

# TD – Apache 2

## Présentation de l'environnement

Il existe plusieurs fichiers de configuration tous présents dans `/etc/apache2/`

- Le fichier **apache2.conf** contient les paramètres généraux et communs à tous les serveurs et plusieurs “Include” vers les autres fichiers.
- Le fichier **ports.conf** contient la liste des ports en écoute.
- Les modules sont dans le répertoire `/etc/apache2/mods-available`
- Les modules activés sont dans `/etc/apache2/mods-enabled`

## Gestion des modules

- Pour activer un module (ce qui revient donc à créer le lien), il est pratique d'utiliser la commande suivante : **sudo a2enmod** nom du module
- Pour désactiver un module **sudo a2dismod** nom du module

## Gestion des sites

- `/etc/apache2/sites-available` contient tous les fichiers de configuration des serveurs web
- `/etc/apache2/sites-enabled` contient la configuration des sites web activés
- **sudo a2ensite** fichier\_conf active un site
- **sudo a2dissite** fichier\_conf désactive un site

## Démarrer Apache

`sudo service apache2 restart` ou `systemctl restart apache2`

Vous pouvez vérifier la syntaxe des fichiers de configuration par la commande : `apache2 -t`

### apache2.conf

1 – `user www-data` et `group www-data`, spécifient le compte anonyme utilisé par le serveur une fois qu'il est lancé. Ce compte doit pouvoir lire les fichiers de configuration et ceux de la racine du serveur HTTP (attention donc aux droits sur les pages web desservies).

2 – `ServerRoot` indique l'adresse du répertoire racine du serveur, où sont stockés les fichiers de configuration du serveur HTTP.

3 – `PidFile /var/run/apache2.pid`, indique le fichier où le serveur en exécution stocke son numéro de processus (PID).

4 – `ErrorLog, fichier error_log`, journalisation des erreurs.

Il est fréquent d'héberger plusieurs serveurs web sur un même poste aussi la déclaration et le paramétrage des différents serveurs est déportée dans des fichiers à placer dans `/etc/apache2/sites-available`

Le fichier `000default` y est déjà présent pour assurer le paramétrage du site principal par défaut dont la racine se trouve, toujours par défaut à `/var/www/`.

Le site par défaut est déjà activé ; on retrouve donc un lien vers ce fichier dans `/etc/apache2/sites-enabled`.

### Étape 1 -installation de 2 cartes réseaux dans la machine virtuelle 1

- Dans la VM ajouter une deuxième carte
- Vérifier le nom des cartes `ip a`
- Dans le paramétrage de Debian associer les adresses mac aux différentes cartes

**Carte mode pont enp0s3**

Adresse IP 192.168.1.127/24

serveur dns 192.168.1.127

**Carte mode interne enp0s8**

Adresse IP 10.10.10.10/24.

- Redémarrer Debian et vérifier la configuration de vos adresses IP via **ip a**
- Vérifier les infos complètes via **command nmcli device show enp0s3**
- Vérifier que votre serveur DNS fonctionne

## Étape 2 – installation d'Apache 2

La commande : **apt install apache2** installera le serveur web avec ses dépendances.

La commande **apt install libapache2-mod-php5** installe le module PHP (optionnel)

Si vous voulez installer la documentation en local : **apt-get install apache2-doc**

Une fois le service installé, testez-le en lançant le navigateur avec l'adresse 127.0.0.1

## Étape 3 – modifier la page d'accueil du site par défaut

Se placer dans le répertoire **/var/www/html**

Renommer le fichier index.html en backindex.html **mv index.html backindex.html**.

Créer un fichier site1.html

```
nano index.html
<html>
<head>
<title>Mon site</title>
</head>
<body>
<H1>Bienvenue sur ma page</H1>
<H4>Test OK </H4>
</body> </html>
```

**Modifier la page d'accueil au niveau du site web**

ouvrir le fichier dir.conf

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

et vérifier que la page index.html est indiquée.

### Redémarrer Apache

```
service apache2 restart ou service apache2 reload
```

Tester la nouvelle page avec votre navigateur

### Étape 4 – modifier l'adresse IP d'accès au site

Par défaut Apache écoute toutes les adresses, pour forcer une adresse particulière:

ouvrir le fichier /etc/apache2/ports.conf

```
nano /etc/apache2/ports.conf
```

```
listen votre adresse:80
```

Puis recharger apache **service apache2 reload**

Tester l'accès au site avec votre navigateur avec les 2 adresses

IP

### Étape 5 – affecter un autre port au site 1

ouvrir le fichier /etc/apache2/ports.conf

```
nano /etc/apache2/ports.conf
```

```
listen votre adresse:8080
```

Puis recharger apache **service apache2 reload**.

Tester l'accès au site avec votre navigateur avec votre adresse IP, puis avec *votre adresse IP:8080*

## Étape 6 – affecter des ports différents à vos 2 adresses

Par défaut Apache écoute toutes les adresses, pour forcer une adresse particulière:

ouvrir le fichier `/etc/apache2/ports.conf`

```
nano /etc/apache2/ports.conf
listen 192.168.x.x:80
listen 10.x.x.x:8080
```

Puis recharger apache `service apache2 reload`

## Étape 7 – utiliser le nom d'hôte pour l'accès au site

Cette solution permet d'accéder au site par son nom et uniquement par son nom, l'accès via adresse IP ne fonctionne plus.

## Étape 8 – sécuriser son site

### Sécurisation des accès à certains répertoires.

On peut régler pour chaque répertoire le droit d'accéder aux pages contenues dans ce répertoire, en fonction de la machine cliente et/ou de l'utilisateur qui se connecte.

Le fichier dans lequel ce paramétrage s'effectue est un fichier de configuration présent dans `/etc/apache2/sites-available/`.

#### Exemple :

On désire autoriser l'accès du répertoire `"/intranet"` aux machines du réseau d'adresse `192.168.100.0/24` et l'interdire à toutes les autres. Il importe de préciser dans quel ordre les règles de restriction vont être appliquées. `"order allow,deny"` (on accepte d'abord et on refuse ensuite).

Créer un répertoire `/intranet` dans votre site et créer une page `«page2.html»`,

```
mkdir /var/www/html/intranet
nano /var/www/html/page2.html
```

Dont le contenu sera le suivant:

```
<html>bienvenue sur la page2</html>
```

Modifier la page `index.html` et ajouter la ligne:

```
<a href=«intranet/page2.html»>vers intranet</a>
```

Tester le lien dans le navigateur

Affecter les règles suivantes dans le fichier de configuration `/etc/apache2/sites-available/`

```
<Directory /var/www/html/intranet>
#Ordre de lecture des règles
Require all denied
Require ip 192.168.100.0
</Directory>
```

Puis recharger apache `service apache2 reload` et tester dans votre navigateur.

Tester l'accès au site via une machine du réseau 192.168.1.0 et d'une machine en 10.0.0.0

## Sécurisation des accès à certaines personnes

### Authentifier l'accès à un répertoire

Ce procédé va permettre de sécuriser l'accès à un répertoire ou à des fichiers. L'accès sera autorisé à une ou plusieurs personnes ou encore à un ou plusieurs groupes de personnes.

**AuthName**, définit ce qui sera affiché au client pour lui demander son nom et son mot de passe,

**AuthType**, définit le mode d'authentification et d'encryptage "basic" avec HTTP/0 ou "MD5" par exemple avec HTTP/1.

**AuthUserFile**, définit le fichier qui contient la liste des utilisateurs et des mots de passe. Ce fichier contient deux champs (Nom d'utilisateur, Mot de passe crypté).

Créer le fichier de mot de passe dans `/usr/local/apache/passwd`

Utilisation de la commande “htpasswd” : `htpasswd -c /usr/local/apache/passwd/passwords toto` puis saisir le mot de passe

`htpasswd /usr/local/apache/passwd/passwords titi` puis saisir le mot de passe *le -c permet de créer le fichier, à n'utiliser que la première fois.*

Vérifier les mots de passe en ouvrant le fichier.

Mettre en place la sécurité d'accès au répertoire en affectant les règles suivantes dans le fichier de configuration `/etc/apache2/sites-available/`

```
<Directory /var/www/html/intranet/>
AuthType Basic
AuthName "Accès contrôlé"
AuthUserFile "/usr/local/apache/passwd/passwords"
Require valid-user
</Directory>
```

Redémarrer Apache et tester l'accès au site intranet

## Autorisation d'accès à un groupe

Activer le module group `a2enmod authz_groupfile`.

Créer un fichier de groupes nommé `groups` par exemple, qui associe des noms de groupes avec une liste d'utilisateurs de ce groupe.

**Direction:** `tutu tata`

Pour ajouter un utilisateur à votre fichier de mots de passe préexistant, entrer:

```
htpasswd /usr/local/apache/passwd/passwords tutu
htpasswd /usr/local/apache/passwd/passwords tata
```

Maintenant, vous devez modifier dans le fichier de configuration `/etc/apache2/sites-available/` comme suit :

```
AuthTypeBasic
AuthName "Groupe Direction"
AuthBasicProviderfile
AuthUserFile "/usr/local/apache/passwd/passwords"
AuthGroupFile "/usr/local/apache/passwd/groups"
Requiregroup GroupName.
```

Redémarrer le service apache et tester avec le navigateur

## Création d'un certificat auto signé pour l'accès chiffré SSL

1. Mettre à jour les paquets  
**sudo apt update**
2. Activez le module SSL  
**sudo a2enmod ssl**
3. Redémarrez Apache pour activer le module :  
**sudo systemctl restart apache2**

## Création du certificat SSL

*Créer les fichiers de clés et de certificats SSL avec la commande openssl*

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout
/etc/ssl/private/autosigne.key -out /etc/ssl/certs/autosigne.crt
```

*Compléter les renseignements*

Country Name *FR*

State or Province Name *IDF*

Locality Name (eg, city) *Paris*

Organization Name (eg, company) *Société*

Organizational Unit Name (eg, section) *Informatique*

Common Name *societe.com*

Email Address *webmaster@societe.com*

## Configuration d'Apache pour utiliser SSL

nano /etc/apache2/sites-available/default-ssl.conf

```
<VirtualHost *:443>
ServerName www.societe.com
SSLEngine on
SSLCertificateFile /etc/ssl/certs/autosigne.crt
SSLCertificateKeyFile /etc/ssl/private/autosigne.key
</virtualhost>
```

Relancer Apache et activer le site et tester avec le navigateur

```
sudo a2ensite default-ssl.conf
sudo systemctl reload apache2
```

Tester avec le navigateur

### Étape 9 – gérer plusieurs sites web sur un même serveur

Déclaration des sites existants

Copie du site par défaut dans les 2 nouveaux sites

```
cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-
available/site1.conf
cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-
available/site2.conf
```

Création des répertoires des sites

```
mkdir /var/www/html/site1
mkdir /var/www/html/site2
```

Création de la page index.html dans le répertoire site1

```
<html><H1>SITE 1</H1></html>
```

Création de la page index.html dans le répertoire site2

```
<html><H1>SITE 2</H1></html>
```

Modification du fichier du site

Saisir les informations suivantes:

```
<VirtualHost192.168.1.127:80>  
ServerName www1.testweb.lan  
DocumentRoot /var/www/html/site1  
</VirtualHost>
```

Activation du site1 **a2ensite site1**

Modification du fichier du site

Saisir les informations suivantes:

```
<VirtualHost10.10.10.10:80>  
ServerName www2.testweb.lan  
DocumentRoot /var/www/html/site2  
</VirtualHost>
```

Activation du site2 **a2ensite site2**