

LAMP

Objectifs :

Être capable d'installer un serveur Web Apache et ses différents composants.
Comprendre et savoir utiliser un serveur Web Apache

Matériel à avoir :

Une machine Debian 11 « propre » ainsi qu'une machine Windows cliente (7 ou 10, au choix).

Consignes :

Installez les deux machines : Linux et Windows.
Configurez-les en IP fixe. Les deux machines doivent communiquer entre elles, et avoir accès à Internet.
Une fois cela fait et vérifié, vous pouvez poursuivre ce TP.

Explications

1. LAMP

LAMP est en fait une suite de 4 paquets :

- L pour la distribution Linux
- A pour le serveur web Apache
- M pour la base de données MySQL et MariaDB
- P pour le langage de programmation Php / Pearl ou Python

La base donnée MySQL est celle utilisée par défaut sur les machines Debian, jusqu'à Debian 8 Jessie. Depuis Debian 9 Stretch et Debian 10 Buster, la base de données par défaut est Maria DB. Nous utiliserons donc celui-là dans notre cours.

Cette suite de logiciel existe aussi sur Windows et s'appelle donc WAMP.

Nous allons installer et configurer ces 4 paquets qui nous permettent de réaliser un serveur Web Apache complet.

2. Configuration de la machine

Si ce n'est pas déjà fait :

- Mettez votre machine en IP fixe, connectée à internet avec résolution DNS fonctionnelle grâce à votre passerelle. Faites un ping www.google.fr pour vérifier.
- Mettez le serveur <http://deb.debian.org/debian/> en tant que miroir et mettez à jour votre serveur.
- Configurez une connexion basique ssh.

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

Installation de LAMP

Téléchargez et installez les paquets nécessaires à LAMP :

- Apache2 le paquet qui permet d'installer le serveur Web en lui-même.
- Libapache2-mod-php c'est le module d'Apache pour exécuter PHP.
- Mariadb-server est le paquet qui permet d'installer le serveur de base de données MariaDB.
- Php est le paquet pour installer php.
- Php-mysql permet à php d'interagir avec le serveur de base de données MariaDB.
- Tous les paquets php-* servent à installer les modules de Php les plus courants.

```
sudo apt install apache2 libapache2-mod-php mariadb-server php php-  
mysql php-cli php-curl php-gd php-intl php-memcache php-xml php-zip  
php-mbstring php-json
```

Le serveur vous demande d'installer une centaine de dépendances avec ces paquets.
Acceptez ces installations, elles sont nécessaires.

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023



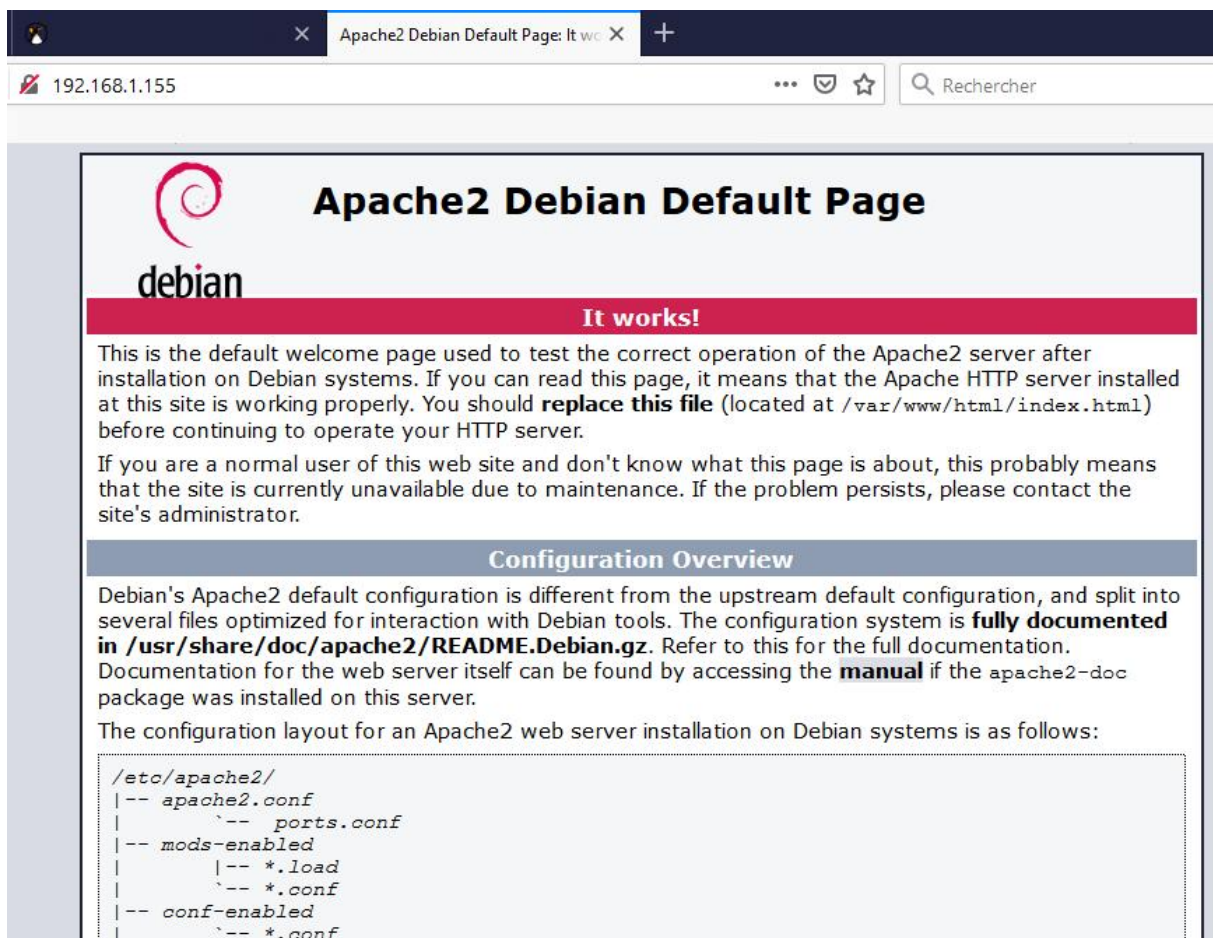
Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse écrite et préalable de l'ADRAR.

LAMP

Tests

3. Fonctionnement Apache

À présent, si on accède au serveur via son adresse IP depuis un navigateur internet, on a la page d'Apache qui apparaît, le serveur web fonctionne.



Auteur :

Morgane BONIN-RICHON

Date création :

30-11-2017

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
 ☑ Sophie POULAKOS
 ☑ Mathieu PARIS

Date révision :

15-04-2023

LAMP

4. Fonctionnement PHP

Déplacez-vous dans le dossier /var/www/html. Il s'agit du dossier racine d'Apache.

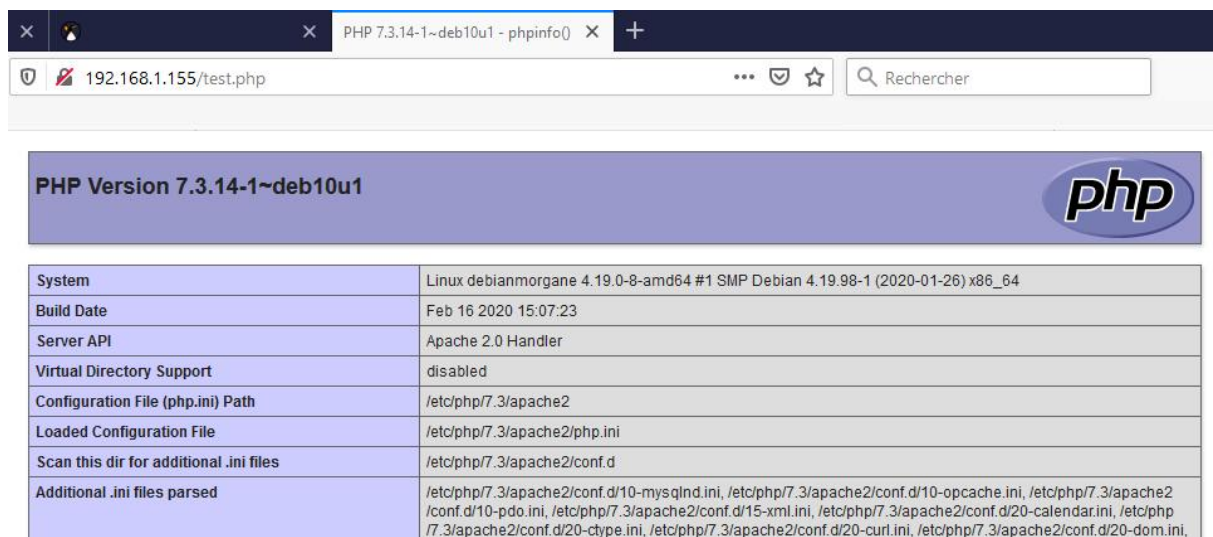
Créons un fichier en « test.php » et mettons dans son contenu une fonction en php qui va afficher toutes les informations sur le serveur PHP. Pour cela, faites :

```
sudo nano test.php
```

```
<?php phpinfo(); ?>
```

Pour tester que le serveur PHP fonctionne il faut donc aller sur votre navigateur internet, et rajouter le nom de votre nouveau fichier php juste après votre adresse IP.

Si la page s'affiche correctement, c'est que le Serveur PHP fonctionne. Sinon, réinstallez-le.



PHP Version 7.3.14-1~deb10u1	
System	Linux debianmorgane 4.19.0-8-amd64 #1 SMP Debian 4.19.98-1 (2020-01-26) x86_64
Build Date	Feb 16 2020 15:07:23
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.3/apache2
Loaded Configuration File	/etc/php/7.3/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.3/apache2/conf.d
Additional .ini files parsed	/etc/php/7.3/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.3/apache2/conf.d/10-opcache.ini, /etc/php/7.3/apache2/conf.d/10-pdo.ini, /etc/php/7.3/apache2/conf.d/15-xml.ini, /etc/php/7.3/apache2/conf.d/20-calendar.ini, /etc/php/7.3/apache2/conf.d/20-ctype.ini, /etc/php/7.3/apache2/conf.d/20-curl.ini, /etc/php/7.3/apache2/conf.d/20-dom.ini,

Vous devriez voir toutes les informations de votre serveur php.

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

5. Fonctionnement MariaDB

On a installé MariaDB, mais il faut le configurer pour pouvoir s'y connecter. Pour cela, faites :

```
sudo mysql_secure_installation
```

- Vous venez d'installer MariaDB, donc root n'a pas encore de mot de passe, donc vous devez juste faire « Entrée ».

```
morgane@debianmorgane:/var/www/html$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
```

- MariaDB utilise un système de connexion sécurisé plus poussé qu'une authentification classique avec un mot de passe. Donc, nous n'allons pas paramétrer de mot de passe pour root. Répondez « N » à la question correspondante.

```
Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] n
... skipping.
```

- Ensuite, le système va vous demander plusieurs questions, répondez « y » à chaque fois pour valider la modification.

```
Remove anonymous users? [Y/n] y
Disallow root login remotely? [Y/n] y
Remove test database and access to it? [Y/n] y
Reload privilege tables now? [Y/n] y
```

=> Cela va permettre de supprimer les utilisateurs anonymes, de désactiver les connexions distantes de root, de supprimer la base de données de tests et ensuite, de recharger la table des privilèges.

À présent, vous pouvez vous authentifier à MariaDB grâce à sa sécurisation spéciale. Pour cela, vous avez simplement à faire :

```
sudo mariadb
```

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

Vous devez obtenir cet affichage :

```
morgane@debianmorgane:/var/www/html$ sudo mariadb
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 73
Server version: 10.3.22-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> █
```

Vous constaterez que vous n'avez pas eu besoin de fournir de mot de passe pour vous connecter, en effet, vous avez simplement utilisé le terme « sudo ». C'est cela qui permet de vous authentifier sur le serveur MariaDB. Car seuls les utilisateurs disposant des privilèges sudo peuvent se connecter à MariaDB.

Pour regarder un peu ce qui se trouve sur votre gestionnaire de BDD, tapez la commande :

```
SHOW DATABASES ;
```

```
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.000 sec)

MariaDB [(none)]> █
```

Vous obtenez un tableau contenant le nom de toutes les bases de données qui existent sur votre serveur et auxquelles vous avez accès grâce à cette connexion administrateur sur votre SGBD (Système de Gestion de Base de Données) MariaDB.

Pour augmenter la sécurité de votre système, vous pouvez créer un utilisateur pour gérer votre base de données et lui accorder des privilèges personnalisés sur la BDD.

On va en premier créer une nouvelle BDD. Toujours en étant connecté à MariaDB grâce à sudo, vous devez utiliser la commande suivante :

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

```
MariaDB [(none)]> CREATE DATABASE exemple_bdd;  
Query OK, 1 row affected (0.000 sec)
```

```
MariaDB [(none)]> █
```

```
CREATE DATABASE exemple_bdd ;
```

Maintenant, créons le nouvel utilisateur tout en lui accordant des droits spécifiques :

```
GRANT ALL ON exemple_bdd.* TO 'usertest'@'localhost' IDENTIFIED BY  
'azerty' WITH GRANT OPTION;
```

```
MariaDB [(none)]> GRANT ALL ON exemple_bdd.* TO 'usertest'@'localhost' IDENTIFIED BY  
'azerty' WITH GRANT OPTION;  
Query OK, 0 rows affected (0.000 sec)
```

Cela donnera à l'utilisateur **usertest** des privilèges complets sur la base de données **exemple_bdd**, tout en empêchant cet utilisateur de créer ou de modifier d'autres bases de données sur votre serveur.

Pour recharger les nouveaux droits, vous devez faire :

```
MariaDB [(none)]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.000 sec)
```

```
FLUSH PRIVILEGES ;
```

Déconnectez-vous de votre compte « admin » en faisant « exit ».

```
MariaDB [(none)]> exit  
Bye
```

```
morgane@debianmorgane:/var/www/html$
```

Et reconnectez-vous à présent sur l'utilisateur que nous venons de créer : **usertest**. Cela se fait grâce à la commande `mariadb -u usertest -p`

`-u` est l'option pour préciser l'utilisateur ensuite. Et `-p` indique à la commande de demander le mot de passe juste après, sinon, le système ne permettra pas votre authentification.

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

```
morgane@debianmorgane:/var/www/html$ mariadb -u usertest -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 76
Server version: 10.3.22-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Testons à présent à quelles BDD vous avez accès. Faites : `SHOW DATABASES ;`

```
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| exemple_bdd |
| information_schema |
+-----+
2 rows in set (0.001 sec)
```

On voit bien que votre utilisateur test n'a pas accès au même nombre de BDD que vous en tant qu'admin. Notre restriction de droit a bien marché !

Vous pouvez vous déconnecter de la communication avec MariaDB en faisant « exit ».

Pour apprendre à vous servir du SGBD MariaDB en ligne de commande, cela se fera un petit plus tard, dans les cours.

Création d'un site web

Maintenant que nous avons un serveur Web 100% fonctionnel, apprenons créer un site internet et à le rendre accessible depuis notre serveur !

On va mettre en place un site web accessible à l'adresse : site exemple1.lan.

6. Contenu du site

Un site web, c'est en premier un contenu qu'on veut diffuser grâce à un serveur Web.

- Le dossier `/var/www` est le dossier qui va accueillir le contenu des sites internet que l'on créer, déplacez-vous dedans.

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

- À l'intérieur de ce dossier, vous devez créer le répertoire `exemple1`, c'est dans ce dossier que nous mettrons la page (ou toutes les pages si vous voulez) de notre site Web.
- Créons donc la première page, la page d'accueil. Il suffit de créer un fichier texte, de le nommer « `index.html` » et d'y écrire du contenu, en HTML, à l'intérieur.
- Dans ce fichier qui sert de page d'accueil, écrivez le code HTML suivant :

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>HELLO WORLD</title>
</head>
<body>
<p> Bienvenue sur mon site Exemple1 :)</p>
</body>
</html>
```

Voilà, nous avons notre première page !

Mais ce n'est pas pour autant que notre site est fonctionnel. Pour qu'il soit accessible, il faut créer un VirtualHost.

7. Configuration du site

VirtualHosts, hôtes virtuels c'est-à-dire l'hébergement et la gestion de plusieurs sites internet sur une seule adresse IP, donc virtuels.

Chaque site internet qu'on va héberger sur notre machine aura un fichier « `.conf` » à son nom, ce qui permettra de lui configurer son VirtualHost. Grâce à cela, on pourra définir (au minimum) son nom et où se situe le contenu du site.

Pour cela, on passe concrètement à la configuration du serveur Web lui-même. On va donc se déplacer sur notre machine Debian, dans le dossier `/etc/apache2`.

Ensuite, la configuration des fichiers de VirtualHost se fait dans le dossier `sites-available`, déplacez-vous à l'intérieur de ce dossier.

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.

LAMP

```
morgane@debianmorgane:/etc/apache2/sites-available$ ls -al
total 20
drwxr-xr-x 2 root root 4096 avril 13 15:45 .
drwxr-xr-x 8 root root 4096 avril 13 15:45 ..
-rw-r--r-- 1 root root 1332 avril  2  2019 000-default.conf
-rw-r--r-- 1 root root 6338 avril  2  2019 default-ssl.conf
morgane@debianmorgane:/etc/apache2/sites-available$
```

Si vous faites un `ls -al`, vous constaterez qu'il y a déjà des fichiers présents :

Le `000-default.conf` qui configure la « page par défaut » d'Apache. Celle que vous obtenez quand vous tapez l'adresse IP de votre serveur sur un navigateur.

Et le `default-ssl.conf` est le fichier par défaut qu'on peut utiliser pour configurer le SSL sur notre serveur.

Nous devons créer un troisième fichier, celui destiné à notre site internet. Comme vous voyez, il est important que votre nom de fichier finisse par « `.conf` » pour être pris en compte. Et par convention, on le nomme avec notre nom de domaine.

Cela fait donc pour nous, la commande :

`sudo nano exemple1.conf` si on est déjà dans le dossier `/etc/apache2/sites-available` ou bien, on rajoute évidemment le chemin absolu du dossier si on n'est pas déjà placé dedans.

À l'intérieur de ce fichier, il faut définir les paramètres de notre site, c'est ce qu'on appelle le VirtualHost :

```
<VirtualHost *:80>
    ServerName exemple1.lan
    ServerAlias www.exemple1.lan
    ServerAlias monsite.lan
    DocumentRoot /var/www/exemple1
</VirtualHost>
```

En bleu ce que vous devez recopier, en faisant attention à la syntaxe

En rouge ce qui est obligatoire

En vert ce qui est optionnel

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

```
ServerName exemple1.lan
ServerAlias monsite.lan
DocumentRoot /var/www/exemple1
</VirtualHost>
```

Ouverture de la configuration du VirtualHost

`*:80` indique que le serveur Web répondra sur le port 80 quelque soit son adresse IP

`ServerName` indique le nom principal de votre site internet

`ServerAlias` indique un second nom / un alias auquel votre site internet pourra répondre

`DocumentRoot` permet d'indiquer le dossier qui contient les pages du site

Fermeture de la configuration du VirtualHost

Si vous ne l'avez pas fait, cherchez sur internet à

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
 ☑ Sophie POULAKOS
 ☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

quoi servent les différentes parties de cette configuration. C'est très important de savoir comment cela fonctionne.

De plus, vous trouverez d'autres paramètres qui peuvent vous sembler intéressants, car ici n'est présenté que le minimum vital pour que votre site fonctionne correctement.

Enregistrez votre fichier et **relancez le serveur Apache**.

```
sudo systemctl restart apache2
```

Si celui-ci ne veut pas redémarrer, c'est probablement qu'il y a une erreur de syntaxe dans le fichier VirtualHost que vous venez de configurer. **Attention**, cela peut être même un espace en trop, ou en moins !

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.

LAMP

8. Activation du virtualHost

Votre site possède un contenu, vous lui avez défini un paramétrage pour le rendre accessible sur un nom donné grâce à son VirtualHost, mais votre site n'est toujours pas accessible depuis internet.

En effet, pour le moment, votre VirtualHost se situe uniquement dans le dossier « sites-available » qui veut dire sites-disponible. Cela signifie que ce dossier va contenir tous les virtualHost de tous les sites qui existent sur la machine.

Mais cela ne veut pas dire que tous ces sites soient accessibles depuis internet.

Pour qu'ils deviennent accessibles, il faut les activer !

Pour cela, on va créer un lien, un raccourci de notre fichier présent dans le dossier « sites-availables » (disponibles) et le placer dans le dossier « sites-enabled » (actifs).

Cela veut dire qu'on active notre VirtualHost. Donc, seuls les VirtualHost actifs sont accessibles depuis internet. La commande pour réaliser cette manipulation est la suivante :

```
sudo a2ensite exemple1.conf
```

`a2` = apache2 et `ensite` = enabled site donc sites-enabled. Cette commande permet donc de créer ce qu'on appelle un lien symbolique entre le sites-available (sites disponibles) d'exemple1 et le sites-enabled (sites activés) d'exemple1.

Si vous ne créez pas le lien symbolique, le site sera simplement disponible et non activé. Donc il ne sera pas visible sur internet.

La commande précédente vous demande de redémarrer à nouveau Apache2 pour prendre en compte la nouvelle configuration.

À présent, si vous faites un `ls -al` de sites-available et un `ls -al` de sites-enabled, vous devriez voir cela :

```
morgane@debianmorgane:/etc/apache2/sites-available$ ls -al
total 24
drwxr-xr-x 2 root root 4096 avril 13 19:47 .
drwxr-xr-x 8 root root 4096 avril 13 15:45 ..
-rw-r--r-- 1 root root 1332 avril  2 2019 000-default.conf
-rw-r--r-- 1 root root 6338 avril  2 2019 default-ssl.conf
-rw-r--r-- 1 root root 117 avril 13 19:47 exemple1.conf
morgane@debianmorgane:/etc/apache2/sites-available$ cd ../sites-enabled/
morgane@debianmorgane:/etc/apache2/sites-enabled$ ls -al
total 8
drwxr-xr-x 2 root root 4096 avril 13 19:47 .
drwxr-xr-x 8 root root 4096 avril 13 15:45 ..
lrwxrwxrwx 1 root root  35 avril 13 15:45 000-default.conf -> ../sites-available/000-default.conf
lrwxrwxrwx 1 root root  32 avril 13 19:47 exemple1.conf -> ../sites-available/exemple1.conf
morgane@debianmorgane:/etc/apache2/sites-enabled$
```

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023

LAMP

Vous constatez que dans le dossier sites-enabled vous avez un lien exemple1.conf (visible par le L au niveau des droits à la place du d pour le dossier ou - pour un fichier) qui pointe -> sur le dossier `sites-available/exemple1.conf`

Nous avons donc activé correctement notre site internet.

Accéder à vos sites

Votre site est bien configuré, il ne reste plus qu'à tester que vous y avez bien accès depuis une machine cliente.

Cependant, tapez dans votre navigateur le nom de votre site internet ne fonctionnera pas maintenant. En effet, comment votre machine cliente saurait résoudre le nom que vous demandez, en l'adresse IP du serveur sur lequel est hébergé le site ? Il a besoin d'un serveur DNS.

Donc, si vous avez un serveur DNS, rajoutez le nom de votre site et ses alias à l'intérieur (écrits comme vous l'avez fait dans le fichier exemple1.conf).

- Si vous n'avez pas de serveur DNS, ce qui est le cas si vous avez bien suivis ce cours sur une machine propre, vous allez apprendre à mettre en place un enregistrement DNS local en utilisant le fichier `/etc/hosts` de votre machine cliente.

=> Pour cela, vous allez rajouter le nom de votre site et de ses alias dans le fichier hosts qui permet de faire la résolution de nom en l'absence de serveur DNS. Sur votre machine cliente Windows, ce fichier se trouve dans `C:/windows/system32/drivers/etc/hosts`.

Rappelez-vous, il fonctionne de la même manière que le fichier hosts de Linux.

Sur une ligne il suffit d'écrire l'adresse IP de la machine et le nom avec lequel vous souhaitez y accéder.

Attention, une ligne par nom/alias. Si vous avez 1 nom et 2 alias il vous faudra 3 lignes.

Attention, éditez le fichier en tant qu'administrateur. Pour cela, vous pouvez utiliser le logiciel Notepad ++ qui détecte lors de l'enregistrement si vous avez besoin des droits administrateurs ou non. Si vous en avez besoin, il vous propose de passer en mode 'Administrateur'. Il nous vous reste plus qu'à modifier votre fichier et l'enregistrer.

```
192.168.1.155    exemple1.lan
192.168.1.155    monsite.lan
```

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

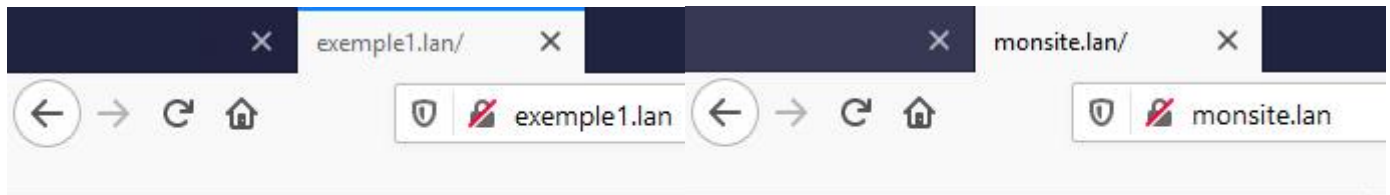
15-04-2023



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.

LAMP

Maintenant, testez le bon fonctionnement de votre site en allant sur un navigateur internet et en tapant dans la barre d'adresse le nom ou l'alias de votre site. S'il apparaît, c'est que tout fonctionne. Sinon, vous avez un problème de configuration d'Apache, ou de votre fichier hosts.



Bienvenu sur mon site Exemple 1 !

Bienvenu sur mon site Exemple 1 !

Bravo, vous savez maintenant créer un site web de base, le paramétrer et le rendre actif.
Passez au tp suivant

Auteur :

Morgane BONIN-RICHON

Relu, validé & visé par :

☑ Jérôme CHRETIENNE
☑ Sophie POULAKOS
☑ Mathieu PARIS

Date création :

30-11-2017

Date révision :

15-04-2023