

# TP Apache2

## Présentation

### Apache2, qu'est-ce que cette histoire de cowboy ?

Apache2 est un serveur HTTP, c'est à dire un module chargé de recevoir et de renvoyer des données selon le protocole HTTP.

En d'autres termes, le navigateur internet de l'internaute envoie des requêtes à apache2 et en retour apache2 renvoie les données des pages à afficher du site internet que l'internaute est en train de consulter.

## Installation & Configuration d'apache2

Pendant toute la procédure il faut absolument avoir le droit root sur le serveur

# apt update & apt upgrade

# apt install apache2

Pour savoir quelle version d'apache a été installée :

```
apachectl -V
```

Pour vérifier le fonctionnement sur serveur : <http://l'ip> du serveur



The screenshot shows a web browser window displaying the 'Apache2 Ubuntu Default Page'. The page features the Apache logo and the text 'It works!' in a red banner. Below this, there is a paragraph explaining that this is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It also provides instructions for users who are not administrators, suggesting they contact the site's administrator if the page is unavailable. A 'Configuration Overview' section follows, detailing the configuration layout for an Apache2 web server installation on Ubuntu systems, listing files like /etc/apache2/apache2.conf, ports.conf, mods-enabled, conf-enabled, and sites-enabled. A bullet point at the bottom states that apache2.conf is the main configuration file.

Les fichiers de configuration sont dans le répertoire : [/etc/apache2](#)  
**Le fichier de configuration d'apache2**

```
root@dns1:/etc/apache2
root@dns1:/home/fadhe1# cd /etc/apache2/
root@dns1:/etc/apache2# ls
apache2.conf  conf-available  conf-enabled  envvars  magic  mods-available  mods-enabled  ports.conf  sites-available  sites-enabled
root@dns1:/etc/apache2#
```

## Le fichier de configuration d'apache2

Le fichier de configuration d'apache2 se trouve ici : **`/etc/apache2/apache2.conf`**

Il peut contenir des inclusions vers d'autres fichiers de configurations secondaires (Include) donc si vous ne trouvez pas certains éléments dans `apache2.conf` il va falloir chercher dans les fichiers inclus.

### Sauvegarde du fichier de configuration

Avant toutes manipulations Il est préférable de sauvegarder du fichier de configuration d'apache2 et en cas de problème on peut revenir en arrière.

```
cp -i /etc/apache2/apache2.conf /etc/apache2/apache2.conf_BACKUP
```

### Vérification de l'inclusion des virtualhosts

Nous verrons plus tard ce que sont des virtualhosts, mais maintenant on va vérifier ce paramètre de configuration :

```
nano /etc/apache2/apache2.conf
IncludeOptional conf-enabled/*.conf
IncludeOptional sites-enabled/*.conf
```

### Vérification des utilisateurs et groupes d'apache

```
nano /etc/apache2/apache2.conf
# These need to be set in /etc/apache2/envvars
```

```
User ${APACHE_RUN_USER}
Group ${APACHE_RUN_GROUP}
```

## Vérifiez la directive DirectoryIndex

On choisit quels fichiers seront interrogés par défaut lorsque vos internautes voudront afficher la page de votre site  
`http://www.example.com/`

```
nano /etc/apache2/mods-enabled/dir.conf

DirectoryIndex index.html index.php index.htm index.x
html
```

## Gestion du service apache2

```
# systemctl start apache 2
```

```
# systemctl restart apache 2
```

```
# systemctl status apache 2
```

```
# systemctl reload apache 2
```

```
# systemctl stop apache 2
```

```
# systemctl enable apache 2
```

## Tester la configuration d'apache2

```
apache2ctl configtest
```

## Activation de quelques modules utiles :

Activer le module Gzip:

```
a2enmod deflate
```

Activer le module Rewrite:

```
a2enmod rewrite
```

Activer le module Expire:

```
a2enmod expires
```

# SECURISATION DE : PACHE2

## Limiter les informations visibles :

---

### Le fichier security

Limitez donc au maximum les informations fournies par Apache en procédant comme ceci :

```
nano -l /etc/apache2/conf-available/security.conf
```

**Ligne : 25**

**ServerTokens Prod**

**Ligne : 36**

**ServerSignature Off**

## Créer les virtualhosts

### Qu'est-ce qu'un virtualhost?

Les virtualhosts ou *hôtes virtuels* permettent d'héberger plusieurs sites internet (nom de domaine) sur un même serveur dédié.

Donc une seule adresse IP pour plusieurs sites internet ou nom de domaine, c'est un peu le principe des hébergements mutualisés.

Donc il vous faudra un virtualhost par nom de domaine ou sous domaine hébergé sur votre serveur dédié.

### Où se trouvent donc les virtualhosts?

Les deux répertoires importants sont :

-/etc/apache2/sites-available

-/etc/apache2/sites-enabled

Le répertoire **sites-available**, contient les fichiers de configurations des sites internet disponibles, par défaut, seul le fichier /etc/apache2/sites-available/default est présent.

Le répertoire **sites-enabled** contient des liens symboliques vers les fichiers du répertoire "sites-available", pour activer un virtualhost il faut juste créer un lien dans ce répertoire.

**Donc :** :  
/etc/apache2/sites-available sont les sites disponibles (available)  
/etc/apache2/sites-enabled sont les sites actifs, activés (enabled)

### Copie de sauvegarde de /etc/apache2/sites-available/default

Dans le répertoire /etc/apache2/sites-available en principe il y a déjà un virtualhost par défaut, on va en faire une copie de sauvegarde mais dans un autre répertoire que nous allons créer. Cette copie de sauvegarde peut être utile en cas de mauvaise manipulation des virtualhosts.

```
mkdir /backup  
cp -i /etc/apache2/sites-available/default /backup/virtual_host_default
```

Pour activer un site déjà créer dans : /etc/apache2/sites-available (C'est-à-dire créer un lien symbolique dans : /etc/apache2/sites-enabled

On utilise la commande suivante :

```
#a2ensite monsite.conf
```

(a2= apache2, en= enable, site = le nom du site qu'on veut activer)

Et pour désactiver :

```
#a2dissite monsite.conf
```

(a2= apache2, dis= disable, site = le nom du site qu'on veut activer)

## Les dossiers qui hébergent les sites (Le contenu => les DATAs )

Le dossier par défaut est : /var/www/html

```
root@dns1: /var/www/html
```

```
root@dns1:/etc/apache2# cd /var/www/html/
```

```
root@dns1:/var/www/html# ls
```

```
index.html
```

```
root@dns1:/var/www/html# █
```

La page : index.html est la page d'accueil du site par défaut

## Création du premier virtualhost

```
#cd /etc/apache2
```

```
#ls
#cd sites-available
#ls
```

On va utiliser le modèle offert par le site par défaut

```
#cp 000-default.conf tssr.conf
```

```
root@dns1:/etc/apache2# ls
apache2.conf  conf-available  conf-enabled  envvars  magic  mods-available  mods-enabled  ports.conf  sites-available  sites-enabled
root@dns1:/etc/apache2# cd sites-available/
root@dns1:/etc/apache2/sites-available# ls
000-default.conf  default-ssl.conf
root@dns1:/etc/apache2/sites-available# cp 000-default.conf tssr.conf
```

```
#nano tssr.conf
```

```
# Virtualhost écoutant sur le port 80
```

```
<VirtualHost *:80>
```

```
#Nom du serveur auquel le vhost doit répondre
```

```
    ServerName dns1.tssr.lan
```

```
#Eventuel alias supplémentaire
```

```
    ServerAlias www.tssr.lan
```

```
# Mail du webmaster
```

```
    ServerAdmin webmaster@tssr.lan
```

```
# Racine des fichiers du site
```

```
    DocumentRoot "/var/www/html/tssr/"
```

```
# Délocaliser pour ce vhost les logs d'erreur
```

```
# Délocaliser pour ce vhost les logs d'accès
```

```
    ErrorLog /var/log/apache2/dns1.tssr.lan-error_log
```

```
    ErrorLog ${APACHE_LOG_DIR}/error.log
```

```
    CustomLog ${APACHE_LOG_DIR}/access.log combined"
```

```
#Définition des droits d'un répertoire
```

```
    <Directory "/var/www/html/tssr/">
```

```
        Options Indexes FollowSymLinks
```

```
        AllowOverride All
```

```
        Require all granted
```

```
    </Directory>
```

```
</VirtualHost>
```

**Une fois le site est configuré, on procède à son activation**

```
#a2ensite tssr.conf
```

Pour que les modifications soient prises en compte on doit recharger la configuration :

```
#systemctl reload apache2  
#systemctl restart apache2  
#systemctl status apache2
```

## **Maintenant notre site est en ligne, mais il est toujours sans contenu**

On va créer le répertoire qui va héberger le contenu, le chemin de ce répertoire a été déclarée dans le fichier de configuration dans la section : **DocumentRoot "/var/www/html/tssr/"**

```
#mkdir /var/www/html/tssr
```

On va créer la page d'accueil index.html sur Windows 10 et on va utiliser Winscp pour la transférer sur notre site :

- 1- Installation du wincsp
- 2- Création de la page index.html
- 3- Transfer de la page vers le répertoire : **/var/www/html/tssr**
- 4- Désactivation du site par défaut : **a2dissite 00.....**
- 5- Test
- 6- Ajout d'un réenregistrement CNAME dans le fichier : dir.tssr.lan (la zone directe dur DNS)