

Administration Linux

M2i Formation – TSSR 2020

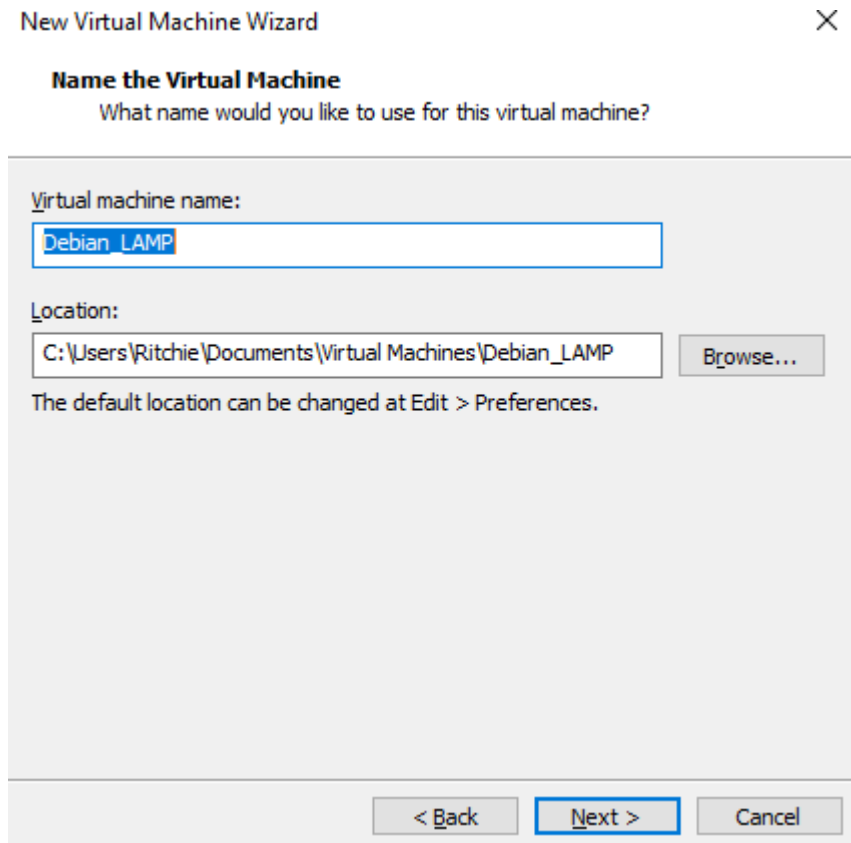
MISE EN PLACE D'UNE INFRASTRUCTURE
RESEAU SOUS LINUX

Messagerie – Pare-Feu - DHCP – DNS

INSTALLATION D'UN SERVEUR LAMP

Remarque : L'acronyme LAMP désigne un ensemble de quatre technologies open source : un système d'exploitation Linux, un serveur web Apache, un système de bases de données MySQL et le langage de programmation PHP.

Ici, on choisira un nom pour notre serveur LAMP



Le reste de l'installation est inchangée par rapport à l'installation de Debian lorsque l'on monte l'ISO.
Toujours vérifier que la carte réseau est en bridge ou NAT.

Notre serveur est prêt pour la configuration LAMP. On vérifie son kernel avec `uname -a` (all)

```
debian@LAMP:~$
debian@LAMP:~$ uname -a
Linux LAMP 4.19.0-8-amd64 #1 SMP Debian 4.19.98-1 (2020-01-26) x86_64 GNU/Linux
debian@LAMP:~$ _
```

Pour commencer on va regarder pour mettre jour notre système

```
debian@LAMP:~$
debian@LAMP:~$ uname -a
Linux LAMP 4.19.0-8-amd64 #1 SMP Debian 4.19.98-1 (2020-01-26) x86_64 GNU/Linux
debian@LAMP:~$ apt update
Lecture des listes de paquets... Fait
E: Impossible d'ouvrir le fichier verrou /var/lib/apt/lists/lock - open (13: Permission non accordée)
E: Impossible de verrouiller le répertoire /var/lib/apt/lists/
debian@LAMP:~$ sudo apt update
-bash: sudo : commande introuvable
debian@LAMP:~$ su -
Mot de passe :
root@LAMP:~# apt update
Atteint :1 http://security.debian.org/debian-security buster/updates InRelease
Atteint :2 http://deb.debian.org/debian buster InRelease
Atteint :3 http://deb.debian.org/debian buster-updates InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
root@LAMP:~# apt full-upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@LAMP:~#
```

Installation de LAMP sur le serveur Debian

Installer Apache

On va utiliser le apt package manager pour l'install :

apt install apache2

Puis on vérifie si Apache est en train de tourner

```

Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
root@LAMP:~# apt full-upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@LAMP:~# apt install apchache2
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
E: Impossible de trouver le paquet apchache2
root@LAMP:~# systectl status apache2
-bash: systectl : commande introuvable
root@LAMP:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2020-03-23 10:53:05 CET; 13min ago
     Docs: https://httpd.apache.org/docs/2.4/
  Process: 394 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 410 (apache2)
    Tasks: 55 (limit: 1120)
   Memory: 12.0M
    CGroup: /system.slice/apache2.service
           └─410 /usr/sbin/apache2 -k start
             └─412 /usr/sbin/apache2 -k start
               └─413 /usr/sbin/apache2 -k start

mars 23 10:53:05 LAMP systemd[1]: Starting The Apache HTTP Server...
mars 23 10:53:05 LAMP apachectl[394]: AH00557: apache2: apr_socketaddr_info_get() failed for LAMP
mars 23 10:53:05 LAMP apachectl[394]: AH00558: apache2: Could not reliably determine the server's fu
mars 23 10:53:05 LAMP systemd[1]: Started The Apache HTTP Server.

```

`systemctl status apache2`

Attention les port 80 et 43 doivent être ouverts (Firewall)

Installer MariaDB

`apt install mariadb-server`

```

root@LAMP:~# apt install mariadb-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  galera-3 gauk libaio1 libcg1-fast-perl libcg1-pm-perl libconfig-inifiles-perl libdbd-mysql-perl
  libdbi-perl libencode-locale-perl libfcgi-perl libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
  liblwp-mediatypes-perl libmariadb3 libmfr6 libreadline5 libsigsegv2 libsnappy1v5
  libterm-readkey-perl libtimedate-perl liburi-perl mariadb-client-10.3 mariadb-client-core-10.3
  mariadb-common mariadb-server-10.3 mariadb-server-core-10.3 mysql-common psmisc rsync socat
Paquets suggérés :
  gauk-doc libclone-perl libmldbm-perl libnet-daemon-perl libsql-statement-perl libdata-dump-perl
  libipc-sharedcache-perl libwww-perl mailx mariadb-test netcat-openbsd tinyca
Les NOUVEAUX paquets suivants seront installés :
  galera-3 gauk libaio1 libcg1-fast-perl libcg1-pm-perl libconfig-inifiles-perl libdbd-mysql-perl
  libdbi-perl libencode-locale-perl libfcgi-perl libhtml-parser-perl libhtml-tagset-perl
  libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
  liblwp-mediatypes-perl libmariadb3 libmfr6 libreadline5 libsigsegv2 libsnappy1v5
  libterm-readkey-perl libtimedate-perl liburi-perl mariadb-client-10.3 mariadb-client-core-10.3
  mariadb-common mariadb-server mariadb-server-10.3 mariadb-server-core-10.3 mysql-common psmisc
  rsync socat
0 mis à jour, 35 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 22,4 Mo dans les archives.
Après cette opération, 170 Mo d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [0/n]

```

Puis on fait tourner le script de sécurité qui va permettre de sécuriser mieux notre DBS avec la commande suivante :

`mysql_secure_installation`

Il nous fait répondre à des questions pour améliorer la configuration par défaut :

```
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n]
... Success!

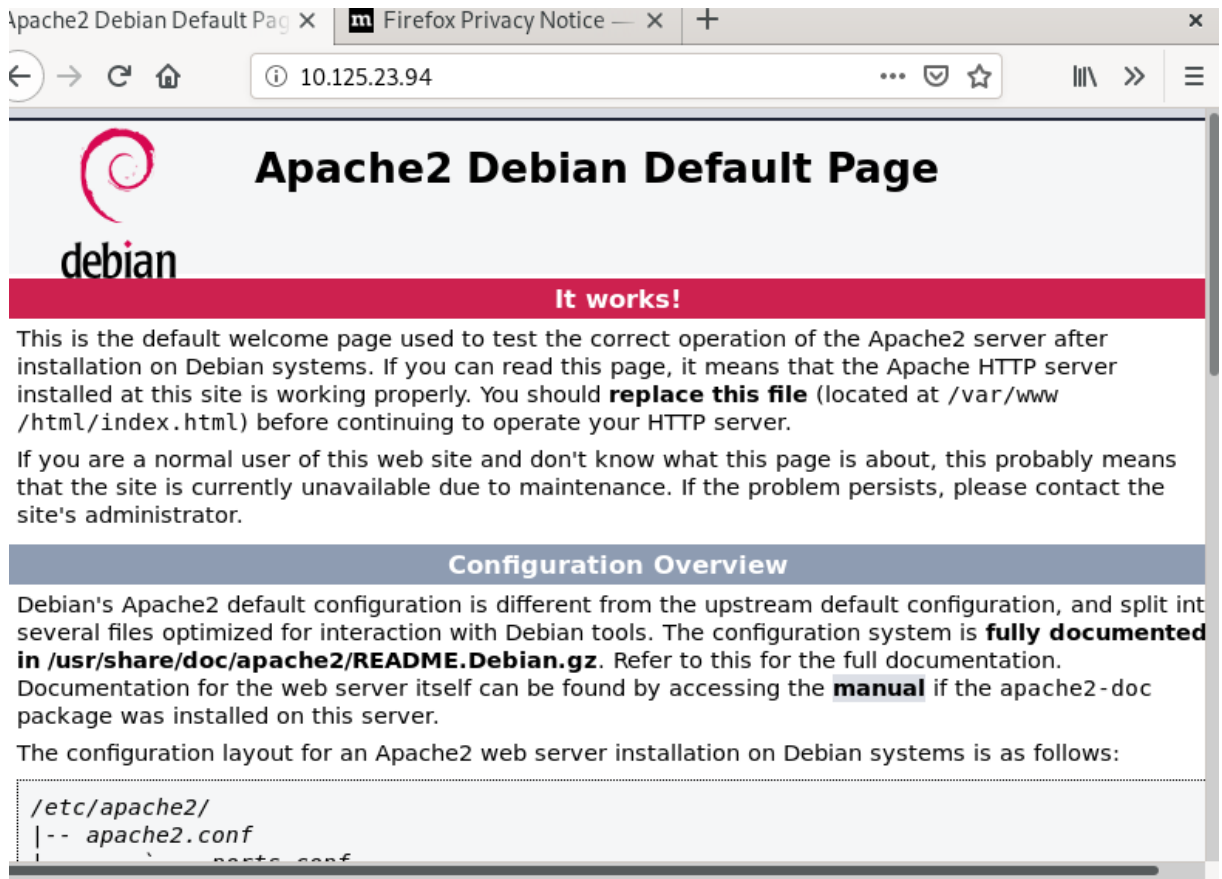
Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
root@LAMP:~#
```

On teste si notre serveur apache fonctionne sur une machine qui a une interface graphique

Pour cela tapez [http:// ip de votre serveur](http://ip de votre serveur)



Pour créer un nouvel utilisateur et DBS on va utiliser les commandes suivantes : on va le faire après avec l'installation de wordpress

Mysql -u root -p

On rentre dans MariaDB puis

create database projet_linux ;

grant all on projet_linux. to projet@localhost identified by Azerty1 with grant option ;*

flush privileges ;

exit ;

Pour vérifier

Mysql -u projet -p

[Installer PHP et Wordpress](#)

apt install php libapache2-mod-php php-mysql

On va créer un DBS pour Wordpress

sudo mysql -u root -p # pour aller dans MariaDB, vous verrez que le nom de la machine change en MariaDB

Puis on crée notre dbs wordpress_db (n'oublie pas les underscorees)

CREATE DATABASE wordpress_db ;

Il faut un utilisateur avec les permissions nécessaires :

utilisateur :wordpress_user

mdp :wordpress

GRANT AL ON wordpress_db. TO 'wordpress_user'@'localhost' IDENTIFIED BY 'wordpress' ;*

Pour confirmer les changements on fait :

FLUSH PRIVILEGES

Et pour sortir

EXIT

Pour installer wordpress il faut des autres extensions avec la commande suivant :

apt install php php-mysql php-curl php-gd php-mbstring php-xml php-xmlrpc php-soap php-intl php-zip

```
root@LAMP:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 53
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)]>
```

Pour que les changements s'effectue on restart apache2

Systemctl restart apache2

Et on télécharge la dernière version de Wordpress : `curl -O https://wordpress.org/latest.tar.gz`

Puis on va extraire : *tar -xvf latest.tar.gz*

```
(Lecture de la base de données... 37474 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../curl_7.64.0-4+deb10u1_amd64.deb ...
Dépaquetage de curl (7.64.0-4+deb10u1) ...
Paramétrage de curl (7.64.0-4+deb10u1) ...
Traitement des actions différées (« triggers ») pour man-db (2.8.5-2) ...
root@LAMP:/var/www/html# curl -O https://wordpress.org/latest.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 11.8M  100 11.8M    0     0  750k      0  0:00:16  0:00:16 --:--:-- 1620k
root@LAMP:/var/www/html#
```

On doit changer les permissions sur les fichiers wordpress

chown -R www-data:www-data /var/www/html/wordpress

find /var/www/html/wordpress/ -type d -exec chmod 750 {} \;

find /var/www/html/wordpress/ -type f -exec chmod 750 {} \;

On renomme notre fichier e config

```
cd wordpress
```

```
sudo mv wp-config-sample.php wp-config.php
```

Et va configurer notre fichier php avec les données de notre database de tout à l'heure

```
nano wp-config.php
```

```
/* ** MySQL settings - You can get this info from your web host ** */
/** The name of the database for WordPress */
define( 'DB_NAME', 'database_name_here' );

/** MySQL database username */
define( 'DB_USER', 'username_here' );

/** MySQL database password */
define( 'DB_PASSWORD', 'password_here' );

/** MySQL hostname */
define( 'DB_HOST', 'localhost' );

/** Database Charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The Database Collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );
```

Il faut configurer Apache pour Wordpress

Dans `/etc/apache2/sites available/000-default.conf` avec `nano`

Dans document/root on rajoute wordpress comme ceci

```
/var/www/html/wordpress/
```

Puis on rajoute les lignes suivantes :

```
<Directory /var/www/html/wordpress/>
```

```
AllowOverride All
```

```
</Directory>
```

Puis on activera l'option permalink sur wordpress

```
a2enmod rewrite
```

Pour tester votre config apache-wordpress

```
Apache2ctl configtest
```

Syntax OK. Pour confirmer les changement vous faites

```
Systemctl restart apache2
```


L'installation depuis une page web

Vous accéder par la page web suivant [http:// ip de votre serveur](http://ip de votre serveur)



