

Serveur Web - VirtualHost

1. Serveur web (configuration minimale)

Procédure installation d'un serveur web sous debian 11

Mettre à jour le système : `# apt update && apt upgrade`

Installation des paquets pour le serveur web `# apt install apache2 php`

Pour vérifier que le serveur web fonctionne correctement vous pouvez taper l'adresse IP dans votre navigateur, vous devriez avoir la page par défaut d'apache2.

Configuration du serveur web - apache2

Les fichiers de configuration d'apache pour les virtualhost se trouve dans le dossier `/etc/apache2/sites-enabled`

Par défaut apache charge les pages contenus dans le dossier `/var/www/html`. Pour changer ce dossier, il faut :

- D'abord créez le dossier avec la commande : `# mkdir /var/www/lsi`
- Allez dans le dossier de configuration : `# cd /etc/apache2/sites-enabled`
- Copiez la configuration par défaut : `# cp 000-default.conf default.conf`
- Suppression la configuration par défaut `# rm 000-default.conf`
- Editez le fichier de configuration nouvellement créé: `# nano default.conf` et modifiez la ligne «DocumentRoot `/var/www/html`» en «DocumentRoot `/var/www/lsi`»
- Redémarrez apache avec la commande suivante : `# systemctl restart apache2`

Apache chargera désormais l'`index.php` contenu dans le dossier `/var/www/lsi`

2. Pages personnelles

Fonctionnement

Supposons que vous êtes l'utilisateur "user1" du réseau et que vous ayez des pages personnelles (Site web!) . Il sera possible d'accéder à vos pages, avec l'adresse suivante:

<http://lsi.lan/~user1/index.html> (ou AdresseIP/~user1/index.html)

- Le (tilde "~") permet d'accéder à votre répertoire personnel.
- La requête sera réellement exécutée sur `"/home/user1/public_html/index.html`.

Ceci est possible grâce à la directive particulière **Userdir public html** qui permet de gérer pour chaque utilisateur (c'est à dire chaque possesseur d'un compte) des pages personnelles.

Dans apache2, cette directive est en fait associé à un module non activé par défaut. Il suffit donc d'activer le module correspondant par la commande `$ sudo a2enmod userdir`, ce qui aura pour effet de créer deux liens vers les fichiers correspondants `userdir.conf` et `userdir.load`.

Note pour désactiver le module il suffit de taper la commande : `$ sudo a2dismod userdir`

Attention, vérifier que le répertoire personnel ne soit pas en mode 700, car personne ne pourrait accéder aux pages personnelles.

La configuration par défaut n'est pas très sécurisée car elle oblige à mettre les droits de lecture à tout le monde (755) sur le répertoire personnel. Il vous est possible de changer la variable UserDir par exemple :

UserDir /home/web

Pensez à changer la valeur de Directory dans le fichier /etc/apache2/mods-available/userdir.conf

Configuration des pages personnelles :

Ajouter dans le fichier /etc/apache2/apache2.conf à

la ligne 57 (environ) : **ServerName 127.0.0.1** (ou l'adresse fixe du serveur)

et à la ligne 115 en environ, modifier/ajouter les lignes suivantes:

User www-data

Group www-data

DirectoryIndex index.html index.php index.xhtml

UserDir public_html

Activer le module userdir : **# cd /etc/apache2/mods-available/**

sudo a2enmod userdir

Redémarrer le serveur apache 2 : **# sudo systemctl restart apache2**

mkdir /etc/skel/public_html

mkdir /etc/skel/logs

echo " <h1>Nouvel espace web crée</h1> " > /etc/skel/public_html/index.html

useradd -g www-data -m user1

Pour tester le bon fonctionnement : <http://localhost/~user1> (ou depuis un client :

<http://AdressIP/~user1>)

3. Les Hôtes Virtuels avec Apache2

Le Serveur Web Apache2 est capable de gérer simultanément plusieurs arborescences Web grâce à la notion d'hôtes Virtuels (Virtual Hosts). Nous allons voir les différentes méthodes pour les mettre en place.

3.1. Les Hôtes Virtuels basés sur l'adresse IP

Création des adresses ip virtuelles à l'interface eth0 (mettre le nom de votre interface au lieu de eth0) (192.168.1.22 par exemple)

sudo ifconfig eth0:0 192.168.1.11

sudo ifconfig eth0:1 192.168.1.12

Ajouter les deux ligne dans le fichier /etc/hosts

192.168.1.11	lsi1.lan
192.168.1.12	lsi2.lan

/etc/apache2/sites-available/lsi1.conf	/etc/apache2/sites-available/lsi2.conf
<VirtualHost 192.168.1.11:80> DocumentRoot /var/www/lsi1.lan ServerName lsi1.lan </VirtualHost>	<VirtualHost 192.168.1.12:80> DocumentRoot /var/www/lsi2.lan ServerName lsi2.lan </VirtualHost>

Pour les tests vous pouvez utiliser le chmod 755 sur ces répertoires.

```
# sudo mkdir /var/www/lsi1.lan
# sudo mkdir /var/www/lsi2.lan
# sudo echo " <h1>Site : LSI1</h1> " > /var/www/lsi1.lan/index.html
# sudo echo " <h1>Site : LSI2</h1> " > /var/www/lsi2.lan/index.html
# sudo chmod -R 755 /var/www/lsi1.lan
# sudo chmod -R 755 /var/www/lsi2.lan
```

Nos deux hôtes virtuels étant configurés, nous allons les activer avec les commandes :

```
# cd /etc/apache2/sites-available
# sudo a2ensite lsi1.conf
# sudo a2ensite lsi2.conf
```

Une fois la configuration activée, testez si la syntaxe est correcte: # sudo apachectl configtest

N'oubliez pas de recharger la configuration du serveur : # sudo service apache2 reload

Pour tester le bon fonctionnement : <http://lsi1.lan> et <http://lsi2.lan>

3.2 Hôtes Virtuels basés sur le numéro de port

Par défaut, le Serveur Web Apache2 est configuré pour écouter sur le port **80**. Il est possible d'utiliser des ports différents pour chaque hôte virtuel.

Dans cet exemple, les deux Sites Virtuels lsi1.lan et lsi2.lan se partagent la même adresse IP :

192.168.1.22 Seul le port d'écoute est différent.

Dans un premier temps, nous allons éditer le fichier /etc/hosts afin de déclarer les 2 noms de domaine associés à la seule et unique adresse IP de notre interface réseau. Ceci permettra la résolution de nom (Nom ⇒ Adresse Ip).

Exemple de fichier hosts :

```
127.0.0.1 localhost
192.168.1.22 lsi1.lan
192.168.1.22 lsi2.lan
```

Ensuite, nous modifions les 2 fichiers de configuration :

/etc/apache2/sites-available/lsi1.conf	/etc/apache2/sites-available/lsi2.conf
<VirtualHost 192.168.1.22: 81 > DocumentRoot /var/www/lsi1.lan ServerName lsi1.lan </VirtualHost>	<VirtualHost 192.168.1.22: 82 > DocumentRoot /var/www/lsi2.lan ServerName lsi2.lan </VirtualHost>

Rajouter dans le fichier /etc/apache2/ports.conf

```
Listen 81
Listen 82
```

Désactivée puis activez alors les hôtes virtuels de la même manière qu'avec la première méthode et rechargez la configuration du serveur Apache2 :

```
# sudo a2dissite lsi1.conf
# sudo a2dissite lsi2.conf
# sudo service apache2 reload
# sudo a2ensite lsi1.conf
# sudo a2ensite lsi2.conf
# sudo service apache2 reload
```

Le site lsi1.lan sera accessible via l'url : http://lsi1.lan:81 et le site lsi2.lan via l'url : http://lsi2.lan:82

3.3. Hôtes Virtuels basés sur le nom de domaine

Cette troisième méthode est la plus utilisée et la plus conseillée. Elle tend même à devenir un standard. Il s'agit simplement d'associer plusieurs noms DNS à une seule adresse IP.

Dans l'exemple suivant, nous allons configurer deux Sites virtuels qui utiliseront tous les deux l'adresse IP 192.168.1.22. Si vous souhaitez configurer un serveur de développement autonome, vous pouvez simplement utiliser l'adresse de bouclage, 127.0.0.1.

Dans un premier temps, nous allons éditer le fichier /etc/hosts afin de déclarer les 2 noms de domaine associés à la seule et unique adresse IP de notre interface réseau. Ceci permettra la résolution de nom (Nom ⇒ Adresse Ip). Exemple de fichier /etc/hosts :

```
127.0.0.1 localhost
192.168.1.22 lsi1.lan lsi2.lan
192.168.1.22 www.lsi1.lan www.lsi2.lan
```

Notre fichier /etc/hosts étant correctement renseigné, nous allons ajouter la directive NameVirtualHost à la configuration du serveur Serveur Web Apache2 en mettant cette ligne dans un nouveau fichier: /etc/apache2/conf.d/virtualhost.conf. Pour ceci, entrez la commande suivante dans un terminal:

```
$ sudo echo "NameVirtualHost 192.168.1.22:80" > /etc/apache2/conf.d/virtual-host.conf
```

Note : Créer le répertoire conf.d si il n'existe pas.

Ensuite, nous modifions deux fichiers de configurations :

/etc/apache2/sites-available/lsi1.conf	/etc/apache2/sites-available/lsi2.conf
<VirtualHost 192.168.1.22:80 > DocumentRoot /var/www/lsi1.lan ServerName lsi1.lan ServerAlias www.lan1.lan </VirtualHost>	<VirtualHost 192.168.1.22:80 > DocumentRoot /var/www/lsi2.lan ServerName lsi2.lan ServerAlias www.lan2.lan </VirtualHost>

Le site lsi1.lan sera accessible via l'url : http://lsi1.lan ou http://www.lsi1.lan

Le site lsi2.lan sera accessible via l'url : http://lsi2.lan ou http://www.lsi2.lan