

LAP

LAMP

Mise en place d'un serveur apache2

Installer apache2 avec la commande `apt install apache2`.

Les fichiers de configuration sont dans `/etc/apache2/`.

Les sites web sont dans `/var/www/html/`.

Démarrer ou gérer le service avec `systemctl start apache2`

Installer PHP avec **`apt install php`**.

La page de base d'apache avec l'ip de la machine et celle ci :

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.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

Mise en place de vhosts

Un vhost permet d'avoir plusieurs sites sur une seule machine.

Les fichiers sont dans `/etc/apache2/sites-available/`.

(exemple de vhost)

```
<VirtualHost *:80>
    ServerName glpi.gsb.sio.jja
    Redirect permanent / https://glpi.gsb.sio.jja/
</VirtualHost>

<VirtualHost *:443>
    ServerAdmin webmaster@localhost
    ServerName glpi.gsb.sio.jja
    DocumentRoot /var/www/glpi/glpi

<Directory /var/www/glpi/glpi>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf

# SSL Engine Switch:
# Enable/Disable SSL for this virtual host.
SSLEngine on

# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
# /usr/share/doc/apache2/README.Debian.gz for more info.
# If both key and certificate are stored in the same file, only the
# SSLCertificateFile directive is needed.
SSLCertificateFile /etc/ssl/certs/glpi.crt
SSLCertificateKeyFile /etc/ssl/private/glpi.key

# Server Certificate Chain:
```

Ajouter un enregistrement DNS pour le domaine.

Activer le site avec `a2ensite exemple.conf`.

Mise en place de certificat

Le SSL permet de sécuriser les échanges entre client et serveur.

Ci dessous mise en place d'un auto signé, pour un signé par autorité de certif :

<https://sioaubusson.fr/books/essentielle/page/certification>

Génération d'un certificat auto-signé

Créer un certificat avec openssl.

screen
(commande openssl dans le terminal)

Les fichiers sont créés dans `/etc/ssl/`.

Copier le fichier SSL avec `cp default-ssl.conf test-ssl.conf`.

screen
(fichier `test-ssl.conf` ouvert)

Ajouter une redirection HTTP vers HTTPS.
Configurer `ServerName` et `DocumentRoot`.
Configurer les chemins des certificats.

screen
(fichier avec `SSLCertificateFile` et `Key`)

Activer SSL avec `a2enmod ssl`.
Activer le site avec `a2ensite test-ssl.conf`.
Recharger apache.

screen
(terminal avec activation SSL)

Accéder au site en HTTPS.

screen
(site sécurisé affiché dans navigateur)

Revision #6

Created 2026-03-23 13:11:56 UTC by Admin

Updated 2026-05-14 14:02:42 UTC by Admin