

Introduction à la sécurité TP 5 - Apache - Configuration avancée

Maciej Korczynski & Simon Fernandez

Septembre 2020

Objectifs

- Configuration avancée d'Apache (durcissement et AuthN avec certificat)

Apache – Authentication - SSL

In previous tutorial we saw how to configure a web server with a authentication by password We will see now how to establish an authentication with a certificate.

Client authentication by certificate

In the previous tutorial you have created you own CA. You can create a certificat per client and sign it with your CA. After this you can add these lines to your virtualhost configuration to force the client authentication by certificate :

```
SSLCACertificateFile /etc/apache2/ssl/ca.crt
SSLRequireSSL
SSLVerifyClient require
SSLVerifyDepth 1
```

You need to active SSL on apache :

```
a2enmod ssl
```

Server certificat

You can also create a certificate to enable https on your website and sign it with your CA (or create a self-signed certificate). To configure your website you need to add these line :

```
SSLEngine on
SSLCertificateFile /etc/apache2/ssl/apache.crt
SSLCertificateKeyFile /etc/apache2/ssl/apache.key
SSLProtocol ALL
```

You also need to active SSL mode on apache.

Note : Do not forget to use a correct value for **Common name**. You can edit your `/etc/hosts` to simulate a DNS server.

Bonus : What is `SSLProtocol` ? Give your opinion on the proper configuration of it

Practice

Establish a authentication for one of your website by certificate. You can also active https for one of your website (or for both).

You will find to some advice about the use of SSL in https://www.ssllabs.com/downloads/SSL_TLS_Deployment_Best_Practices_1.3.pdf

Apache – Hardening

mod_security

`mod_security` offers an open-source web application firewall (call WAF) and a intrusion detection and prevention system for web applications.

It is widely used. To use it you need to install the package `libapache2-modsecurity` and to active in apache the `mod mod-security` . As a good configuration pratice, you can follow the OWASP Rule Set.

mod_evasive

`mod_evasive` offers a open-source system to detect and prevent HTTP brute-force attack (DOS/DDOS). Like `mod_security`, it is widely used. To install it you need the package `libapache2-mod-evasive`.

Then create the file `/etc/apache2/mods-available/mod-evasive.conf` and activate the mod `mod-evasive`. An example of `mod-evasive.conf` file :

```
<ifmodule mod_evasive20.c>
    DOSHashTableSize 3097
    DOSPageCount 2
    DOSSiteCount 50
    DOSPageInterval 1
    DOSSiteInterval 1
    DOSBlockingPeriod 10
    DOSLogDir /var/log/mod_evasive
    DOSEmailNotify yourmail@mail.com
    DOSWhitelist 127.0.0.1
</ifmodule>
```

You can specify a command with `DOSSystemCommand` in case of attack. For example block the IP with `iptables`. The problem is that the apache deamon should not have the right to modify `iptables` rules. A better way is to combine `mod_evasive` with `fail2ban`.

Practice :

Enhance the security of your website. For example you can active `mod_security` and `mod_evasive`. Do not hesitate to configure some other features (enable custom apache logging per virtualhost, disable following links, ...)