

1. Introduction

Le Web est un des nombreux services disponibles sur Internet. Il s'agit de l'ensemble des documents accessibles par le protocole HTTP (par un navigateur Web). Une caractéristique essentielle du Web est la capacité de passer d'un document à un autre par des liens hypertexte. Un site Web est quant à lui un ensemble de pages Web, liées entre elles.

Pour accéder à un site Web, il vous faut utiliser un client Web, communément appelé navigateur, par exemple : firefox, Internet explorer, google chrome ...

Le protocole qui nous intéresse ici est HTTP qui est l'acronyme anglais de Protocole de Transfert HyperTexte. Apache est le plus populaire des serveurs HTTP. Il est produit par la « Apache Software Foundation ».

2. Installation

Afin de pouvoir compiler correctement Apache 2 sur Ubuntu, il faut dans un premier temps installer ces dépendances :

```
sudo apt-get install apache2
```

Pour démarrer Apache :

```
sudo service apache2 start
```

Ouvrez maintenant votre navigateur et allez sur la page `http://localhost`. Si vous voyez le message « It works! » c'est que vous avez réussi et Apache fonctionne parfaitement. Sinon les messages d'erreurs se trouvent dans le répertoire : `/usr/local/apache2/logs`.

3. configuration

Maintenant que le serveur est installé et fonctionne correctement, nous allons voir les bases à connaître de sa configuration. Rendez-vous dans le répertoire `/etc/apache2/`, et regardez les fichiers le composant :

```
cd /etc/apache2
ls
.conf  conf.d  envvars  httpd.conf  mods-available  mods-enabled  ports.conf
sites-available  sites-enabled
```

La plupart de ces fichiers sont plus ou moins spécifiques à debian/ubuntu et nous permettent de séparer la configuration en plusieurs parties.

- `httpd.conf` est le fichier utilisé par apache1, il est conservé vide dans apache2 pour assurer la rétrocompatibilité. Il ne nous servira pas ;
- `envvars` est utilisé pour définir des variables d'environnement propres à apache ;
- `ports.conf` est plus intéressant. Regardez son contenu :

```
sudo gedit ports.conf
```

La ligne `Listen 80` demande à apache d'écouter sur le port 80. Si vous utilisez HTTPS vous aurez à y spécifier le port 443. Bien sûr on peut utiliser n'importe quel port mais restons dans les standards.

- `apache2.conf` est le fichier principal de configuration ;
- `mods-available` contient la liste des modules d'apache installés ;

- *mods-enabled* celle des modules utilisés ;
- *sites-available* contient la liste des vhosts installés ;
- *sites-enabled* celle des vhosts utilisés.

3.1. Hôtes virtuels

Le serveur Web Apache2 est capable de gérer simultanément plusieurs arborescences Web grâce à la notion d'hôtes Virtuels (Virtual Hosts). Nous allons voir les différentes méthodes pour les mettre en place.

Nous renseignons le fichier `/etc/hosts` afin d'assurer la résolution de nom (Nom → Adresse Ip) des hôtes Virtuels que nous allons créer. Ceci n'est nécessaire que si votre serveur DNS (Serveur Bind) n'est pas installé et/ou configuré pour vos domaines ou encore, que les noms DNS attribués aux hôtes Virtuels sont purement fictifs ou ne vous appartiennent pas.

Dans un premier temps nous allons définir un nom DNS pour notre site `lari3.com`. Il faut ajouter cette information au fichier `/etc/hosts` pour que la résolution DNS pointe sur la machine locale (127.0.0.1) :

```
sudo gedit /etc/hosts
```

et rajouter la ligne suivante :

```
127.0.0.1 www.lari3.com
```

Puis nous allons créer un dossier dans le dossier `/var/www` qui est par défaut la racine d'apache.

```
sudo mkdir /var/www/lari3
```

Dans `/etc/apache2/sites-available` se trouve un fichier nommé *default*, recopiez-le avec le nom *lari3*

```
sudo cp /etc/apache2/sites-available/default /etc/apache2/sites-available/lari3
```

Ouvrez ensuite le fichier *lari3* avec votre éditeur de texte favori puis modifiez de cette façon :

```
<VirtualHost *:80>
    ServerAdmin votre-mail@lari3.com
    ServerName lari3.com
    ServerAlias www.lari3.com
    DocumentRoot /var/www/lari3
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/lari3>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>
```

Garder le reste tel qu'il est.



Pour terminer, il vous suffit de créer des liens du fichier nouvellement créé dans le dossier `/etc/apache2/sites-enabled`. Pour ce faire, une commande a été faite spécialement :

```
sudo a2ensite lari3
```

Puis, afin de recharger la configuration d'Apache :

```
sudo service apache2 reload
```

3.2. PHP

On a donc maintenant un apache fonctionnel, mais on va terminer avec un petit bonus et installer le module `mod_php`. Les modules sont des extensions d'apache qui ajoutent des fonctionnalités diverses.

On va ainsi installer `mod_php5` qui ajoutera comme son nom l'indique le support de php5 directement dans apache.

Installez le paquet `libapache2-mod-php5`, attendez le redémarrage d'apache, et voilà !

On va vérifier que cela marche. Regardez le dossier `mods-enabled`, vous pourrez y voir les fichiers `php5.conf` et `php5.load`. Là aussi, il s'agit de liens symboliques vers `mods-available` comme pour les vhosts, et vous devinerez l'utilité des commandes `a2enmod` et `a2dismod`.

Créez ensuite le fichier `/var/www/test.php`, et écrivez-y le contenu suivant :

```
<?php phpinfo(); ?>
```

Ouvrez ensuite cette page dans votre navigateur (`http://localhost/test.php`), vous y verrez la page interprétée (la fonction `phpinfo` affiche une page html d'informations sur votre installation de php).