



## Installation - Sécurisation Apache2 sur une Debian Sarge .

### - Installation des packages :

```
apt-get install apache2
```

```
apt-get install openssl
```

```
apt-get install php4-cli
```

```
ap-get install php4-mysql
```

```
apt-get install libapache2-mod-security
```

```
apt-get install awstats
```

**contrôle** : dans un navigateur verifier si apache tourne (<http://localhost>)

### Création du certificat pour le mode SSL

```
apache2-ssl-certificate
```

répondre aux questions qui sont posées

**contrôle** : vérifier la création du fichier `/etc/apache2/apcahe.pem` et d'un lien symbolique sur ce fichier. Ce fichier contient la clé publique et la clé privée qui seront utilisées par apache. Il existe une autre méthode plus complexe avec openssl pour créer ces clés.

### Activation du mode SSL

- ▶ Editer le fichier `/etc/apache2/httpd.conf` et ajouter les lignes ci-dessous

```
SSLProtocol -all +SSLv2  
SSLCipherSuite SSLv2:+HIGH:+MEDIUM:+LOW:+EXP  
SSLCertificateFile /etc/apache2/ssl/apache.pem
```

- ▶ Editer le fichier `/etc/apache2/ports.conf` et ajouter si elle n'existe pas déjà la ligne ci-dessous

```
Listen 443
```

### Configuration des sites par défaut pour fonctionner en mod ssl.

```
cp /etc/apache2/sites-available/default cp  
/etc/apache2/sites-available/default-ssl
```

- ▶ Editer le fichier `/etc/apache2/sites-available/default-ssl` et modifier/ajouter comme ci

Rubrique : [Installation](#)

Le : samedi 24 décembre 2005

Par : Laurent Deschaumes

Visites : 121

dessous(Ajout de la ligne « `SSLEngine on` » et ajout de « `:443` » après « `NameVirtualHost *` » et « `VirtualHost *` » )

```
NameVirtualHost *:443

<VirtualHost *:443>

    SSLEngine on

...

...
```

- ▶ Activer le site avec la commande suivante

```
a2ensite default-ssl
```

- ▣ Activer le mod ssl d'apache avec la commande suivante

```
a2enmod ssl
```

- ▶ Forcer Apache à relire son fichier de configuration

```
/etc/init.d/apache2 reload
```

**contrôle :** dans un navigateur verifier si apache tourne en mode ssl <https://localhost>

- ▶ Pour créer une partie du site (chemin physique `/var/www/secret`) uniquement accessible en https editer le fichier `/etc/apache2/sites-available/default-ssl` et ajouter entre les tags `VirtualHost` :

```
<VirtualHost *:443>

.....

Alias /secret "/var/www/secret/"

    Directory "/var/www/secret/">
        AllowOverride None
        Order deny,allow
        Allow from all
    </Directory>

...

...

</VirtualHost>
```

- ▣ Forcer Apache à relire son fichier de configuration

```
/etc/init.d/apache2 reload
```

**contrôle :** dans un navigateur verifier si <https://localhost/secret> fonctionne, et <http://localhost/secret> ne fonctionne pas .

## Activation/désactivation des modules d'Apache

- ▶ Pour voir les modules disponibles tapez la commande

```
a2enmod
```

puis pour activer le module

```
a2enmod <nom_module>
```

- ▶ Pour voir les modules chargés tapez la commande

```
a2dismod
```

puis pour désactiver le module

```
a2dismod <nom_module>
```

- Forcer Apache à relire son fichier de configuration

```
/etc/init.d/apache2 reload
```

## Activation/désactivation site/répertoire virtuel

- ▶ Dans le répertoire /etc/apache2/sites-available créez un fichier de configuration de site dans l'exemple suivant je crée un répertoire virtuel dans le fichier mon\_repertoire\_virtuel :

```
Alias /test/ "/var/www/test/"
<Directory "/var/www/test/">
    Options Indexes MultiViews FollowSymLinks
    AllowOverride None
    Order deny,allow
    Allow from all
</Directory>
```

- Pour activer ce répertoire virtuel tapez la commande

```
a2ensite mon_repertoire_virtuel
```

et forcer Apache à relire son fichier de configuration

```
/etc/init.d/apache2 reload
```

- ▶ Pour désactiver ce répertoire virtuel tapez la commande

```
a2dissite mon_repertoire_virtuel
```

et forcer Apache à relire son fichier de configuration

```
/etc/init.d/apache2 reload
```

Les commandes `a2ensite` et `a2dissite` créent/suppriment des liens symboliques dans le répertoire `/etc/apache2/sites-enabled/`

## Configuration du `mod_security` d'Apache

- Editer le fichier `/etc/apache2/httpd.conf` et rajouter les lignes suivantes

```
# Turn ModSecurity On
SecFilterEngine On

# Reject requests with status 403
SecFilterDefaultAction "deny,log,status:403"

# Some sane defaults
SecFilterScanPOST On
SecFilterCheckURLEncoding On
SecFilterCheckUnicodeEncoding Off

# Accept almost all byte values
SecFilterForceByteRange 1 255

#Injection SQL
SecFilter "delete[[:space:]]+from"
SecFilter "insert[[:space:]]+into"
SecFilter "select.+from"

#Attaque par chemin transversal
SecFilter "\.\.\/"

#Injection javascript
SecFilter "<script"

#Attaques XSS
SecFilter "<.+>"
SecFilter "<[[:space:]]*script"
```

```
#Necessite headers HTTP_USER_AGENT ou HTTP_POST
SecFilterSelective "HTTP_USER_AGENT|HTTP_HOST" "^$"

# Server masking is optional
# SecServerSignature "Microsoft-IIS/5.0"
SecUploadDir /tmp
SecUploadKeepFiles Off

# Only record the interesting stuff
SecAuditEngine RelevantOnly
SecAuditLog logs/audit_log

# You normally won't need debug logging
SecFilterDebugLevel 0
SecFilterDebugLog logs/modsec_debug_log

# Only accept request encodings we know how to handle
# we exclude GET requests from this because some (automated)
# clients supply "text/html" as Content-Type
SecFilterSelective REQUEST_METHOD "!(GET|HEAD)$" chain
SecFilterSelective HTTP_Content-Type \
"!(^application/x-www-form-urlencoded$|^multipart/form-data;)"

# Do not accept GET or HEAD requests with bodies
SecFilterSelective REQUEST_METHOD "^(GET|HEAD)$" chain
SecFilterSelective HTTP_Content-Length "!"^$"

# Require Content-Length to be provided with
# every POST request
```

Pour que tout fonctionne créez le répertoire **logs** dans **/etc/apache2**

Je n'ai pas inventé ces règles le les ai pêchées dans la doc directement sur le site de mod\_security.

► Forcer Apache à relire son fichier de configuration

```
/etc/init.d/apache2 reload
```

Vérifier cycliquement ou mettre sous surveillance le fichier `/etc/apache2/log/audit.log` dans lequel apparaîtront les tentatives d'intrusion.

## Configuration de AWSTATS

Copier le fichier de configuration standard de AWSTATS

```
cp /etc/awstats/awstats.conf.local /etc/awstats.mon_serveur.conf
```

```
Editer le fichier etc/awstats.mon_serveur.conf  
et renseigner les lignes LogFile (fichier de log  
d'apache) et SiteDomain (personnalisation affichage)  
.....  
LogFile="/var/log/apache2/access.log"  
.....  
SiteDomain="Nom de mon serveur"  
.....  
LogFormat=1
```

Il existe bon nombre d'autres options que je vous laisse découvrir

```
Pour vérifier que tout fonctionne lancez la commande  
/usr/lib/cgi-bin/awstats.pl -config= mon_serveur
```

Vous devriez avoir une réponse de ce style

```
Update for config "/etc/awstats/awstats.mon_serveur.conf"  
With data in log file "/var/log/apache2/access.log"...  
Phase 1 : First bypass old records, searching new record...  
Direct access to last remembered record has fallen on another record.  
So searching new records from beginning of log file...  
Phase 2 : Now process new records (Flush history on disk after 20000  
hosts)...  
Jumped lines in file: 0  
Parsed lines in file: 17400  
Found 0 dropped records,  
Found 0 corrupted records,  
Found 0 old records,
```

```
Found 17400 new qualified records.
```

Il n'y a plus qu'à automatiser cette tâche avec crontab

commande

```
crontab -e
```

ajouter la ligne suivante pour une mise à jour des stats tout les jours à minuit

```
0 0 * * * /usr/lib/cgi-bin/awstats.pl -config=mon_site
```

Pour atteindre les stats depuis un butineur

```
http://127.0.0.1/cgi-bin/awstats.pl?config=mon_site->http://127.0.0.1/cgi-bin
```

La configuration web de AWSTATS a été ajoutée au fichier /etc/apache2/httpd.conf

par mesure de sécurité restreindre l'accès web des stats en modifiant la ligne

```
<Directory "/usr/local/awstats/wwwroot">
....
....Allow from all
....
</Directory>
```

en

```
<Directory "/usr/local/awstats/wwwroot">
....
....Allow from 127.0.0.1
....
</Directory>
```

► Forcer Apache à relire son fichier de configuration

```
/etc/init.d/apache2 reload
```