

UNIVERSITE DE KINSHASA
FACULTE DES SCIENCES ET TECHNOLOGIES
MENTION MATHEMATIQUE STATISTIQUE ET INFORMATIQUE



TRAVAIL PRATIQUE DE SYSTEME
D'EXPLOITATION

GROUPE N°1 : CONFIGURATION DU SERVEUR WEB SOUS LINUX

PROMOTION : LICENCE 2 LMD

- BYAMUNGU AMANI Kevin
- MALEBA DIYAZOLA Sang Précieux
- MALONGONAY BESIA Loïs
- MASELA KAYA Gemima
- MASENGA MAKIESE Exaucé
- MOMBESA GLODI Glodi
- MUDIOLO MULANGA Amédée
- MUMBE DIANTUADI Benjamin
- N'SANDUNU MUKALA Esther
- NAKALENGI MBUDA Dieu Avu

DIRIGE PAR : Prof. KASENGEDIA MOTUMBE
Ass. KANINGINI Junior

ANNEE ACADEMIQUE 2022-2023

CONFIGURATION DU SERVEUR WEB SOUS LINUX

La configuration d'un serveur web sous Linux consiste à mettre en place et personnaliser les différents paramètres et fonctionnalités du serveur web afin de répondre aux besoins spécifiques de l'utilisateur. Elle est essentielle pour assurer son bon fonctionnement et sa sécurité.

En résumé, la configuration d'un serveur web sous Linux permet de personnaliser et de sécuriser le fonctionnement du serveur, de gérer les sites web hébergés, de contrôler l'accès aux ressources, et de surveiller les activités du serveur. (Source : <https://expernet.re/formation/configuration-dun-serveur-web-sous-linux/> <https://openclassrooms.com/fr/courses/1733551-gerez-votre-serveur-linux-et-ses-services/5236051-installez-le-serveur-web-libre-apache-httpd>). Voici quelques raisons pour lesquelles la configuration d'un serveur web sous Linux est importante :

- ✓ Gestion des permissions d'accès : La configuration permet de définir les permissions d'accès aux fichiers et répertoires du serveur web, ce qui permet de contrôler qui peut accéder aux ressources du serveur (<https://expernet.re/formation/configuration-dun-serveur-web-sous-linux/>).
- ✓ Sécurisation de l'accès à distance : La configuration permet de mettre en place des mesures de sécurité pour protéger l'accès au serveur web depuis des connexions à distance. Cela peut inclure l'utilisation de certificats SSL/TLS, la configuration du pare-feu pour filtrer le trafic entrant, et l'utilisation de protocoles sécurisés comme SSH. (<https://expernet.re/formation/configuration-dun-serveur-web-sous-linux/>).
- ✓ Gestion des modules et des fonctionnalités : La configuration permet d'activer ou de désactiver des modules et des fonctionnalités spécifiques du serveur web. Par exemple, on peut activer le support de PHP, MySQL ou d'autres langages de programmation pour étendre les fonctionnalités du serveur. (<https://expernet.re/formation/configuration-dun-serveur-web-sous-linux/>).
- ✓ Configuration des Virtual hosts : Les Virtual hosts permettent d'héberger plusieurs sites web sur le même serveur en utilisant des noms de domaine différents. La configuration permet de définir les paramètres spécifiques à chaque site web, comme le répertoire racine, les alias de domaine, et les paramètres de sécurité. (<https://openclassrooms.com/fr/courses/1733551-gerez-votre-serveur-linux-et-ses-services/5236051-installez-le-serveur-web-libre-apache-httpd>).
- ✓ Gestion des logs : La configuration permet de définir les paramètres de journalisation (logs) du serveur web, ce qui permet de suivre les activités du serveur, de détecter les erreurs et les problèmes de performance, et d'analyser les statistiques d'utilisation. (<https://expernet.re/formation/configuration-dun-serveur-web-sous-linux/>).

Il est ici question de la configuration du serveur web sous linux, or on ne peut pas passer à une configuration sans pour autant se rassurer d'avoir installé le serveur dont il est question de configurer, pour enfin finir par le tester. Avant de passer à la configuration vérifiez votre distribution :



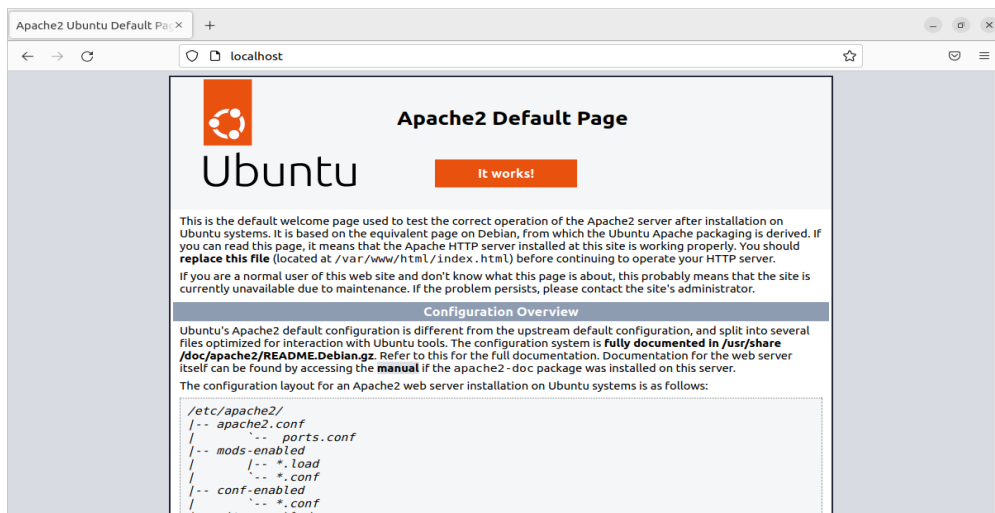
```
m2g@M2G: ~  
m2g@M2G:~$ cat /etc/os-release  
PRETTY_NAME="Ubuntu 22.04 LTS"  
NAME="Ubuntu"  
VERSION_ID="22.04"  
VERSION="22.04 (Jammy Jellyfish)"  
VERSION_CODENAME=jammy  
ID=ubuntu  
ID_LIKE=debian  
HOME_URL="https://www.ubuntu.com/"  
SUPPORT_URL="https://help.ubuntu.com/"  
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"  
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"  
UBUNTU_CODENAME=jammy  
m2g@M2G:~$
```

ETAPES DE LA CONFIGURATION DU SERVEUR WEB SOUS LINUX

Voici les étapes principales pour configurer un serveur web sous Linux :

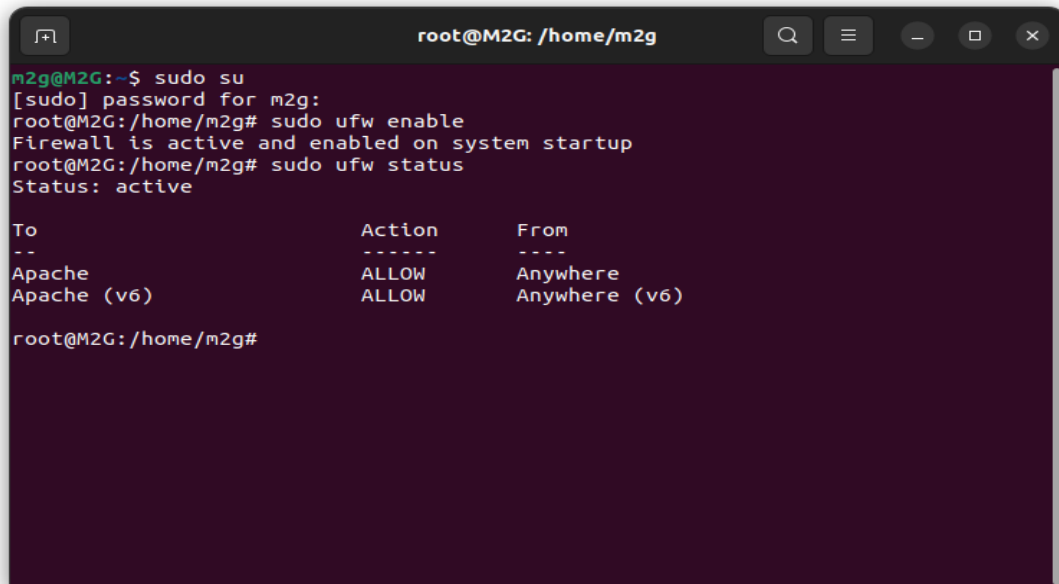
1. LA MISE A JOUR DES BIBLIOTHEQUES : cette première étape est très importante, car il rassurera la bonne continuité des autres étapes. Pour ce faire :
 - Ouvrez un terminal.
 - Exécutez la commande « **sudo apt update** » pour mettre à jour le gestionnaire de paquets.
2. INSTALLATION APACHE : il s'agit ici d'installer les composants nécessaires pour la configuration de notre serveur web.
 - Dans le terminal, exécutez la commande « **sudo apt install apache2** » pour installer Apache (<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-22-04>).
 - Lorsque vous êtes invité à confirmer l'installation, appuyez sur `Y` puis sur `ENTER` (<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-22-04>).
 - Après avoir fini l'installation vous pourrez la vérifier qu'il est installé en tapant juste : **localhost** dans le navigateur ou encore faire dans le terminal :

« **Sudo systemctl status apache2** »



3. CONFIGURATION DU PARE-FEU POUR APACHE : La configuration du pare-feu pour « **apache** » est essentielle pour assurer la sécurité de votre serveur web et de vos applications.
 - Exécutez la commande « **sudo ufw allow in Apache** » pour autoriser le trafic HTTP sur le port 80. (<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-22-04>).
 - Vérifiez que le pare-feu a été configuré correctement en exécutant la commande « **sudo ufw status** » si vous obtenez le message : *status inactive* lorsque vous exécutez cette commande, cela signifie que le pare-feu **UFW** est actuellement désactivé. Par conséquent, les règles de pare-feu que vous avez pu configurer avec « **sudo ufw allow in Apache** » ne sont pas appliquées.
 - Pour activer le pare-feu **UFW**, utiliser la commande « **sudo ufw enable** », Vous devriez avoir le message : « **firewall is active and enabled on system startup** »

indiquant que le pare-feu est activé. Une fois activé, vous pouvez à nouveau exécuter « **sudo ufw status** » pour vérifier l'état et le règle du pare-feu.

A terminal window titled 'root@M2G: /home/m2g' showing the execution of 'sudo su' to become root, then 'sudo ufw enable' to enable the firewall, and 'sudo ufw status' to check its status. The status is 'active'. A table of rules is displayed, showing a rule for Apache (v6) allowing traffic from anywhere.

```
m2g@M2G:~$ sudo su
[sudo] password for m2g:
root@M2G:/home/m2g# sudo ufw enable
Firewall is active and enabled on system startup
root@M2G:/home/m2g# sudo ufw status
Status: active

To Action From
--
Apache ALLOW Anywhere
Apache (v6) ALLOW Anywhere (v6)

root@M2G:/home/m2g#
```

4. INSTALLATION DE MySQL :

- Dans le terminal, exécutez la commande « **sudo apt install mysql-server** » pour installer MySQL. (<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-22-04>).
- Lorsque vous êtes invité à confirmer l'installation, appuyez sur 'Y' puis sur 'ENTER' (<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-22-04>).

5. CONFIGURATION DE MySQL :

- Entrer la commande « **sudo mysql** » pour ensuite utiliser la base **USE mysql** ; c'est la base de donnée stockée dans le serveur. Nous avons ici la possibilité de configurer plusieurs choses. (<https://youtu.be/nouvelle-techno.fr>)

A terminal window titled 'm2g@M2G: ~' showing the execution of 'sudo mysql' to enter the MySQL monitor. It displays the MySQL version (8.0.35-0ubuntu0.22.04.1) and prompts the user to enter a password. The user enters 'mysql', and the prompt changes to 'mysql>'. The user then enters 'USE mysql;', and the prompt changes to 'mysql>'.

```
m2g@M2G:~$ sudo mysql
[sudo] password for m2g:
Welcome to the MySQL Monitor.  Commands end with ; or \g.
Your MySQL connection id is 23
Server version: 8.0.35-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql>
mysql> USE mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql>
```

- En entrant la requête ***select host, user, plugin FROM user*** nous allons modifier le plugin d'authentification « ***auth_socket*** » à « ***MySQL native password*** » pour de raison de compatibilité générale. Ce plugin stocke les informations d'identification de l'utilisateur dans la base de données MySQL elle-même et utilise des méthodes d'authentification basée sur un mot de passe spécifique à MySQL. Commençons par entrer **SHOW DATABASES** (voir l'image)

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| phpmyadmin |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> select host, user, plugin FROM user;
+-----+-----+-----+
| host | user | plugin |
+-----+-----+-----+
| localhost | debian-sys-maint | caching_sha2_password |
| localhost | mysql.infoschema | caching_sha2_password |
| localhost | mysql.session | caching_sha2_password |
| localhost | mysql.sys | caching_sha2_password |
| localhost | phpmyadmin | caching_sha2_password |
| localhost | root | auth_socket |
+-----+-----+-----+
6 rows in set (0.00 sec)
```

```
mysql> UPDATE user SET plugin="mysql_native_password" WHERE user="root";
Query OK, 1 row affected (0.09 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select host, user, plugin FROM user;
+-----+-----+-----+
| host | user | plugin |
+-----+-----+-----+
| localhost | debian-sys-maint | caching_sha2_password |
| localhost | mysql.infoschema | caching_sha2_password |
| localhost | mysql.session | caching_sha2_password |
| localhost | mysql.sys | caching_sha2_password |
| localhost | phpmyadmin | caching_sha2_password |
| localhost | root | mysql_native_password |
+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.02 sec)

mysql> exit
Bye
```

6. INSTALLATION DE PHP :

- Dans le terminal, exécutez la commande « **sudo apt install php libapache2-mod-php php-mysql** » pour installer PHP et les modules nécessaires
- Pour les modules vous pouvez faire « **sudo apt install php-curl php-gd php-intl php-json php-mbstring php-xml php-zip** »
L'installation des modules pour PHP dans la configuration d'un serveur web est essentielle pour permettre au serveur d'exécuter des applications PHP et de prendre en charge les fonctionnalités spécifiques de ces applications.
- Lorsque vous êtes invité à confirmer l'installation, appuyez sur `Y` puis sur **ENTER** (<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-22-04>).

7. CONFIGURATION DU phpMyAdmin :

Dans la configuration de phpMyAdmin, on accède au fichier de configuration (**config.inc.php**) pour décommenter la ligne **`$cfg['Servers'][$i]['AllowNoPassword'] = TRUE;`** en utilisant un éditeur de texte comme Nano. Voici comment :

- Accéder au fichier de configuration :
 - Le fichier de configuration de phpMyAdmin, **config.inc.php**, contient les paramètres de configuration pour l'application.
 - Ce fichier est généralement situé dans le répertoire **/etc/phpmyadmin/** sur les systèmes Linux.
- Utiliser un éditeur de texte comme Nano :
 - Nano est un éditeur de texte en ligne de commande couramment utilisé sur les systèmes Linux, Il permet de modifier les fichiers de configuration de manière simple et rapide.
- Décommenter la ligne **`$cfg['Servers'][$i]['AllowNoPassword'] = TRUE;`** :
 - La ligne **`$cfg['Servers'][$i]['AllowNoPassword'] = TRUE;`** permet d'autoriser la connexion **MySQL/MariaDB** sans mot de passe.
- Pour décommenter la ligne avec Nano :
 - Ouvrez le fichier de configuration avec Nano en utilisant la commande suivante dans le terminal : « **sudo nano /etc/phpmyadmin/config.inc.php** »
(<https://www.simplified.guide/phpmyadmin/enable-login-without-password>).
 - Recherchez la ligne **`$cfg['Servers'][$i]['AllowNoPassword'] = TRUE;`** dans le fichier.
 - Supprimez les **//** au début de la ligne pour la décommenter.
 - Enregistrez les modifications et quittez Nano en appuyant sur **Ctrl + X**, puis en confirmant les modifications.

```
GNU nano 6.2 /etc/phpmyadmin/config.inc.php
/* Select mysql if your server has it */
$cfg['Servers'][$i]['extension'] = 'mysql';
/* Optional: User for advanced features */
$cfg['Servers'][$i]['controluser'] = $dbuser;
$cfg['Servers'][$i]['controlpass'] = $dbpass;
/* Optional: Advanced phpMyAdmin features */
$cfg['Servers'][$i]['pmadb'] = $dbname;
$cfg['Servers'][$i]['bookmarktable'] = 'pma_bookmark';
$cfg['Servers'][$i]['relation'] = 'pma_relation';
$cfg['Servers'][$i]['table_info'] = 'pma_table_info';
$cfg['Servers'][$i]['table_coords'] = 'pma_table_coords';
$cfg['Servers'][$i]['pdf_pages'] = 'pma_pdf_pages';
$cfg['Servers'][$i]['column_info'] = 'pma_column_info';
$cfg['Servers'][$i]['history'] = 'pma_history';
$cfg['Servers'][$i]['table_uiprefs'] = 'pma_table_uiprefs';
$cfg['Servers'][$i]['tracking'] = 'pma_tracking';
$cfg['Servers'][$i]['userconfig'] = 'pma_userconfig';
$cfg['Servers'][$i]['recent'] = 'pma_recent';
$cfg['Servers'][$i]['favorite'] = 'pma_favorite';
$cfg['Servers'][$i]['users'] = 'pma_users';
$cfg['Servers'][$i]['usergroups'] = 'pma_usergroups';
$cfg['Servers'][$i]['navigationhiding'] = 'pma_navigationhiding';
$cfg['Servers'][$i]['savedsearches'] = 'pma_savedsearches';
$cfg['Servers'][$i]['central_columns'] = 'pma_central_columns';
$cfg['Servers'][$i]['designer_settings'] = 'pma_designer_settings';
$cfg['Servers'][$i]['export_templates'] = 'pma_export_templates';

/* Uncomment the following to enable logging in to passwordless accounts,
 * after taking note of the associated security risks. */
// $cfg['Servers'][$i]['AllowNoPassword'] = TRUE;
```

Une fois que vous avez décommenté la ligne `$cfg['Servers'][$i]['AllowNoPassword'] = TRUE;` dans le fichier de configuration de phpMyAdmin, vous pourrez vous connecter à MySQL/MariaDB sans mot de passe. (<https://stackoverflow.com/questions/5908752/can-anyone-confirm-that-phpmyadmin-allownopassword-works-with-mysql-databases>) (<https://www.liquidweb.com/kb/error-login-without-a-password-is-forbidden-by-configuration-see-allownopassword-solved/>)

CAS DE NON ACCES A phpmyadmin

Il se peut qu'après avoir configuré votre serveur web, vous rencontrez certains problèmes notamment le problème d'accès à phpmyadmin au niveau de votre navigateur.

Voici les étapes à suivre pour une bonne configuration :

- ✓ Ouvrez le fichier de configuration d'Apache avec un éditeur de texte. Selon votre distribution, le fichier de configuration peut être situé à des emplacements différents. Voici quelques exemples :

Ubuntu/Debian : dans notre cas par exemple nous ferons :

« `sudo nano /etc/apache2/apache2.conf` »

CentOS/Fedora :

« `sudo nano /etc/httpd/conf/httpd.conf` »

- ✓ Recherchez la ligne contenant la directive ***IncludeOptional***. Elle ressemblera à ceci :

IncludeOptional `chemin/vers/le/fichier.conf`

Dans le processus de directive ***IncludeOptional***, il est préférable lors de la configuration de choisir celui qui correspondra à vos besoins.

Les directives ***IncludeOptional conf-enabled/*.conf*** et ***IncludeOptional sites-enabled/*.conf*** sont utilisées dans la configuration d'Apache pour inclure des fichiers de configuration à partir de certains répertoires spécifiques. La différence réside dans les types de fichiers de configuration inclus.

- ***IncludeOptional conf-enabled/*.conf*** : Cette directive est généralement utilisée pour inclure des fichiers de configuration globaux qui s'appliquent à tous les sites hébergés sur le serveur. Ces fichiers de configuration peuvent contenir des directives telles que des paramètres globaux d'Apache, des modules supplémentaires, etc. Les fichiers inclus dans le répertoire ``conf-enabled`` doivent avoir l'extension ``.conf``.
- ***IncludeOptional sites-enabled/*.conf*** : Cette directive est utilisée pour inclure des fichiers de configuration spécifiques à chaque site hébergé sur le serveur. Chaque fichier inclus correspond à un site virtuel (**VirtualHost**) spécifique. Les fichiers inclus dans le répertoire ``sites-enabled`` doivent également avoir l'extension ``.conf``.

En fonction de votre configuration spécifique et de l'endroit où se trouve le fichier de configuration de **phpMyAdmin**, vous devez choisir l'une de ces directives pour inclure le fichier de configuration approprié. Si le fichier de configuration de phpMyAdmin est un fichier de configuration global qui s'applique à tous les sites, vous pouvez utiliser « ***IncludeOptional conf-enabled/*.conf*** ». Si le fichier de configuration de **phpMyAdmin** est spécifique à un site virtuel, vous pouvez utiliser « ***IncludeOptional sites-enabled/*.conf*** ».

- ✓ Assurez-vous que le chemin spécifié dans la directive ***IncludeOptional*** pointe vers le bon fichier de configuration de phpMyAdmin. Par exemple, si le fichier de configuration de phpMyAdmin est situé dans `/etc/phpmyadmin/apache.conf`, modifiez la ligne pour qu'elle ressemble à ceci :

« ***IncludeOptional /etc/phpmyadmin/apache.conf*** »

- ✓ Enregistrez les modifications et fermez le fichier de configuration.
- ✓ Redémarrez le serveur Apache pour que les modifications prennent effet. Vous pouvez utiliser la commande suivante selon votre distribution :

Ubuntu/Debian : dans notre cas,

« **`sudo service apache2 restart`** »

CentOS/Fedora :

« **`sudo systemctl restart httpd`** »

Après avoir effectué ces étapes, essayez à nouveau d'accéder à **localhost/phpmyadmin/** dans votre navigateur. Cela devrait vous permettre d'accéder à l'interface de phpMyAdmin.

TESTE

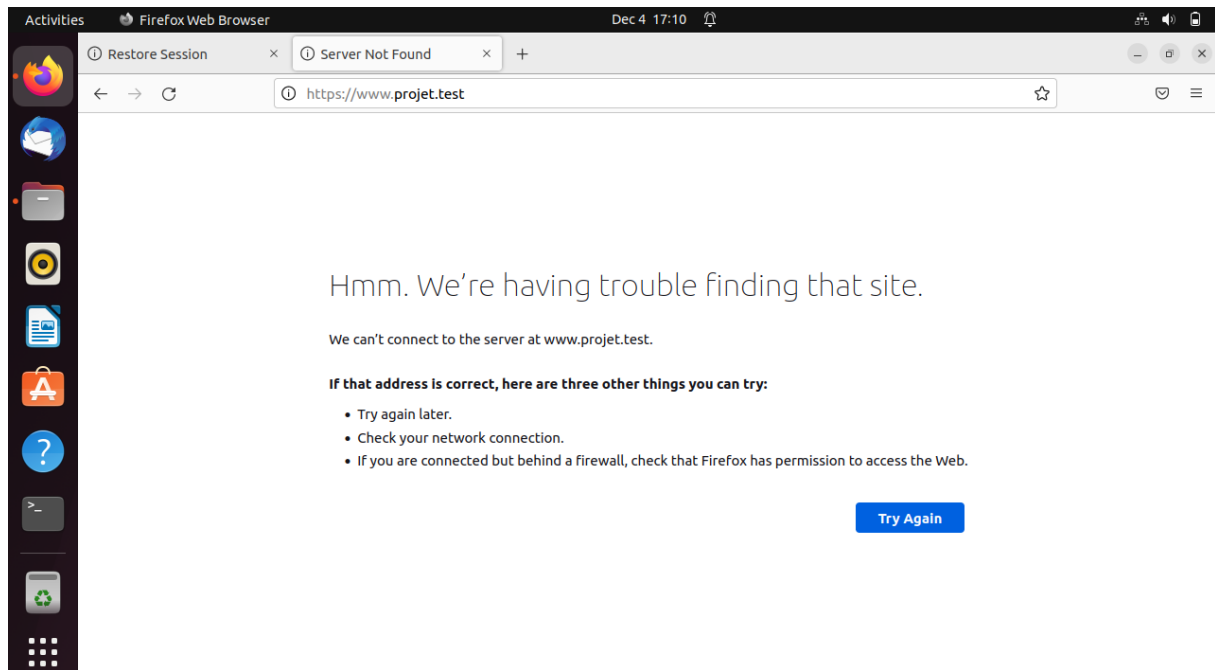
On souhaite avoir un site en local dénommé : **projet.test**

Pour ce faire, on va aller en sudo pour créer le dossier ou encore le répertoire de notre site, en faisant : « **sudo mkdir/var/www/projet.test** », puis dans ce dossier nous allons par exemple introduire une page web **index.html** que nous pourrons au niveau de nano entrer quelque ligne des balises.

```
root@M2G: /home/m2g
m2g@M2G:~$ sudo su
root@M2G:/home/m2g# sudo nano /var/www/projet.test/index.html
root@M2G:/home/m2g# sudo nano /var/www/projet.test/index.html
```

```
GNU nano 6.2 /var/www/projet.test/index.html
<h1>PROJET DE SYSTEME D'EXPLOITATION</h1>
<h2>CONFIGURATION DU SERVEUR WEB SOUS LINUX </h2>
<P>Ce projet consiste à faire la configuration du serveur web sous linux.<br>
cette page est juste un essai après avoir configuré notre serveur web.<br>
Ce travail est réalisé par : </P>
<li>MASELA KAYA GEMIMA </li>
<li>MOMBESA GLODI </li>
<li>MASENGA EXAUCE </li>
<li>MUMBE BENJAMIN </li>
<li>MUDIOLO AMEDEE</li>
<li>NAKALENGA AVU</li>
<li>MALANGONAY LOIS</li>
<li>NSANDUNU ESTHER</li>
<li>BYAMUNGU KEVIN</li>
<li>MALEBA SANG PRECIEUX</li>
<p></p>
<p></p>
<p></p>
<h3>MERCI!!!!</h3>
[ Read 19 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut      ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste    ^J Justify  ^_ Go To Line
```

Pour enregistrer **^x** puis **ENTER**. Nous pourrons à présent se rendre dans le navigateur, sauf que ce dernier va nous renvoyer des erreurs.



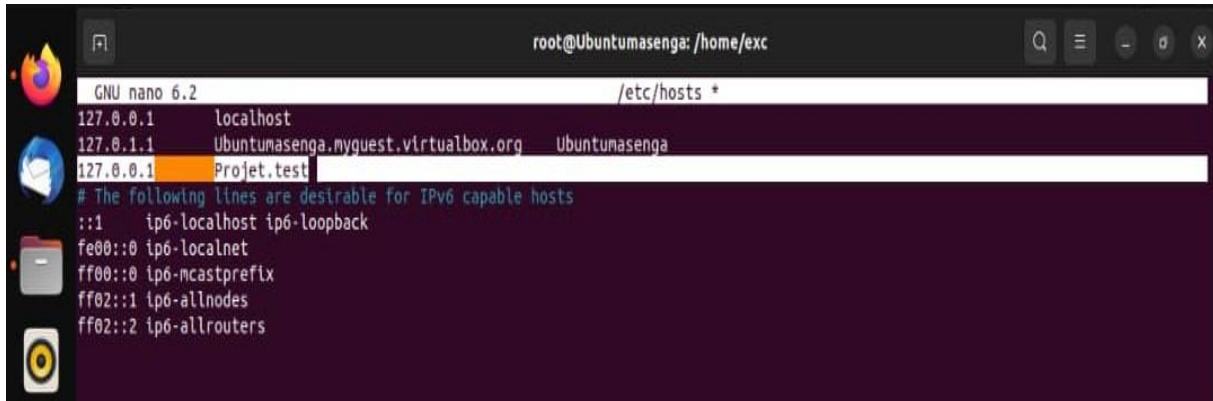
Au niveau d'apache on va créer une nouvelle configuration pour notre site en lui donnant une adresse bien particulière. Pour ce faire : « **sudo nano/etc/apache2/sites-available/projet.test.conf** ». **sites-available** est un dossier qui se trouve dans **apache2** qui nous permet de lister tous les sites disponibles sur notre machine. C'est dans ce fichier qu'on va faire du Virtual Host. L'utilisation de Virtual Host permet de créer des hôtes virtuels, c'est-à-dire des sites ou des domaines individuels, qui peuvent être servis par une seule instance d'apache. Chaque hôte virtuel est configuré pour diriger les visiteurs vers un répertoire spécifique contenant les informations des sites correspondant.

```
GNU nano 6.2 /etc/apache2/sites-available/Projet.test.conf *
<virtualHost *:80>
  ServerName Projet.test
  ServerAlias www.Projet.test
  DocumentRoot "/var/www/Projet.test"
  <Directory "/var/www/Projet.test">
    Options FollowSymLinks
    AllowOverride all
    Require all granted
  </Directory>
  ErrorLog /var/log/apache2/error.Projet.test.log
  CustomLog /var/log/apache2/access.Projet.test.log combined
</virtualHost>
```

AG Help PO Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark
AX Exit ^R Read File ^_ Replace ^U Paste ^D Justify ^_ Go To Line M-E Redo M-G Copy

Après avoir enregistré, nous pourrions activer le site en faisant la commande : « **sudo a2ensite projet.test** » puis recharger en faisant « **sudo systemctl reload apache2** »

Et comme notre site ne correspond pas à une adresse IP d'un serveur hébergeant un domaine spécifique, nous ferons donc appel au **Host**. Il permet de créer de redirections de noms d'hôtes personnalisés vers des adresses IP spécifiées. Pour ce faire : « **sudo nano /etc/hosts** »



```
root@UbuntuMasenga: /home/exc
GNU nano 6.2 /etc/hosts *
127.0.0.1 localhost
127.0.1.1 UbuntuMasenga.myquest.virtualbox.org UbuntuMasenga
127.0.0.1 projet.test
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Après avoir fini cette modification on enregistre en faisant **Ctrl + X** puis **ENTER**.

Nous pourrions à nouveau nous rendre dans notre navigateur pour pouvoir tester à nouveau :



Tout marche maintenant comme voulu !!

En conclusion, il est important de savoir que la configuration d'un serveur web dépend de besoin de tout un chacun, d'où certaines spécificités lors de cette configurations n'ont pas été considérées. Il est aussi important de savoir que c'est le mode super utilisateur qui nous donne la possibilité de faire cette configuration, car il nous donne la possibilité d'aller un peu plus loin.

