

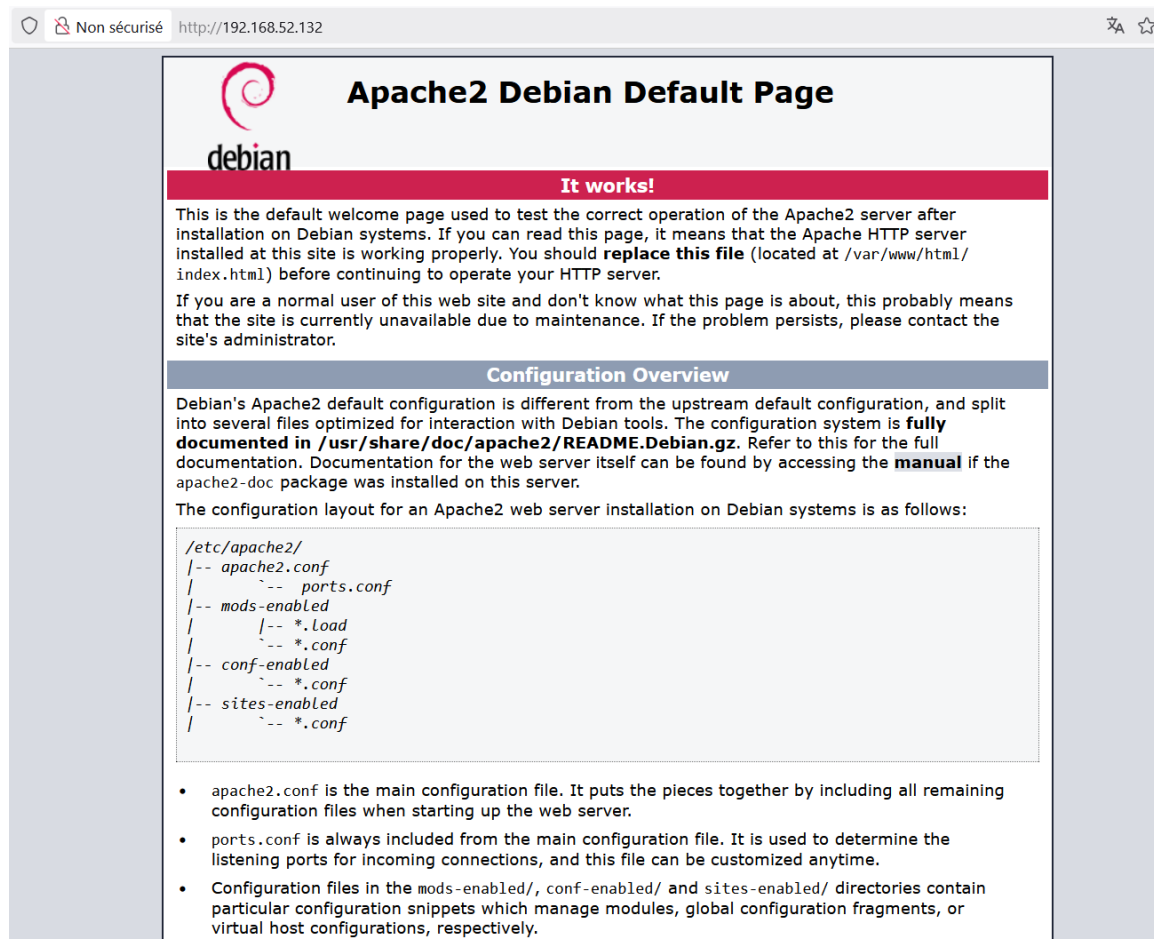
Installation simple d'un serveur Apache et création d'hôtes virtuels.

1) Installation du serveur Apache

L'installation du serveur Apache se fait avec la commande :

```
apt-get install apache2.
```

Après l'installation, on constate que le service Apache est automatiquement lancé, et le serveur est accessible en local via (pour moi) l'adresse 192.168.52.132 sur le port HTTP 80.



The screenshot shows a web browser window with the address bar displaying "http://192.168.52.132". The page title is "Apache2 Debian Default Page". The page content includes the Debian logo, a red banner saying "It works!", and a message stating that the Apache HTTP server is installed and working properly. It also provides instructions for replacing the default file and a configuration overview section.

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.

2) Configuration du serveur Web avec un seul Hôte Virtuel

Le fichier de configuration principal de Apache est:

`/etc/apache2/apache2.conf`

Le fichier de configuration de l'hôte virtuel par défaut est:

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

Le répertoire 'sites-available' contient les configurations disponibles, tandis que 'sites-enabled' contient les configurations activées.

Pour tester le serveur, la commande `telnet localhost 80` permet de se connecter.

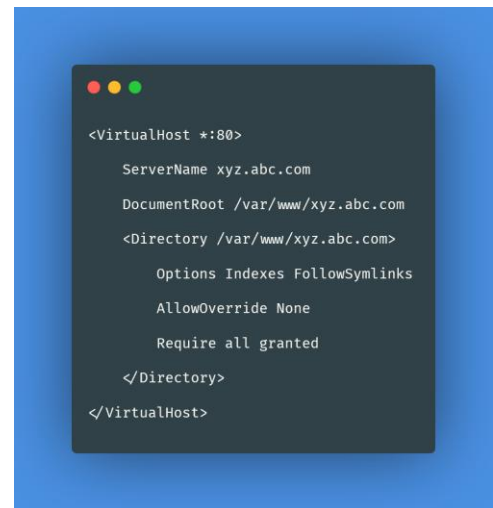
Après avoir modifié le répertoire des données du site et créé un fichier `index.html`, le serveur est relancé avec la commande `service apache2 restart`. Le test avec `telnet` montre que la nouvelle page est servie.

3) Configuration avec plusieurs Hôtes Virtuels

Pour mettre en place un serveur avec plusieurs hôtes virtuels comme `abc.abc.com` et `xyz.abc.com`, il faut créer un fichier de configuration pour chaque hôte dans `/etc/apache2/sites-available`, puis les activer avec `a2ensite`.

A screenshot of a terminal window showing the configuration for a VirtualHost named 'abc.abc.com'. The configuration is enclosed in <VirtualHost *:80> and </VirtualHost> tags. Inside, it sets ServerName to abc.abc.com, DocumentRoot to /var/www/abc.abc.com, and includes a <Directory> block for /var/www/abc.abc.com with options Indexes FollowSymlinks, AllowOverride None, and Require all granted.

```
<VirtualHost *:80>
    ServerName abc.abc.com
    DocumentRoot /var/www/abc.abc.com
    <Directory /var/www/abc.abc.com>
        Options Indexes FollowSymlinks
        AllowOverride None
        Require all granted
    </Directory>
</VirtualHost>
```

A screenshot of a terminal window showing the configuration for a VirtualHost named 'xyz.abc.com'. The configuration is enclosed in <VirtualHost *:80> and </VirtualHost> tags. Inside, it sets ServerName to xyz.abc.com, DocumentRoot to /var/www/xyz.abc.com, and includes a <Directory> block for /var/www/xyz.abc.com with options Indexes FollowSymlinks, AllowOverride None, and Require all granted.

```
<VirtualHost *:80>
    ServerName xyz.abc.com
    DocumentRoot /var/www/xyz.abc.com
    <Directory /var/www/xyz.abc.com>
        Options Indexes FollowSymlinks
        AllowOverride None
        Require all granted
    </Directory>
</VirtualHost>
```

Le serveur web utilise l'en-tête HTTP Host envoyé par le client pour savoir quel site doit être servi.