

Installation d'un serveur Apache2

Introduction :

Dans le cadre du contexte GSB nous avons la nécessité d'installer et de déployer correctement un serveur Apache2 sous linux. Pour cela nous allons utiliser une machine virtuelle munie de Debian 10.5.

Installation :

Téléchargement du paquet d'Apache2 :

Nous procédons au téléchargement du paquet d'apache grâce au paquet ci-dessous.

```
apt-get install apache2 -y
```

Téléchargement des paquets de PHP & MariaDB :

Nous allons maintenant télécharger les paquets de PHP ainsi que du module d'apache liant à PHP.

```
apt install php libapache2-mod-php mariadb-server php-mysql -y
```

Téléchargement des modules PHP les plus connus :

```
apt install php-curl php-gd php-intl php-json php-mbstring php-xml php-zip -y
```

Téléchargement de la documentation d'Apache2 :

```
apt-get install apache2-doc
```

Accès à la documentation : `/usr/share/doc/`

Sauvegarde de la configuration d'Apache2 :

Si nous effectuons une action par mégarde détruisant un fichier de configuration, nous serons bien content de retrouver les fichiers non-modifiés.

```
cp -rp /etc/apache2 /etc/apache2.init
```

Commandes liées au serveur Apache2 :

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

Consulter les fichier d'erreurs d'Apache2 :

```
tail -f /var/log/apache2/error.log
```

Création de l'accès public (accès via l'ip):

Création du fichier où seront stockés les fichier du serveur virtuel web.

```
mkdir /var/www/html/public
```

Suppression de la configuration par défaut.

```
echo "000-default"|a2dissite
```

On renomme le fichier 000-default.conf en public.conf.

```
mv /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/public.conf
```

Nous changeons le contenu de public.conf.

```
<VirtualHost *:80>

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/public

    ErrorLog /var/log/apache2/error.log
    CustomLog /var/log/apache2/access.log combined

</VirtualHost>
```

On active la nouvelle configuration.

```
echo "public"|a2ensite
```

Nous redémarrons le serveur apache.

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

Enfin nous créons le fichier index.html :

(chemin : /var/www/html/public/index.html)

```
<html>
  <body>
    <h1>It works! SIO (PUBLIC)</h1>
    <p>This is the default web page for this server.</p>
  </body>
</html>
```

Pour tester ce premier site virtuel, nous avons simplement à accéder à l'IP de notre serveur sur un navigateur.



Création du premier site (accessible avec un nom de domaine) :

Le nom de domaine sera : site1.nehemie.pod4.sio.lan

N'oubliez pas de créer le nom dans votre serveur DNS !

Création du dossier lié au site1 :

```
mkdir /var/www/html/site1
```

Nous allons copier coller le fichier public.conf pour partir d'une bonne base.

```
cp /etc/apache2/sites-available/public.conf /etc/apache2/sites-available/site1.conf
```

Nous l'éditions comme ceci :

```
<VirtualHost *:80>
    ServerName site1.nehemie.pod4.sio.lan
    ServerAlias www.site1.nehemie.pod4.sio.lan
    DocumentRoot /var/www/html/site1

    ErrorLog /var/log/apache2/error.log
    CustomLog /var/log/apache2/access.log combined
```

```
</VirtualHost>
```

Nous activons le site :

```
echo "site1"|a2ensite
```

Nous redémarrons le serveur apache.

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

Nous créons le fichier index.html du site 1 :

(chemin : /var/www/html/site1/index.html)

```
<html>
  <body>
    <h1>It works! SIO (SITE1)</h1>
    <p>This is the default web page for this server.</p>
  </body>
</html>
```

Création d'un second site :

Création du dossier lié au site2 :

```
mkdir /var/www/html/site2
```

Nous allons copier coller le fichier site1.conf pour partir d'une bonne base.

```
cp /etc/apache2/sites-available/site1.conf /etc/apache2/sites-available/site2.conf
```

Nous l'éditions comme ceci :

```
<VirtualHost *:80>
    ServerName site2.nehemie.pod4.sio.lan
    ServerAlias www.site2.nehemie.pod4.sio.lan
    DocumentRoot /var/www/html/site2

    ErrorLog /var/log/apache2/error.log
```

```
CustomLog /var/log/apache2/access.log combined
```

```
</VirtualHost>
```

Nous activons le site :

```
echo "site2"|a2ensite
```

Nous redémarrons le serveur apache.

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

Nous créons le fichier index.html du site 2 :

(chemin : /var/www/html/site2/index.html)

```
<html>
  <body>
    <h1>It works! SIO (SITE2)</h1>
    <p>This is the default web page for this server.</p>
  </body>
</html>
```

Création d'un troisième site avec SSL :

Création d'un certificat :

Nous allons générer un certificat qui sera en suite signé par notre professeur qui endossera le rôle d'autorité de certification.

Premièrement nous allons créer le dossier où sera stocké notre certificat.

```
cd /srv ; mkdir ssl ; cd ssl/
```

Nous allons créer une clé RSA de 2048 bits.

```
openssl genrsa -out clefnehemie.key 2048
```

```
root@srv-deb-nem:/srv/ssl# openssl genrsa -out clefnehemie.key 2048
Generating RSA private key, 2048 bit long modulus (2 primes)
.....+++++
....+++++
e is 65537 (0x010001)
root@srv-deb-nem:/srv/ssl# ls
clefnehemie.key
```

Acti
Accé

La commande ls permet d'afficher les fichiers et/ou dossier qui sont présent dans le répertoire courant.

Maintenant que notre clé RSA est créée, nous pouvons générer un certificat.

```
openssl req -new -key clefnehemie.key -out certificatnehemie.csr
```

```
root@srv-deb-nem:/srv/ssl# openssl req -new -key clefnehemie.key -out certificatnehemie.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FR
State or Province Name (full name) [Some-State]:PACA
Locality Name (eg, city) []:TOULON
Organization Name (eg, company) [Internet Widgits Pty Ltd]:LYC_BONA
Organizational Unit Name (eg, section) []:BTS_SIO
Common Name (e.g. server FQDN or YOUR name) []:site3.nehemie.pod4.sio.lan
Email Address []:nehemiebarkia@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:nehemie
An optional company name []:
```

```
root@srv-deb-nem:/srv/ssl# ls
certificatnehemie.csr  clefnehemie.key
root@srv-deb-nem:/srv/ssl#
```

C'est terminé; nous avons bien créer notre certificat. Nous avons donc besoin maintenant de le faire certifier par une autorité de certification (dans notre cas simulé par notre professeur). Nous allons donc lui donner par mail le certificat et ainsi, ils nous enverra un certificat signé.

Le fichier qu'on lui communique se nomme donc "certificatnehemie.csr".

Nous recevons de la part de l'autorité de certification un fichier comportant l'extension ".crt".

Création du dossier lié au site3 :

```
mkdir /var/www/html/site3
```

Nous allons copier coller le fichier site2.conf pour partir d'une bonne base.

```
cp /etc/apache2/sites-available/site2.conf /etc/apache2/sites-available/site3.conf
```

Nous l'éditions comme ceci :

```
<VirtualHost *:80>
    ServerName site3.nehemie.pod4.sio.lan
    DocumentRoot /var/www/html/site3
    Redirect permanent / https://site3.nehemie.pod4.sio.lan
    ErrorLog /var/log/apache2/error.log
    CustomLog /var/log/apache2/access.log combined

</VirtualHost>

<VirtualHost *:443>
    ServerName site3.nehemie.pod4.sio.lan
    DocumentRoot /var/www/html/site3

    SSLEngine On
    SSLCertificateFile /srv/ssl/certificatnehemie.crt
    SSLCertificateKeyFile /srv/ssl/clefnehemie.key
    ErrorLog /var/log/apache2/error.log
    CustomLog /var/log/apache2/access.log combined

</VirtualHost>
```

Nous activons le site :

```
echo "site3"|a2ensite
```

Nous activons le module ssl :

```
a2enmod ssl
```

Nous redémarrons le serveur apache.

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

Nous créons le fichier index.html du site 3 :

(chemin : /var/www/html/site3/index.html)

```
<html>
  <body>
    <h1>It works! SIO (SITE3)</h1>
    <p>This is the default web page for this server.</p>
  </body>
</html>
```

Création d'un troisième site avec un système d'authentification :

Création du dossier lié au site4 :

```
mkdir /var/www/html/site4
```

Nous créons le fichier de configuration apache pour le site :

```
<VirtualHost *:80>
  ServerName site4.nehemie.pod4.sio.lan
  DocumentRoot /var/www/html/site4

  ErrorLog /var/log/apache2/error.log
  CustomLog /var/log/apache2/access.log combined

  <Directory />
    Options FollowSymLinks
    AllowOverride all
```

```
</Directory>
```

```
<Directory /var/www/html/site4/admin>
```

```
Options Indexes MultiViews
```

```
AllowOverride all
```

```
order allow,deny
```

```
Allow from all
```

```
</Directory>
```

```
</VirtualHost>
```

Nous créons bien les dossiers admin et public.

```
mkdir admin  
mkdir publique
```

Nous créons le fichier .htpasswd

```
htpasswd -cm .htpasswd nem
```

Nous créons le fichier .htaccess.

```
AuthName "Acces Restricted"  
AuthType Basic  
AuthUserFile /var/www/html/site4/.htpasswd  
Require valid-user
```

Il faut faire attention sur le chemin du fichier .htpasswd sinon cela ne marchera pas.

Revision #53

Created 30 September 2020 12:28:32 by Nehemie

Updated 26 December 2022 13:39:25