# SECURISATION D'UN SERVEUR APACHE

BERETE IBRAHIMA

DOUCOURE SAMBOU

MANTELLATO XAVIER

# SOMMAIRE

# Limitations contre les DOS

- De façon à limiter la portée des attaques de type Denial of Service, il est conseillé de limiter le nombre de connexions simultanées **MaxClients** et en particulier le nombre de connexions persistantes **MaxKeepAliveRequests.** Celles-ci sont apparues avec la norme HTTP 1.1. Elles permettent d'effectuer des requêtes successives lors de la même connexion, ce qui augmente les performances du serveur.

- **Exemple pour  un petit serveur:**

  - MaxClients 150
  - KeepAlive On
  - MaxKeepAliveRequests 100
  - KeepAliveTimeOut 5
  -

# Virtual Host

- Apache  permet la définition de Virtual Host, c'est-à-dire que le même serveur peut héberger, y compris sur une même adresse IP, plusieurs sites différenciés par leur nom. Pour limiter les risques liés à une panne des serveurs DNS ou à des manipulations frauduleuses, il convient de définir le VirtualHost par une adresse IP puis de préciser son nom.

- **Exemple :**

- <VirtualHost 194.57.201.103>
- ServerName www.esiea.fr ….
- </VirtualHost>
-

# Mesure défensive

- Plus sérieusement, il est fortement conseillé de tout interdire par défaut :

- <Directory />

Order deny, allow

Deny

- </Directory >

- Ensuite, il ne reste qu'à valider l'accès aux répertoires correspondant aux sites

- **Order** Indique dans quel ordre les directives **deny** et **allow** sont évaluées. **Deny from all** interdit l'accès depuis partout. On aurait pu indiquer un nom de machine, un nom de domaine, une adresse IP, un couple IP/masque de réseau.

# PHP

- PHP a été conçu spécifiquement pour la programmation web en tenant compte d'impératif de sécurité. Il est grandement paramétrable. Il est conseillé de définir les options suivantes dans son fichier de configuration /etc/php.ini

- Safe_mode = on
- Expose_php = off
- Max_execution_time = 30 ; Maximum execution time of each script, in seconde
- Memory_quotes_gcp = On
- Display_ errors = Off
- Sql.safe_mode = On

- **Safe_mode** surveille les fichiers accédés et interdit l'usage de commande à risque, **expose_php** contrôle l'affichage de sa banner, **max_execution_time** et **mermory_limit** limite les risques de Dos, **magic_quotes_gpc** ajoute des quotes pour les données reçues des GET/POST et des cookies.

# FAIL2BAN

- **Fail2Ban permet de bannir des adresses IP automatiquement si elles ne respectent pas les règles qu'on définit au préalable.**
- **Installation**
  - Pour installer fail2ban, on utilise le gestionnaire de paquets

```
sudo apt-get install fail2ban
```

## Valeurs par défaut:

- On change les valeurs par défaut en fonction de nos besoins, ces valeurs se trouvent dans section **[DEFAULT].**
- Le premier attribut est **ignoreip**. Fail2Ban ne bannira aucune adresse dans cette liste **,** chaque adresse IP est séparé par un espace. On ajoute l'adresse IP de notre machine distant afin de ne pas nous faire bannir.

```
ignoreip = 127.0.0.1/8 192.168.1.43
```

- Le seconde attribut est le **bantime**, elle définit le nombre de secondes que le bannissement d'un IP doit durer. La valeur par défaut est de 600 secondes(10min), on va l'augmenter à 1800 secondes(30 minutes).

```
bantime = 1800
```

- Le troisième attribut définit en secondes l'étendue des lignes de logs à utiliser pour déterminer si le maxretry pour l'authentification a été déclencher.

```
finetime = 1800
```

```
maxretry = 6
```

## Notifications par mail

- Il est possible de recevoir un mail quand un bannissement prend place. Pour ce faire il faut avoir mis en place le Mail Transfer Agent  MTA à l'aide de Postfix au autres. Après avoir mis en place MTA il faut le dire à fail2ban en modifiant toujours dans la section[**DEFAULT**], l'attribut mta.

```
mta = sendmail
```

- Par la suite il faut préciser le mail vers lequel seront envoyés les mails de notification.

```
destmail = root@localhost
```

- Et le nom de l'expéditeur.

```
sender = Fail2Ban_Server01
```

- Puis **action_mwl** pour bannir le client suspect et envoyer un mail avec un rapport "whois" et les lignes de log.

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

# Configurer Fail2Ban pour Apache

- Pour se faire nous aurons à modifier cinq jails [**apache**],[**apache-noscript**],[**apache-overflows**],[**apache-badbots**] et [**apache-dos**].
- Jail**[apache]**

```
/apache
```

- Insérer  les lignes ci-dessous sans dupliquer le tag**[apache].**

```
[apache]
enabled = true
port = http,https
filter = apache-auth
logpath = /var/log/apache2/*error.log
bantime = 600
maxretry = 3
findtime = 600
ignoreip = 192.168.1.43  #<---Votre IP pour ne pas vous bannir
```

- Jail[apache-noscript]

```
/apache-noscript
```

- Insérer les lignes ci-dessous sans dupliquer le tag[apache-noscript]

```
[apache-noscript]
enabled = true
port = http,https
filter = apache-noscript
logpath = /var/log/apache2/*error.log
bantime = 600
maxretry = 3
findtime = 600
ignoreip = 192.168.1.43  #<---Votre IP pour ne pas vous bannir
```

- Jail[apache-overflows]

```
/apache-overflows
```

- Insérer les lignes ci-dessous sans dupliquer le tag**[apache-overflows]**.

```
[apache-overflows]
enabled = true
port = http,https
filter = apache-overflows
logpath = /var/log/apache2/*error.log
bantime = 600
maxretry = 2
findtime = 600
ignoreip = 192.168.1.43 #<---Votre IP pour ne pas vous bannir
```

- **Jail [apache-badbots]**

```
/apache-badbots
```

- Insérer les lignes ci-dessous sans dupliquer le tag **[apache-badbots]**.

```
[apache-badbots]
enabled = true
port = http,https
filter = apache-badbots
logpath = /var/log/apache2/*error.log
bantime = 600
maxretry = 2
findtime = 600
ignoreip = 192.168.1.43 #<---Votre IP pour ne pas vous bannir
```

- Jail**[apache-dos]**

```
/apache-dos
```

- Insérer les lignes ci-dessous sans dupliquer le tag **[apache-dos]**.

```
[apache-dos]
enabled = true
port = http,https
filter = apache-dos
logpath = /var/log/apache2/access.log
bantime = 600
maxretry = 300
findtime = 300
ignoreip = 192.168.1.43 #<---Votre IP pour ne pas vous bannir
action = iptables[name=HTTP, port=http, protocol=tcp]
```

- Filter [apache-dos]
  - créer le fichier filtre pour le jail[apache-dos]

```
root@ubuntu:~# vim /etc/fail2ban/filters.d/apache-dos.conf
```

- Inssérer les lignes ci-dessous.

```
[Definition]
failregex = ^<HOST> -.*"(GET|POST).*
ignoreregex =
```

# Test de Fail2Ban

- Maintenant que la configuration est terminée, nous allons redémarrer le service Fail2Ban. Si tout a bien été configuré il ne devrait pas y avoir d'erreurs.
- Vérifier si Fail2Ban a bien accès aux logs. Ctrl+c pour annuler le **tail**

```
root@ubuntu:~# tail -f /var/log/fail2ban.log
```

- Utiliser Apache Benchmark ou votre Booter préférer depuis une autre machine sur le réseau pour simuler une attaque DOS sur votre serveur. Ici Apache Benchmark avec 1000 requêtes et 20 threads sur le domaine ou l'adresse IP.

```
root@ubuntu:~# ab -n 1000 -c 20 192.168.213.129
```

- À la fin du benchmark, refaites la commande tail précédent si vous avez annulé. Le résultat est le suivant (IP dynamique):

```
2016-10-14 23:16:36,916 fail2ban.actions: WARNING [apache-dos] Ban 109.62.17.88
```

```
2016-10-14 23:21:37,150 fail2ban.actions: WARNING [apache-dos] Unban 109.62.17.88
```

# SOURCES

- https://www.supinfo.com/articles/single/2660-proteger-votre-vps-apache-avec-fail2ban
- https://www.cgsecurity.org/Articles/apache.html