2ban-ssh (1 references)
target     prot opt source               destination
REJECT     all  --  client-failban.home  anywhere             reject-with icmp-port-unreachable
RETURN     all  --  anywhere             anywhere

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

- Si vous retentez une connexion SSH avec le client, vous pourrez constater que la connexion est refusée par le serveur.

- Attendez 3 minutes et affichez de nouveau les règles IPtables. Vous pourrez constater que le client n'est plus bloqué.

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
fail2ban-ssh-ddos  tcp  --  anywhere             anywhere             multiport dports ssh
fail2ban-ssh  tcp  --  anywhere             anywhere             multiport dports ssh
```

```
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

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

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

2) Protection de Apache2

Fail2Ban propose plusieurs sécurités pour protéger votre serveur Apache. Voici les filtres à mettre en place si vous voulez protéger votre serveur.

a) Filtres existants

- Activer les modules suivants en définissant la valeur du paramètre **"enabled"** à **"true"**.

- La première sécurité permet de détecter les échecs d'authentifications.

```
[apache]

enabled  = true
port     = http,https
filter   = apache-auth
logpath  = /var/log/apache*/*error.log
maxretry = 4
```

- Activez aussi la sécurité **"apache-multiport"**.

```
[apache-multiport]

enabled   = false
port      = http,https
filter    = apache-auth
logpath   = /var/log/apache*/*error.log
maxretry  = 6
```

- La sécurité **"apache-noscript"** permet de détecter les attaques de type "Exploit" ou des attaques sur des vulnérabilités PHP.

```
[apache-noscript]

enabled  = true
port     = http,https
filter   = apache-noscript
logpath  = /var/log/apache*/*error.log
maxretry = 6
```

- La sécurité **"apache-overflows"** permet de détecter les attaques de type "overflow" (surcharge) du service apache.

```
[apache-overflows]

enabled  = true
port     = http,https
filter   = apache-overflows
```

```
logpath  = /var/log/apache*/*error.log
maxretry = 2
```

- La sécurité **"apache-nohome"** permet de détecter les tentatives de recherche d'un répertoire "Home" sur le serveur.

```
[apache-nohome]

enabled  = true
filter   = apache-nohome
port     = http,https
logpath  = /var/log/apache*/*error.log
maxretry = 2
```

- La sécurité **"Apache-modsecurity"** permet d'analyser les logs générer par le module "security" d'Apache.

[apache-modsecurity]

enabled = true
filter = apache-modsecurity
port = http,https
logpath = /var/log/apache*/*error.log
maxretry = 2

- La sécurité **"apache-badbots"** permet de détecter et bloquer certain "Bots" connus.

```
[apache-badbots]
enabled  = true
port     = http,https
filter   = apache-badbots
logpath  = /var/log/httpd/*access_log
bantime  = 172800
maxretry = 1
```

- La sécurité **php-url-fopen** permet de détecter et bloquer les attaques via des scripts PHP voulant exécuter des URLs.

```
[php-url-fopen]

enabled = false
port    = http,https
filter  = php-url-fopen
logpath = /var/www/*/logs/access_log
```

b) Ajout de filtres personnalisés

Nous allons créer trois modules personnalisés :

- Le filtre **"apache-404"** qui permet de bannir les utilisateurs qui font trop d'erreurs 404. Généralement ce sont ceux qui recherchent des pages d'administration en modifiant aléatoirement l'URL.
- Le filtre **"apache-w00tw00t"** qui permet de bannir un scanner de faille « w00tw00t » fréquemment utilisé, on en retrouve la trace dans les logs.
- Le filtre **"apache-admin"** permet de protéger votre espace d'administration si vous en avez un.

– Commencez par ajouter les lignes suivantes dans le fichier **"jail.conf"** :

```
[apache-404]

enabled = true
port = http
filter = apache-404
logpath = /var/log/apache*/access*.log
maxretry = 3

[apache-w00tw00t]
enabled = true
filter = apache-w00tw00t
action = iptables[name=Apache-w00tw00t,port=80,protocol=tcp]
logpath = /var/log/apache*/access*.log
maxretry = 1

[apache-admin]

enabled = true
port = http
filter = apache-admin
logpath = /var/log/apache*/error*.log
maxretry = 6
```

- Créez le fichiers "filter" pour "apache-404" dans le répertoire **"/etc/fail2ban/filter.d/"**

```
vim /etc/fail2ban/filter.d/apache-404.conf
```

- Ajoutez les lignes suivantes :

```
# Fail2Ban configuration file
#
#
#

[Definition]

# Option:  failregex
#
# Values:  TEXT
#
failregex = <HOST> - - \[.*?\] ".*?" 404
# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
#
ignoreregex =
```

- Créez le fichier "filter" pour "apache-w00tw00t" dans le répertoire **"/etc/fail2ban/filter.d/"**

```
vim /etc/fail2ban/filter.d/apache-w00tw00t.conf
```

- Ajoutez les lignes suivantes :

```
[Definition]

# Option:  failregex
#
# Values:  TEXT
#
failregex = ^<HOST> -.*"GET \/w00tw00t\.at\.ISC\.SANS\.DFind\:\).*".*
# Option:  ignoreregex
# Notes.:  regex to ignore. If this regex matches, the line is ignored.
# Values:  TEXT
#
```

```
ignoreregex =
```

- Créez le fichier "filter" pour "apache-admin" dans le répertoire **"/etc/fail2ban/filter.d/"**

```
vim /etc/fail2ban/filter.d/apache-admin.conf
```

- Ajoutez les lignes suivantes :

```
[Definition]

# Option: failregex
# Notes.: regex to match the password failure messages in the logfile. The
# host must be matched by a group named "host". The tag "<HOST>" can
# be used for standard IP/hostname matching.
#

failregex = [[]client <HOST>[]] File does not exist: .*admin|PMA|mysql

# Option: ignoreregex
# Notes.: regex to ignore. If this regex matches, the line is ignored.
# Values: TEXT
#
ignoreregex =
```

- Redémarrez le service Fail2ban.

```
service fail2ban restart
```

c) Tester les filtres

Vous pouvez tester vos filtres avec la commande suivante (à adapter suivant les filtres).

```
fail2ban-regex /var/log/apache2/access.log /etc/fail2ban/filter.d/apache-404.conf
```

3) Testez l’état des jails

Avec la commande **"fail2ban-client"** vous pouvez vérifier l’état de vos **"jails"** :

- Tapez la commande suivante :

```
fail2ban-client status
```

- Vous devez obtenir ceci :

```
Status
|- Number of jail:      9
<span class="base64"
title="PGNvZGUgY2xhc3M9InNwaXBfY29kZSBzcGlwX2NvZGVfaW5saW5lIiBkaXI9Imx0ciI+LSBKYWlsIGxpc3Q6ICZuYnNwOyAmbmJzcDsgJm5ic3A7ICZ
uYnNwOyAmbmJzcDsgYXBhY2hlLXcwMHR3MDB0LCBhcGFjaGUtYWRtaW4sIGFwYWNoZS1ub3NjcmlwdCwgYXBhY2hlLW5vaG9tZSwgc3NoLWRkb3MsIGFwYWNoZ
S1tdWx0aXBvcnQsIHNzaCwgYXBhY2hlLTQwNCwgYXBhY2hlCiZsdDsvY29kZSZndDsKClRvdWpvdXJzIGF2ZWMgbGEgY29tbWFuZGUge3smcXVvdDtmYWlsMmJ
hbi1jbGllbnQmcXVvdDt9fSwgdm91cyBwb3V2ZXogdsOpcmlmaWVyIGwnw6l0YXQgZCd1biB7eyZxdW90O2phaWxzJnF1b3Q7fX0gcGFydGljdWxpZXIuCgotI
FRhcGV6IGxhIGNvbW1hbmRlIHN1aXZhbnRlIDoKCiZsdDtjb2RlJmd0OwpmYWlsMmJhbi1jbGllbnQgc3RhdHVzIGFwYWNoZS00MDQKJmx0Oy9jb2RlJmd0Owo
KLSBWb3VzIGRldmV6IG9idGVuaXIgY2VjaSA6CgombHQ7Y29kZSZndDsKU3RhdHVzIGZvciB0aGUgamFpbDogYXBhY2hlLTQwNAp8LSBmaWx0ZXIKfCAmbmJzc
```

```
Dt8LSBGaWxlIGxpc3Q6ICZuYnNwOyAmbmJzcDsgJm5ic3A7ICZuYnNwOy92YXIvbG9nL2FwYWNoZTIvYWNjZXNzLmxvZwp8ICZuYnNwO3wtIEN1cnJlbnRseSBmYWlsZWQ6IDAKfDwvY29kZT4="></span>- Total failed:     0
<span class="base64"
title="PGNvZGUgY2xhc3M9InNwaXBfY29kZSBzcGlwX2NvZGVfaW5saW5lIiBkaXI9Imx0ciI+LSBhY3Rpb24KICZuYnNwOyB8LSBDdXJyZW50bHkgYmFubmVkOiAwCiAmbmJzcDsgfDwvY29kZT4="></span>- IP list:
   `- Total banned:     0
```

4) Configurer l’envoie de mails

Pour recevoir un mail lorsque fail2ban bloque une adresse IP, vous devez éditez le fichier **"jail.conf"**.

```
vim /etc/fail2ban/jail.conf
```

- Modifiez le paramètre **"destemail"** par l’adresse du destinataire.

```
[DEFAULT]
destemail = root@localhost
```

- Modifiez le paramètre **"action"** afin d’ajouter l’envoi de mail dans les actions par défaut.

- Pour envoyer un mail contenant le whois, placez la variable sur :

```
action = %(action_mw)s
```

- Pour envoyer un mail avec le whois ET les logs, placez la variable sur :

```
action = %(action_mwl)s
```

IV) Autoriser une adresse bannis

Si vous vous êtes bannis par erreur, vous pouvez tapez la commande suivant pour supprimer le blocage :

```
fail2ban-client set <jail> unbanip <ip>
```

- : Correspond au filtre qui vous a bannis par exemple **"ssh"**.
- : Correspond à l’adresse IP bloqué.

D’autres options sont possibles, voici quelques arguments :

- **start** => Démarrer le service
- **reload** => Recharger tous les filtres
- **reload [filtre]** => Recharger un filtre
- **stop** => Arrêter le service
- **status** => Voir le statut du service

Idum