

Mise en place d'un serveur LAMP

Alizée LECERF – BTS A1

Qu'est qu'un serveur LAMP ?

Un serveur LAMP est un serveur web qui permet d'héberger un site internet ou une application.

LAMP est l'acronyme d'un système Linux (L), d'Apache (A), MySql ou MariaDB (M) et de PHP (P).

Pour installer un serveur LAMP, les prérequis sont :

- Une machine Linux fonctionnelle
- Une connexion internet
- Un utilisateur avec des droits administrateurs

Sommaire :

1. Installation d'Apache2
2. Installation de PHP
3. Installation d'une base de données (MySQL, MariaDB)
4. Créer un VirtualHost
5. Sécuriser l'accès au site avec un SSL

1. INSTALLATION D'APACHE2

1. Mettre à jour les paquets :

```
sudo apt update
```

2. Installation d'Apache2 :

```
sudo apt install -y apache2
```

-Y permet d'accepter toutes les demandes lors de l'installation du programme.

3. Vérifier si Apache2 est bien actif :

```
sudo systemctl status apache2
```

```
alizee@Linux:~$ sudo systemctl status apache2
```

```
• apache2.service - The Apache HTTP Server
```

```
Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enab>
```

```
Active: active (running) since Wed 2024-01-31 11:21:34 CET; 23min ago
```

```
Docs: https://httpd.apache.org/docs/2.4/
```

```
Main PID: 22537 (apache2)
```

```
Tasks: 55 (limit: 13111)
```

```
Memory: 23.1M
```

```
CPU: 295ms
```

```
CGroup: /system.slice/apache2.service
```

```
└─22537 /usr/sbin/apache2 -k start
```

```
└─22538 /usr/sbin/apache2 -k start
```

```
└─22539 /usr/sbin/apache2 -k start
```

```
Jan 31 11:21:33 Linux systemd[1]: Starting apache2.service - The Apache HTTP Se>
```

```
Jan 31 11:21:34 Linux systemd[1]: Started apache2.service - The Apache HTTP Ser>
```

4. Chercher l'adresse IP :

```
ip a
```

```
alizee@Linux:~$ ip a
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default  
t qlen 1000
```

```
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

```
inet 127.0.0.1/8 scope host lo
```

```
valid_lft forever preferred_lft forever
```

```
inet6 ::1/128 scope host noprefixroute
```

```
valid_lft forever preferred_lft forever
```

```
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr  
oup default qlen 1000
```

```
link/ether 08:00:27:f3:ae:68 brd ff:ff:ff:ff:ff:ff
```

```
inet 192.168.249.56/24 brd 192.168.249.255 scope global dynamic noprefixrout
```

```
e enp0s3
```

```
valid_lft 3474sec preferred_lft 3474sec
```

```
inet6 2a04:cec0:c028:f6c5:1ae7:a0a:a720:9c71/64 scope global temporary dynam
```

```
ic
```

```
valid_lft 7148sec preferred_lft 7148sec
```

```
inet6 2a04:cec0:c028:f6c5:a00:27ff:fe3:ae68/64 scope global dynamic mngtmpa
```

```
ddr noprefixroute
```

```
valid_lft 7148sec preferred_lft 7148sec
```

```
inet6 fe80::a00:27ff:fe3:ae68/64 scope link noprefixroute
```

```
valid_lft forever preferred_lft forever
```

5. Aller sur le site grâce à l'adresse IP pour vérifier qu'Apache2 est bien installé :



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
```

2.Installation de PHP

1. Installation de PHP :

```
sudo apt install -y php
```

2. Téléchargement des paquets supplémentaires :

Il est nécessaire de télécharger des paquets supplémentaires de php afin de permettre des interactions entre PHP et notre base de données MariaDB ou Mysql.

```
sudo apt install -y php-pdo php-mysql php-curl php-xml
php-mbstring php-zip php-gd
```

3. Vérifier l'installation de PHP :

On s'assure que l'installation de PHP a bien été fait en allant dans un fichier pour pouvoir ouvrir une page avec les informations de PHP.

```
sudo nano /var/www/html/info.php
```

```
GNU nano 7.2 /var/www/html/info.php *
<?php
phpinfo();
?>
```

Ensuite on se rend sur cette adresse 192.168.249.56/info.php

C'est votre adresse IP suivi du nom que vous avez donné au nom du dossier juste au-dessus.

192.168.249.56/info.php

PHP Version 8.2.7

System	Linux Linux 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-30) x86_64
Build Date	Jun 9 2023 19:37:27
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.2/apache2
Loaded Configuration File	/etc/php/8.2/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.2/apache2/conf.d
Additional .ini files parsed	/etc/php/8.2/apache2/conf.d/10-mysqld.ini, /etc/php/8.2/apache2/conf.d/10-opcache.ini, /etc/php/8.2/apache2/conf.d/10-pdo.ini, /etc/php/8.2/apache2/conf.d/15-xml.ini, /etc/php/8.2/apache2/conf.d/20-calendar.ini, /etc/php/8.2/apache2/conf.d/20-ctype.ini, /etc/php/8.2/apache2/conf.d/20-curl.ini, /etc/php/8.2/apache2/conf.d/20-dom.ini, /etc/php/8.2/apache2/conf.d/20-exif.ini, /etc/php/8.2/apache2/conf.d/20-ffi.ini, /etc/php/8.2/apache2/conf.d/20-fileinfo.ini, /etc/php/8.2/apache2/conf.d/20-ftp.ini, /etc/php/8.2/apache2/conf.d/20-gd.ini, /etc/php/8.2/apache2/conf.d/20-gettext.ini, /etc/php/8.2/apache2/conf.d/20-iconv.ini, /etc/php/8.2/apache2/conf.d/20-mbstring.ini, /etc/php/8.2/apache2/conf.d/20-mysqli.ini, /etc/php/8.2/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.2/apache2/conf.d/20-phar.ini, /etc/php/8.2/apache2/conf.d/20-posix.ini, /etc/php/8.2/apache2/conf.d/20-readline.ini, /etc/php/8.2/apache2/conf.d/20-shmop.ini, /etc/php/8.2/apache2/conf.d/20-simplexml.ini, /etc/php/8.2/apache2/conf.d/20-sockets.ini, /etc/php/8.2/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.2/apache2/conf.d/20-sysvsem.ini, /etc/php/8.2/apache2/conf.d/20-sysvshm.ini, /etc/php/8.2/apache2/conf.d/20-tokenizer.ini, /etc/php/8.2/apache2/conf.d/20-xmlreader.ini, /etc/php/8.2/apache2/conf.d/20-xmlwriter.ini, /etc/php/8.2/apache2/conf.d/20-xsl.ini, /etc/php/8.2/apache2/conf.d/20-zip.ini
PHP API	20220829
PHP Extension	20220829
Zend Extension	420220829
Zend Extension Build	API420220829.NTS
PHP Extension Build	API20220829.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
Zend Max Execution Timers	disabled
IPv6 Support	enabled

3.Installation d'une base de données tels que MySQL ou MariaDB

Ici on va utiliser MariaDB.

1. Installation de MariaDB :

```
sudo apt install -y mariadb-server
```

2. Sécuriser MariaDB :

```
sudo mariadb-secure-installation
```

3. Rentrer ces données pour l'installation de MariaDB :

```
Switch to unix_socket authentication [Y/n] n
... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.
```

4. Se connecter à la base de données MariaDB :

```
sudo mariadb -u root -p
```

```
alizee@Linux:~$ sudo mariadb -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.11.4-MariaDB-1~deb12u1 Debian 12
```

5. Vérifier si on dispose bien d'une base de données :

```
SHOW DATABASES;
```

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.001 sec)
```

6. Créer une database :

```
CREATE DATABASE nomdelabase;
```

7. Vérifier que la database s'est bien créée :

```
SHOW DATABASES;
```

```
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| alili |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.001 sec)
```

8. Sortir de MariaDB :

EXIT

4. Créer un VirtualHost

1. Créer un dossier pour changer l'adresse de notre site :

Il faut taper dans /etc/apache2/sites-available/nomdevotresite

```
sudo nano /etc/apache2/sites-available/alili.localhost.conf
```

2. Rentrer ces données dans le dossier pour le configurer :

*<VirtualHost *:80>*

ServerName example.com

ServerAlias www.example.com

ServerAdmin webmaster@example.com

DocumentRoot /srv/web/example.com/www

<Directory /srv/web/example.com/www>

Options -Indexes +FollowSymLinks +MultiViews

AllowOverride none

Require all granted

</Directory>

ErrorLog /var/log/apache2/error.example.com.log

CustomLog /var/log/apache2/access.example.com.log combined

</VirtualHost>

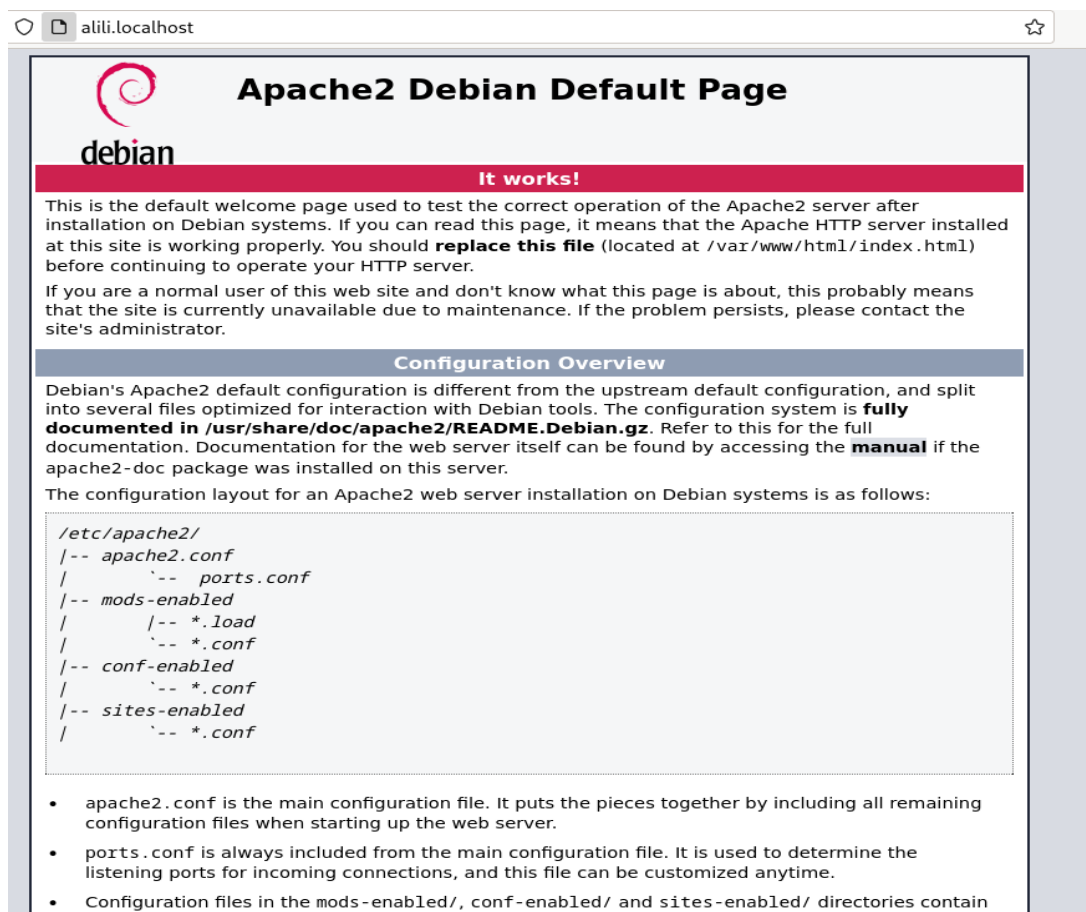
```

GNU nano 7.2 /etc/apache2/sites-available/alili.localhost.conf
<VirtualHost *:80>
    ServerName alili.localhost
    ServerAlias www.alili.localhost
    ServerAdmin alili.localhost
    DocumentRoot /srv/web/alili.localhost/www
    <Directory /srv/web/alili.localhost/www>
        Options -Indexes +FollowSymLinks +MultiViews
        AllowOverride none
        Require all granted
    </Directory>
    ErrorLog /var/log/apache2/error.alili.localhost.log
    CustomLog /var/log/apache2/access.alili.localhost.log combined
</VirtualHost>

```

3. Redémarrer Apache2 :

```
sudo systemctl restart apache2
```



The screenshot shows a web browser window with the address bar set to `alili.localhost`. The page title is "Apache2 Debian Default Page". Below the title, there is a red banner with the text "It works!". The main content area contains a paragraph explaining that this is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. It also mentions that the configuration system is fully documented in `/usr/share/doc/apache2/README.Debian.gz`. Below this, there is a section titled "Configuration Overview" which lists the configuration files and their locations:

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

Below the list, there are three bullet points explaining the purpose of these files:

- `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...

5.Sécuriser l'accès au site avec un SSL

1. Mettre Apache2 en mode SSL :

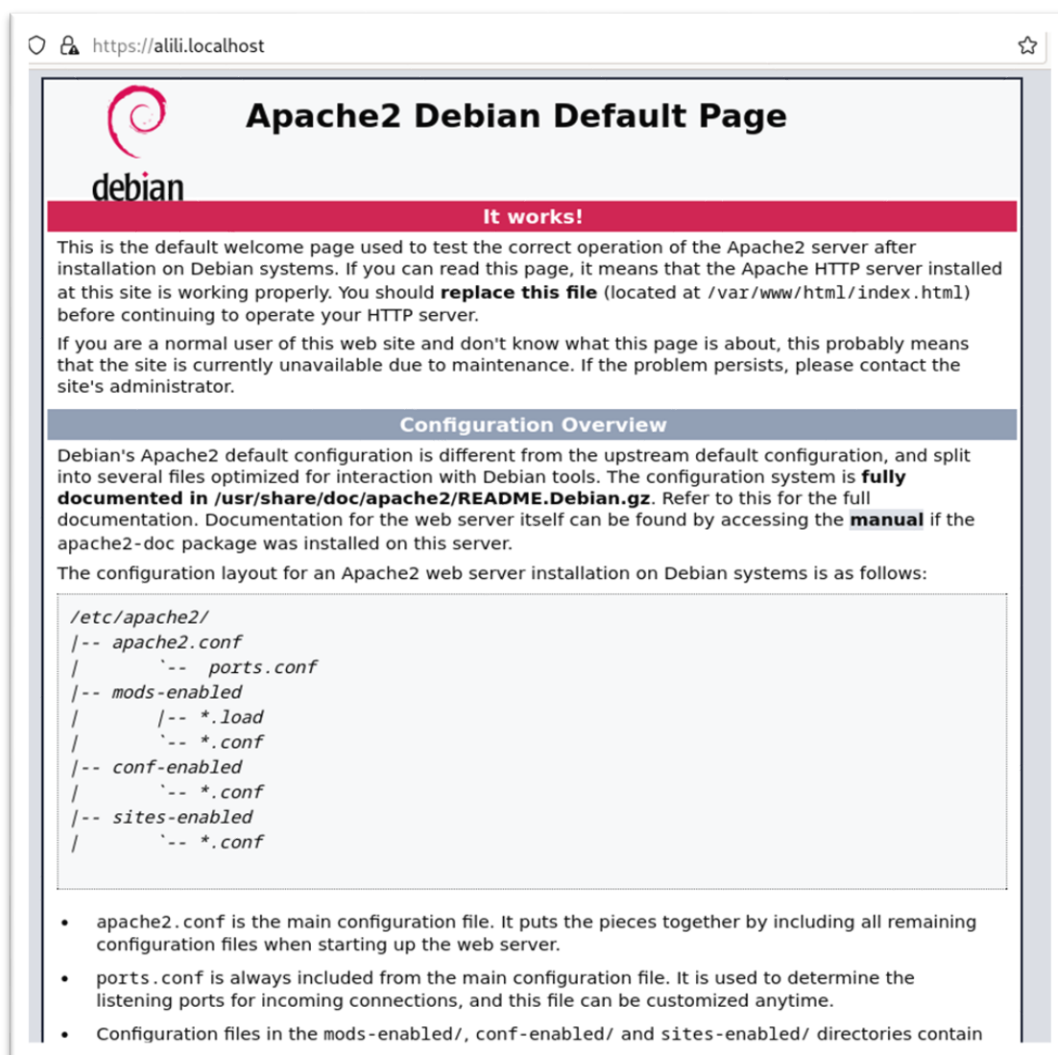
```
sudo a2enmod ssl
```

2. Activer le SSL sur le site :

```
sudo a2ensite default-ssl
```

3. Redémarrer Apache2 :

```
sudo systemctl restart apache2
```



The screenshot shows a web browser window with the address bar displaying `https://alili.localhost`. The page title is "Apache2 Debian Default Page". The page features the Debian logo and a red banner that says "It works!". Below the banner, there is a paragraph of text explaining that this is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. It mentions that if the page is visible, the Apache HTTP server is working properly, and it advises replacing the file at `/var/www/html/index.html` before continuing to operate the HTTP server. It also provides information for normal users who might not know what the page is about, suggesting that the site might be unavailable due to maintenance and advising them to contact the site's administrator.

Below the text, there is a section titled "Configuration Overview". This section explains that Debian's Apache2 default configuration is different from the upstream default configuration and is split into several files optimized for interaction with Debian tools. It states that the configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. It refers to this for the full documentation and mentions that documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The section then states: "The configuration layout for an Apache2 web server installation on Debian systems is as follows:" and provides a directory tree structure:

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

Below the directory tree, there is a list of three bullet points:

- `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