

TP LAMP

Contents

1. Introduction et Installation	1
2. Test.....	2
3. Utilisation	4
4. Apache	9
5. Utilisation de MySQL (MariaDB)	23
1) Introduction	23
2) Installation.....	24
3) Fonctionnement.....	24
4) Commande mysql.....	24

1. Introduction et Installation

Acronyme LAMP:

- Linux : le systeme d'exploitation
- Apache : le serveur HTTP
- MySQL: le serveur de base de données (MariaDB)
- PHP : le langage de script

Installation: `apt install apache2 php mariadb-server libapache2-mod-php php-mysql`

J'ai installer apache2, php, mariadb et MySQL

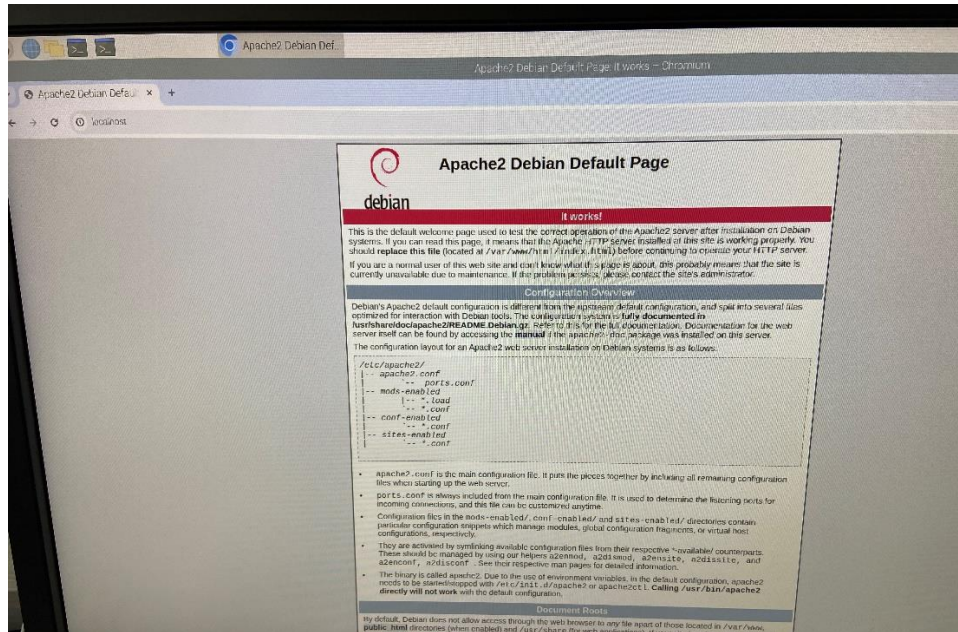
```
jack@raspberrypi:~ $ sudo apt install apache2 php mariadb-server libapache2-mod-php php-mysql
```

Lancement : `service apache2 start`

J'ai démarrer apache2

```
jack@raspberrypi:~ $ service apache2 start
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentification requise pour démarrer « apache2.service ».
Authenticating as:,,, (jack)
Password:
==== AUTHENTICATION COMPLETE ====
jack@raspberrypi:~ $
```

Test : <http://localhost>



2. Test

Base de donnée : `service mysqld start`

```
jack@raspberrypi:~$ service mysqld start
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentification requise pour démarrer « mariadb.service ».
Authenticating as: ,, (jack)
Password:
==== AUTHENTICATION COMPLETE ====
jack@raspberrypi:~$
```

Test console : `sudo mysql -u root -p`

Mot de passe

exit

```
jack@raspberrypi:~$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 33
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12

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)]>
```

J'ai ensuite créé une base de données qui s'appelle « wordpress »

```
MariaDB [(none)]> CREATE DATABASE wordpress
-> ;
Query OK, 1 row affected (0,001 sec)

MariaDB [(none)]> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| performance_schema      |
| sys                     |
| wordpress                |
+-----+
5 rows in set (0,006 sec)

MariaDB [(none)]> 
```

Test PHP dans /var/www/html

Nano phpinfo.php

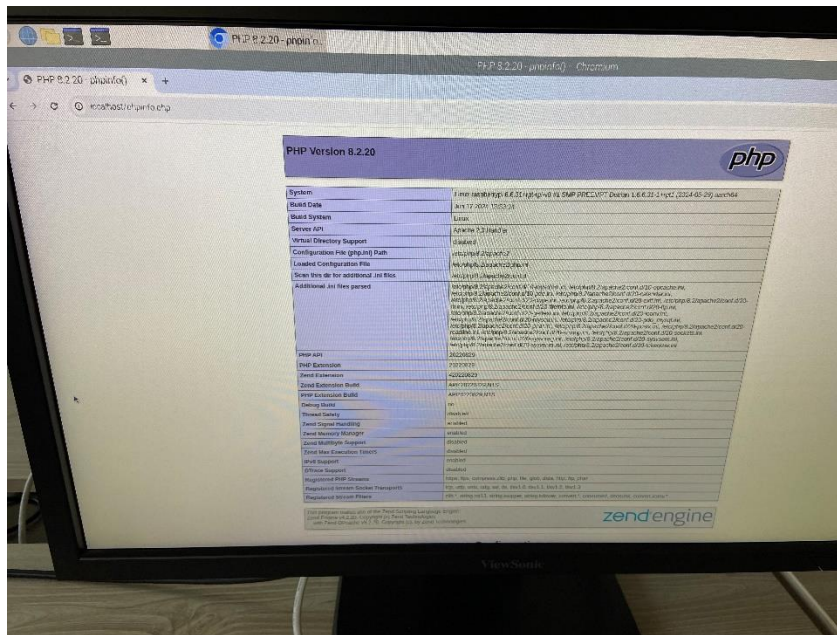
```
<?php
    Phpinfo();

?>
```

```
jack@raspberrypi:/var/www/html $ sudo touch phpinfo.php
jack@raspberrypi:/var/www/html $ ls
index.html  phpinfo.php
```

```
GNU nano 7.2                                phpinfo.php *
<?php
    phpinfo();
?>
```

<http://localhost/phpinfo.php>



Le module mysql est-il installé ?

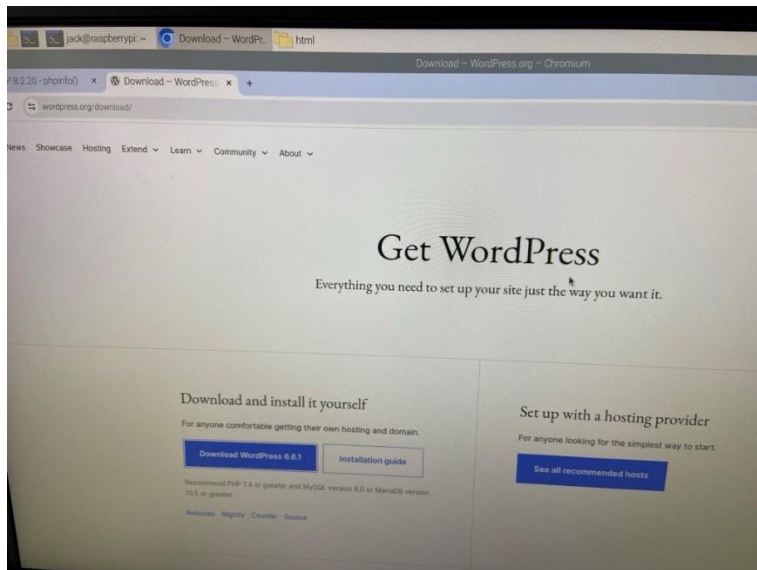
MySQL est installé car on voit le module dans phpinfo.php

mysql		
MySQL Support		enabled
Client API library version	mysqlnd 8.2.20	
Active Persistent Links	0	
Inactive Persistent Links	0	
Active Links	0	
Options		
	Local MySQL	Remote MySQL
mysql.allow_local_infile	Off	Off
mysql.allow_remote_login	On	On
mysql.allow_remote_lock	1 (1=1)	1 (1=1)
mysql.default_host	127.0.0.1	127.0.0.1
mysql.default_port	3306	3306
mysql.default_socket	/var/lib/mysql/mysql.sock	/var/lib/mysql/mysql.sock
mysql.default_user	root	root
mysql.local_infile_directory	/var/lib/mysql	/var/lib/mysql
mysql.max_connections	Unlimited	Unlimited
mysql.max_persistent	Unlimited	Unlimited
mysql.max_backlog_on_cached_plink	Off	On
mysqlnd		
mysqlnd		enabled
Version	mysqlnd 8.2.20	
Compression	supported	
core SSL	supported	
extended SSL	supported	
Command buffer size	4096	
Read buffer size	32768	
Read timeout	65535	
Collecting statistics	Yes	
Collecting memory statistics	No	

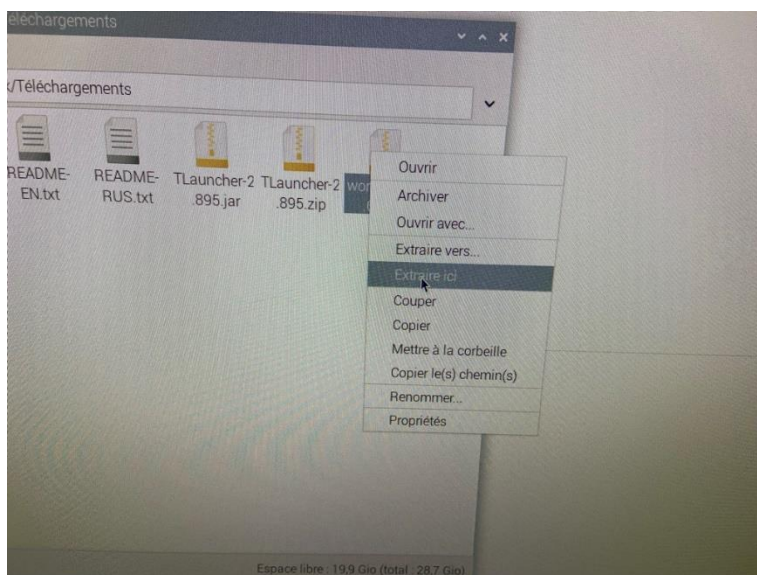
3. Utilisation

- On peut installer phpmyadmin pour administrer graphiquement les bases de données : `apt install phpmyadmin`
Test : <http://localhost/phpmyadmin>
- On install Wordpress
 - Télécharger l'archive
 - Le décompresser dans le dossier /var/www/html
 - <http://localhost/wordpress>

On télécharge l'archive .zip sur le site web de wordpress (<https://fr.wordpress.org/download/>)



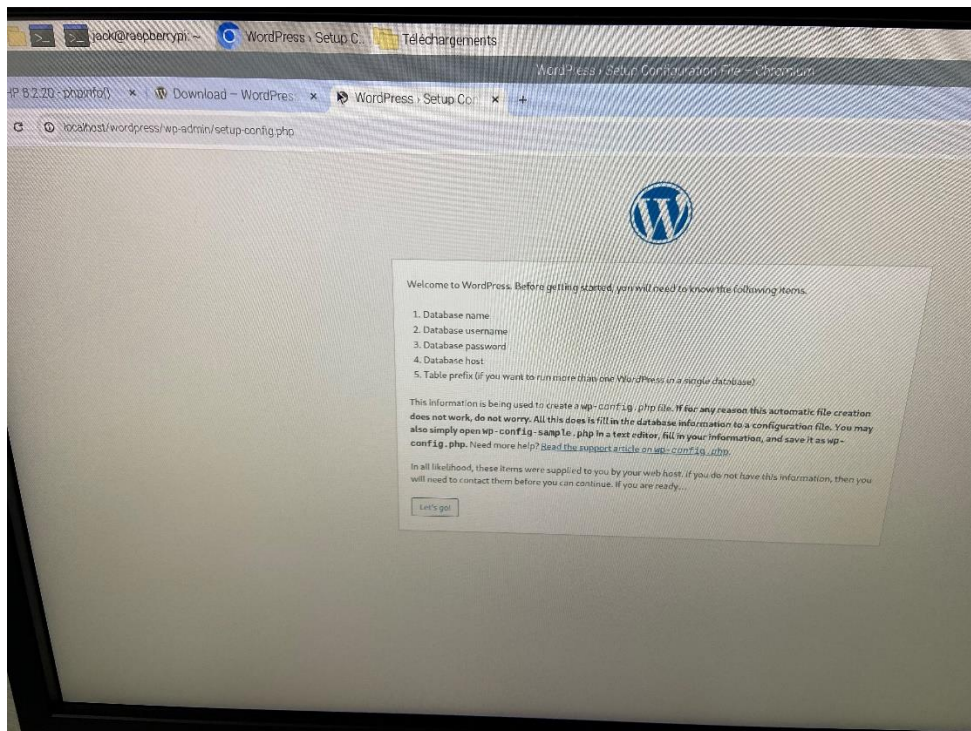
Ensuite j'extrais le dossier dans /home/jack/Téléchargement



Ensuite on utilise la commande cp -r pour copier le fichier dans /var/www/html

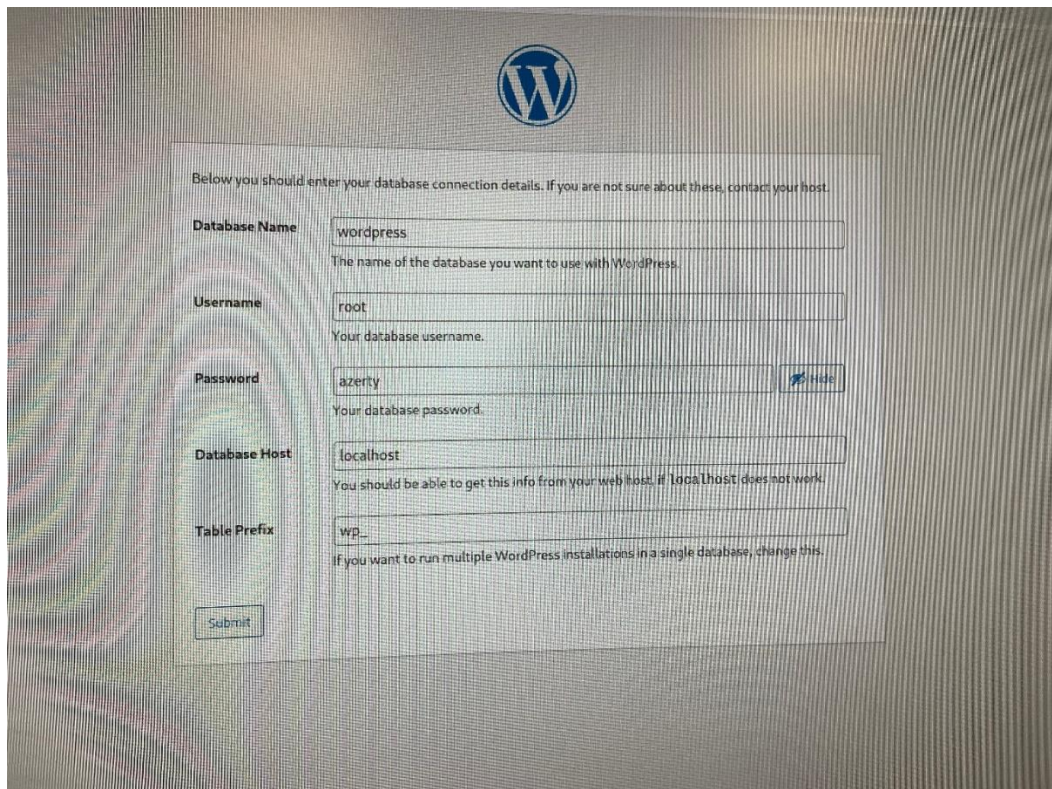
```
jack@raspberrypi:~/Téléchargements $ sudo cp -r wordpress /var/www/html/  
jack@raspberrypi:~/Téléchargements $ cd /var/www/html  
jack@raspberrypi:/var/www/html $ ls  
index.html  phpinfo.php  wordpress
```

Ensuite sur me navigateur du raspberry on tape <http://localhost/wordpress/wp-admin/setupconfig.php>

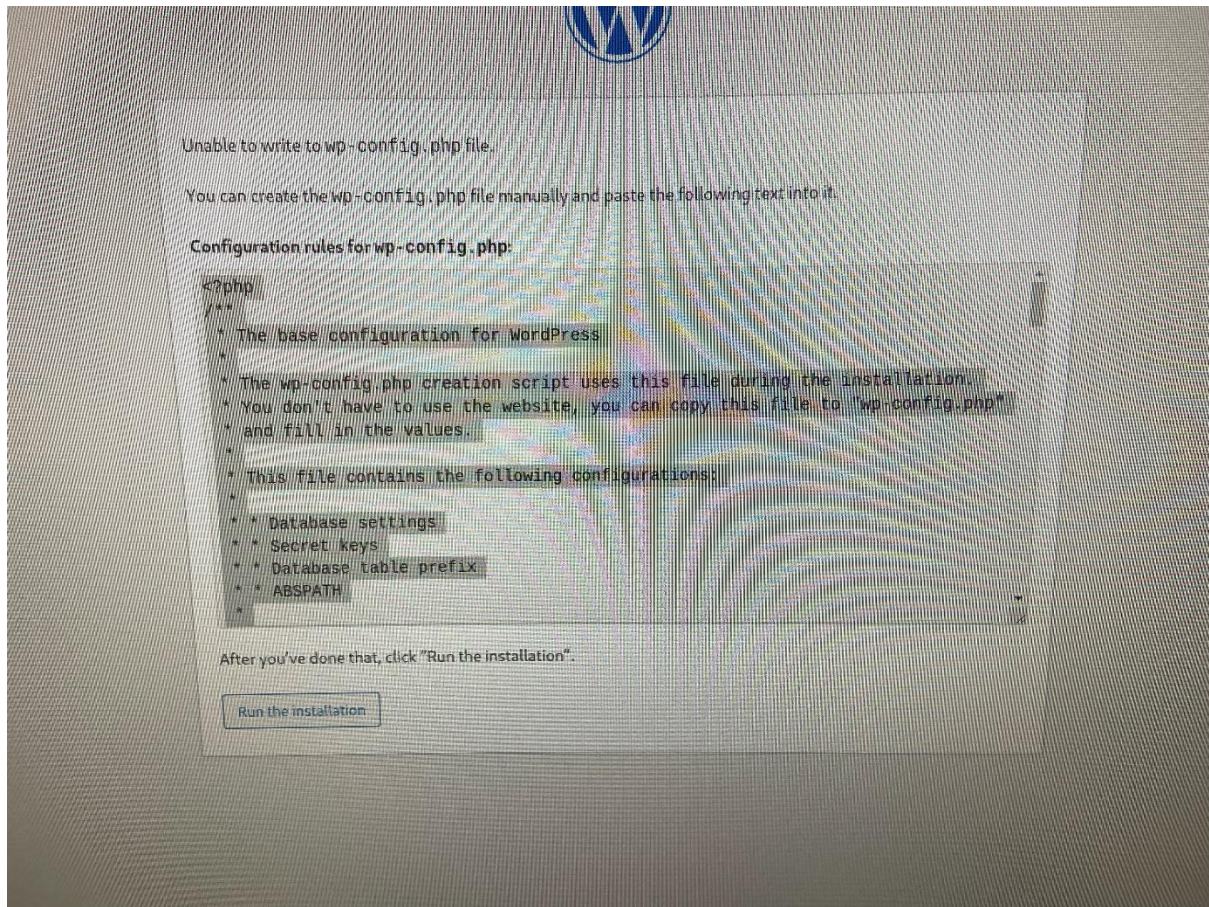


On click sur « Let's go ! »

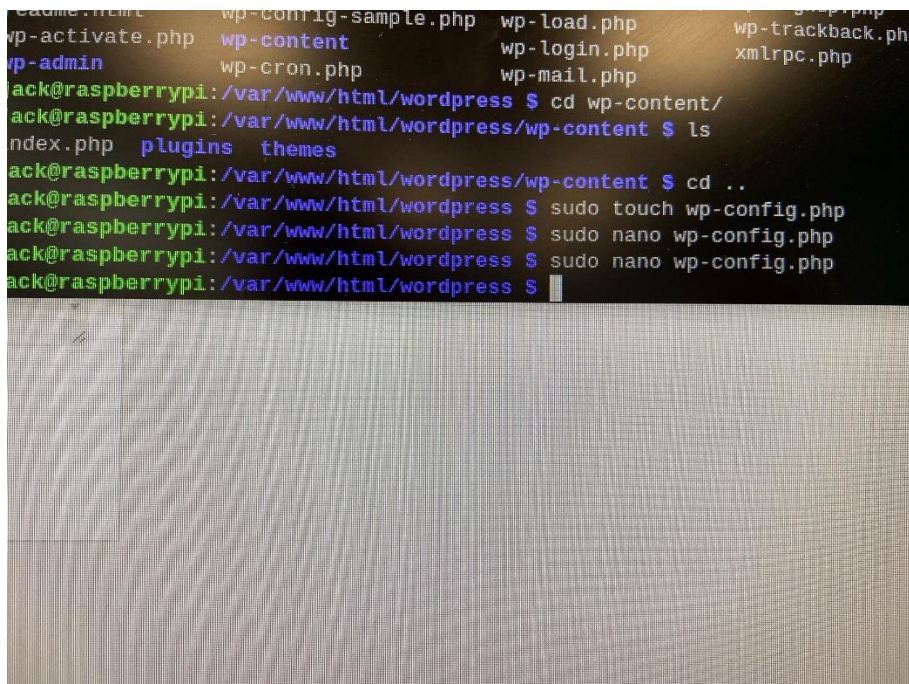
Et on rentre les informations ci-dessous.



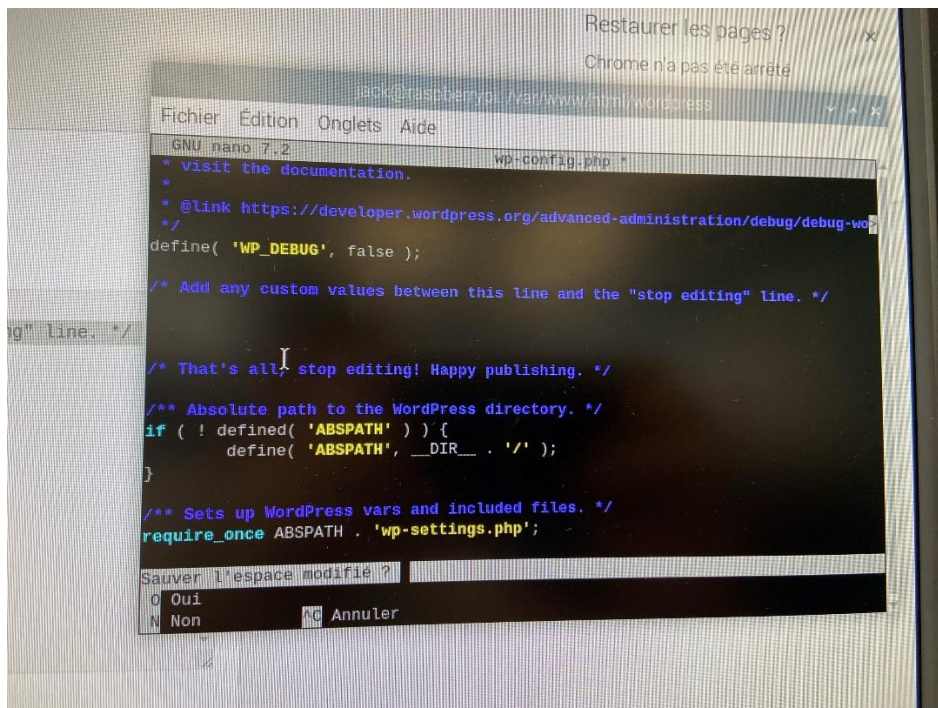
Ensuite on a cette page web :



On va ensuite dans les fichier /var/www/html/wordpress pour créer le fichier wp-content.php



On copy les configuration dans le dossier wp-content.php



Ensuite on a lancer l'installation et remplis les information ci-dessous.

Bienvenue

Bienvenue dans la très célèbre installation en 5 minutes de WordPress ! Vous n'avez qu'à remplir les informations demandées ci-dessous et vous serez prêt à utiliser la plus extensible et puissante plateforme de publication de contenu au monde.

Informations nécessaires

Veuillez renseigner les informations suivantes. Ne vous inquiétez pas, vous pourrez les modifier plus tard.

Titre du site

Identifiant

Les identifiants ne peuvent utiliser que des caractères alphanumériques, des espaces, des tirets bas (" _ "), des traits d'union (" - "), des points et le symbole @.

Mot de passe

Forte

Important : Vous aurez besoin de ce mot de passe pour vous connecter. Pensez à le stocker dans un lieu sûr.

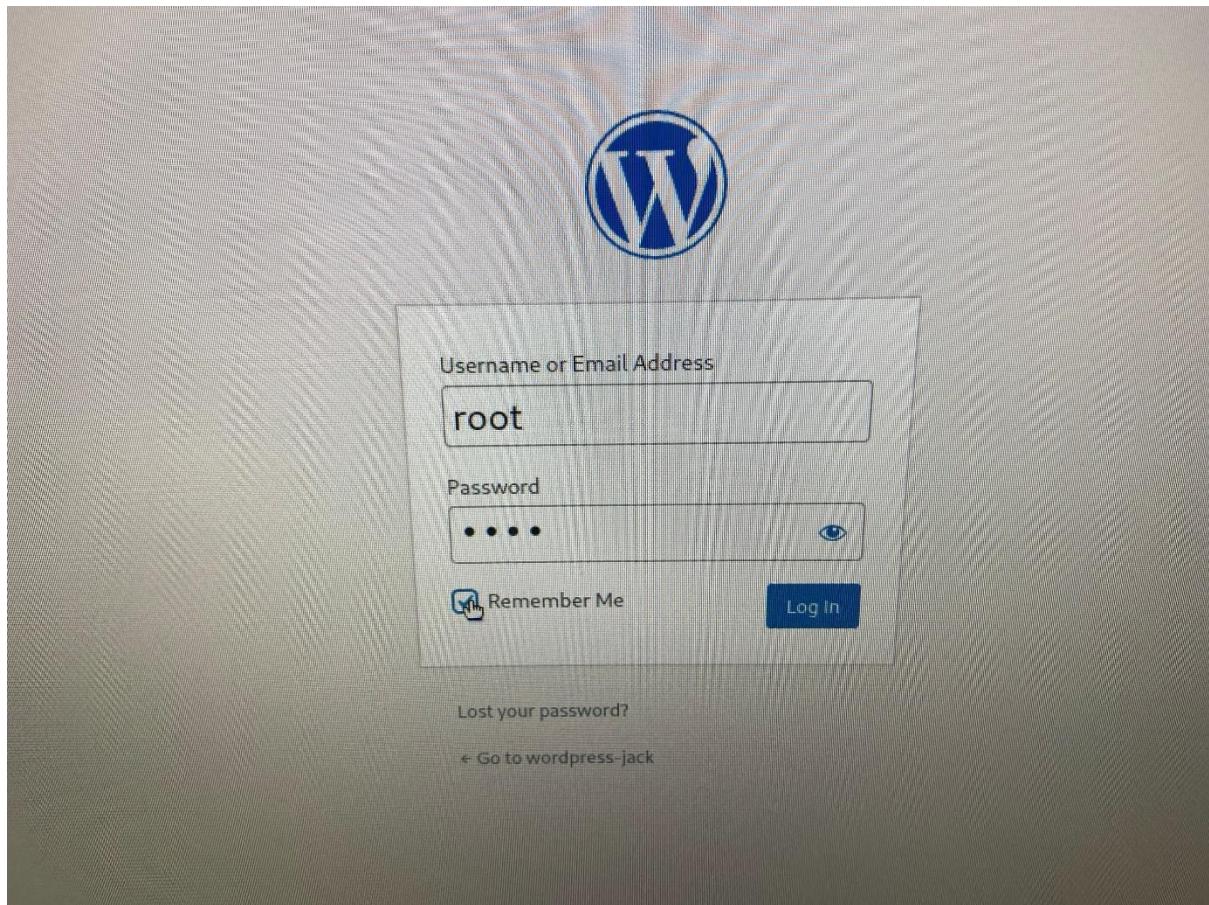
Votre e-mail

Vérifiez bien cette adresse e-mail avant de continuer.

Visibilité par les moteurs de recherche ☐ Demander aux moteurs de recherche de ne pas indexer ce site

Certains moteurs de recherche peuvent décider de l'indexer malgré tout.

Ensuite dans le login et mot de passe j'utiliser jack.williams@les-charmillles.fr et azerty en tant que mot de passe.



4. Apache

1) Historique

Internet : Interconnexion Mondiale des réseaux, créé dans les années 60 au ministères de la defence americain

World Wide Web (www) : systeme hypertexte constitué par l'ensemble des pages reliées par le protocole http, créé en 1989 au CERN a Geneve.

Mosaic : premier navigateur web graphique en 1993, renommé

Netscape Navigateur en 1994, puis **Mozilla Firefox** en 2004. De son coté,

Microsoft développe son **Internet Explorer** a partir de 1995.

2) Le protocol http

Deuxième et dernière version en 2015.

Déroulement d'une connexion :

- Connexion du client au serveur,
- Envoi par le client d'une requête GET,
- Réponse du serveur,
- Envoi par le client d'une requête de fermeture,
- Reponse du serveur,
- Fermeture de la connexion.

Requêtes courantes :

- GET : Récupération d'un document
- HEAD : récupération des en-têtes seulement
- POST : envoi de données au serveur
- PUT : envoi d'un fichier
- DELETE : suppression d'un fichier
- CONNECT : accès serveur sécurisé HTTPS

En-têtes client (requête) :

- Host : nom du site recherché
- Référer : nom du site qui nous a renvoyé
- User-agent : votre navigateur
- Accept : format de fichier accepté
- Accept-language : langue acceptée

En-têtes serveur (réponse) :

- Date : du serveur
- Server : logiciel du serveur (Apache, IIS, Nginx)
- Content-type : format du fichier envoyé
- Content-length : taille du fichier

Codes réponses du serveur en 5 catégories :

- 100 a 199 : message informatif
- 200 a 299 : succès de la requête
- 300 a 399 : redirection
- 400 a 499 : erreur
- 500 a 599 : erreur interne serveur

Test avec Wireshark

Not Found

The requested URL was not found on this server.

Apache/2.4.61 (Debian) Server at 192.168.0.201 Port 80

144	15.881062	34.107.221.82	192.168.0.248	HTTP	270	HTTP/1.1 200 OK (text/plain)
166	16.184407	192.168.0.201	192.168.0.248	HTTP	578	HTTP/1.1 200 OK (text/html)
228	21.166967	192.168.0.248	192.168.0.201	HTTP	435	GET /favicon.ico HTTP/1.1
231	21.173523	192.168.0.201	192.168.0.248	HTTP	546	HTTP/1.1 404 Not Found (text/h
2322	196.809711	192.168.0.248	192.168.0.201	HTTP	472	GET /wordpress/rshfhjf HTTP/1.1
2324	196.839866	192.168.0.201	192.168.0.248	HTTP	546	HTTP/1.1 404 Not Found (text/h

Créer un fichier `motdepasse.htm` dans `/var/www/html`

```
<html>
```

```
<head>
```

```
<title> test mot de passe</title>
```

```
</head>
```

```
<body>
```

```
<h1>TEST</h1>
```

```
<form action = "motdepasse.htm" method="post">
```

```
<input type = "password" name="password"/>
```



```

        <button type = "submit">Envoyer</button>

    </form>

</body>

</html>

```

J'ai créé le fichier motdepasse.htm dans /var/www/html

```

jack@raspberrypi:/var/www/html $ sudo touch motdepasse.htm
jack@raspberrypi:/var/www/html $ sudo nano motdepasse.htm

```

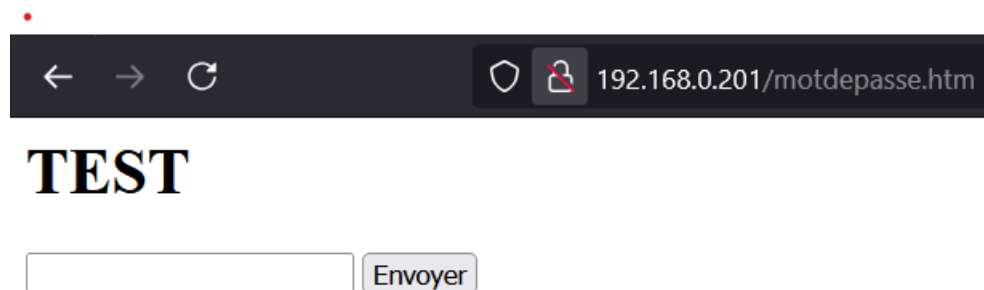
Ensuite j'ai ajouter le code html dans motdepasse.htm

```

GNU nano 7.2                                motdepasse.htm *
<html>
Document was last saved: Just now
    <title> test mot de passe</title>
  </head>
  <body>
    <h1>TEST</h1>
    <form action = "motdepasse.htm" method="post">
      <input type = "password" name="password"/>
      <button type = "submit">Envoyer</button>
    </form>
  </body>
</html>

```

Ensuite je suis aller sur le site web



Et j'ai lancer une trame Wireshark en même temps que j'ai remplis le mot de passe.



La fondation Apache :

- ONG créée en 1995 pour développer des logiciels libres.
- Des centaines de bénévoles sur une centaines de projets
- Licence Apache : a la différence de la GNU-GPL, on n'est pas obligé de republier sous licence apache

Le projet http Apache :

- Version 2.4 en 2012
- Conception modulaire : mod-php, mod-ssl
→ [Phpinfo.php](#)

3) Configuration de base

- Démarrage
Démon apache2 ou httpd
Nom de la machine : localhost
Commande hostname
/etc/hosts
Déclarer un nom

```
jack@raspberrypi:/etc $ sudo hostname williams
jack@raspberrypi:/etc $ hostname
williams
jack@raspberrypi:/etc $
```

```
GNU nano 7.2 hosts *
127.0.0.1 williams
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

127.0.1.1 raspberrypi
```

- Fichier de configuration : /etc/apache2/apache2.conf

Questions :

- Quel est le port par défaut ? Quelle directive ?
On utilise la directive listen pour écouter sur le port 80

```
Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>
```


- Y a-t-il d'autres fichiers de configuration ? Ou ?

Oui dans le /etc/apache/conf-enabled

```
jack@raspberrypi:/etc/apache2 $ cd conf-enabled/
jack@raspberrypi:/etc/apache2/conf-enabled $ ls
charset.conf          other-vhosts-access-log.conf  serve-cgi-bin.conf
localized-error-pages.conf  security.conf
```

- Directive pour charger un module ?

Charger les modules dans /mods-enabled

```
jack@raspberrypi:/etc/apache2 $ cd mods-enabled/
jack@raspberrypi:/etc/apache2/mods-enabled $ ls
access_compat.load  authz_host.load  dir.load          negotiation.conf  setenvif.load
alias.conf          authz_user.load  env.load          negotiation.load  status.conf
alias.load          autoindex.conf  filter.load       php8.2.conf       status.load
auth_basic.load     autoindex.load  mime.conf         php8.2.load
authn_core.load     deflate.conf     mime.load         reqtimeout.conf
authn_file.load     deflate.load     mpm_prefork.conf reqtimeout.load
authz_core.load     dir.conf        mpm_prefork.load setenvif.conf
```

- Sous quelle identité (User/groupe) fonctionne le serveur ?

Root apache

- Quel est le répertoire racine ?

La racine /var/www/html

- Quelle page est renvoyée si l'adresse est un dossier ?

Index.html dans /var/www/html

- Comment sont traitées les adresses

<http://serveur/cgi-bin/toto>

Fichier de commande Shell

- Où sont les fichiers de logs

```
jack@raspberrypi:/var $ cd log/apache2/
jack@raspberrypi:/var/log/apache2 $ ls
access.log  error.log  other_vhosts_access.log
jack@raspberrypi:/var/log/apache2 $
```

Accéder d'une autre machine et vérifier access.log

On voit les connexions de l'autre PC dans /var/log/apache2/access.log

```
192.168.0.221 - - [02/Sep/2024:14:06:03 +0200] "GET /icons/openlogo-75.png HTTP/1.1" 200 6040
"http://192.168.0.201/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML
, like Gecko) Chrome/125.0.0.0 Safari/537.36"
192.168.0.221 - - [02/Sep/2024:14:06:03 +0200] "GET /favicon.ico HTTP/1.1" 404 491 "http://19
2.168.0.201/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Geck
o) Chrome/125.0.0.0 Safari/537.36"
jack@raspberrypi:/var/log/apache2 $
```

- Accès aux dossiers

Créer un dossier webftp dans /var/www/html

```
jack@raspberrypi:/var/www/html $ sudo mkdir webftp
jack@raspberrypi:/var/www/html $ ls
index.html motdepasse.htm phpinfo.php webftp wordpress
jack@raspberrypi:/var/www/html $
```

Copier dedans le fichier motdepasse.htm et accéder



```
jack@raspberrypi:/var/www/html $ sudo cp -r motdepasse.htm webftp/
jack@raspberrypi:/var/www/html $
```

Supprimer motdepasse.htm de l'URL. **Que se passe-t-il ?**

Il nous montre tout les sous dossier dans webftp.



Index of /webftp

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 motdepasse.htm	2024-09-02 14:17	269	

Apache/2.4.62 (Debian) Server at 192.168.0.201 Port 80

Renommer motdepasse.htm en index.htm et actualiser.

Pourquoi ?

```
jack@raspberrypi:/var/www/html/webftp $ sudo mv motdepasse.htm index.htm
jack@raspberrypi:/var/www/html/webftp $ ls
index.htm
```


Dans le fichier apache2.conf :

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

```
Options -Indexes
```

```
</Directory>
```

Tester avec <http://192.168.0.201/webftp>

Forbidden



You don't have permission to access this resource.

Apache/2.4.62 (Debian) Server at 192.168.0.201 Port 80

```
# See http://httpd.apache.org/docs/2.4/ to
# the directives and /usr/share/doc/apache2
# hints.
<Directory /var/www/html/webftp>
Options -Indexes
</Directory>
```

Avec le +

Index of /webftp

Name	Last modified	Size	Description
 Parent Directory	-		
 test.htm	2024-09-02 14:17	269	

Apache/2.4.62 (Debian) Server at 192.168.0.201 Port 80

```
# See http://httpd.apache.org/docs/2.4/ to
# the directives and /usr/share/doc/apache2
# hints.
<Directory /var/www/html/webftp>
Options +Indexes
</Directory>
# Summary of how the Apache 2 configuration
# The Apache 2 web server configuration
# upstream's suggested way to configure the
# default Apache2 installation a
```

4) Script CGI simple

- Dans votre dossier personnel, créer un fichier date.cgi :

```
#!/bin/bash
Echo "content-type : text/html"
Echo ""
Date +"%A %d %B %Y %T"
```

Donner les droits d'exécution et tester

J'ai créer le fichier et écrit le script avec nano

```
jack@raspberrypi:~ $ sudo touch date.cgi
jack@raspberrypi:~ $ sudo nano date.cgi
```

```
#!/bin/bash
echo "Content-type:text/html"
echo ""
date +"%A %d %B %Y %T"
```

j'ai donner les droit d'exécution sur le fichier date.cgi

```
jack@raspberrypi:~ $ sudo chmod +x date.cgi
jack@raspberrypi:~ $
```

On lance le fichier

```
jack@raspberrypi:~ $ ./date.cgi
content-type : text/html

lundi 02 septembre 2024 15:22:53
jack@raspberrypi:~ $
```

- Déplacer le script dans /usr/lib/cgi-bin

```
jack@raspberrypi:~ $ sudo mv date.cgi /usr/lib/cgi-bin/
jack@raspberrypi:~ $
```

Modifier le fichier conf :

```
<Directory "/usr/lib/cgi-bin">
```

```
    AllowOverride None
```

```
    Options +ExecCGI
```

```
    AddHandler cgi-script .cgi
```

```
    Require all granted
```

```
</Directory>
```

Déplacer le script dans /var/www/html/cgi-bin

```
jack@raspberrypi:/usr/lib/cgi-bin $ sudo mv date.cgi /var/www/html/cgi-bin/
```

Modifier le fichier conf

```
<Directory "/var/www/html/cgi-bin">
    allowOverride None
    options +ExecCGI
    addHandler cgi-script .cgi
    require all granted
</Directory>
```

Deuxième exemple : listproc.cgi

```
# !/bin/bash
```

```
echo "Content-type : text/plain"
```

```
echo
```

```
user=$(echo "$QUERY_STRING" | grep -oP '(?<=user=)\w+')
```

```
ps -fu $user
```

5) Protection des accès

- Créer un dossier [secret](#)

Configurer Apache pour que ce dossier soit accessible uniquement à partir du serveur, et pas d'un client. Je vous baise

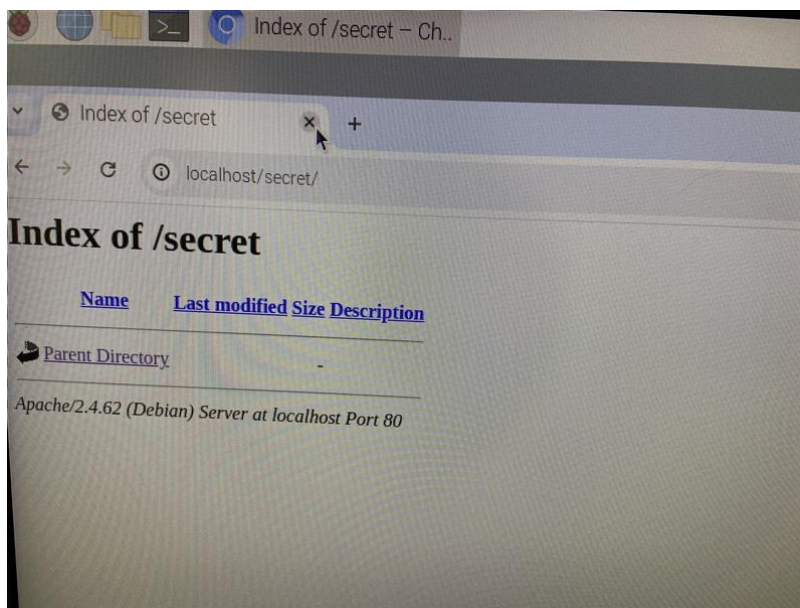
```
jack@raspberrypi:/var/www/html $ sudo mkdir secret
```

```
<Directory /var/www/html/secret>
    Deny from all
    Allow from localhost
</Directory>
```


Sur le PC client



Sur le serveur Apache :



Dans le fichier access.log on voit les utilisateurs qui ont essayer de se connecter

```
192.168.0.248 - - [03/Sep/2024:10:05:16 +0200] "GET /cgi-bin/date.cgi HTTP/1.1" 404 492 "-" >
192.168.0.248 - - [03/Sep/2024:10:05:16 +0200] "GET /favicon.ico HTTP/1.1" 404 491 "http://1>
192.168.0.248 - - [03/Sep/2024:10:27:19 +0200] "GET /listproc.cgi HTTP/1.1" 404 492 "-" "Mo>
192.168.0.248 - - [03/Sep/2024:10:27:19 +0200] "GET /favicon.ico HTTP/1.1" 404 491 "http://1>
192.168.0.248 - - [03/Sep/2024:10:44:48 +0200] "GET /dates.cgi HTTP/1.1" 404 492 "-" "Mozill>
192.168.0.248 - - [03/Sep/2024:11:04:33 +0200] "GET /secret HTTP/1.1" 403 495 "-" "Mozilla/5>
192.168.0.248 - - [03/Sep/2024:11:04:34 +0200] "GET /favicon.ico HTTP/1.1" 404 491 "http://1>
localhost - - [03/Sep/2024:11:06:32 +0200] "GET /secret HTTP/1.1" 301 568 "-" "Mozilla/5.0 (>
localhost - - [03/Sep/2024:11:06:32 +0200] "GET /secret/ HTTP/1.1" 200 649 "-" "Mozilla/5.0 >
localhost - - [03/Sep/2024:11:06:32 +0200] "GET /icons/blank.gif HTTP/1.1" 200 431 "http://l>
localhost - - [03/Sep/2024:11:06:32 +0200] "GET /icons/back.gif HTTP/1.1" 200 499 "http://lo>
localhost - - [03/Sep/2024:11:06:32 +0200] "GET /favicon.ico HTTP/1.1" 404 487 "http://local>
```

- On veut protéger l'accès à un dossier par mot de passe :
Créer un dossier `prive`

```
jack@raspberrypi:/var/www/html $ sudo mkdir prive
```

Vérifier qu'il y a une directive `AccessFileName`

```
# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives. See also the AllowOverride
# directive.
#
AccessFileName .htaccess
```

```
jack@raspberrypi:/var/www/html/prive $ sudo touch .htaccess
jack@raspberrypi:/var/www/html/prive $ ls -a
.  ..  .htaccess
jack@raspberrypi:/var/www/html/prive $ ls
jack@raspberrypi:/var/www/html/prive $
```

Dans le dossier `prive`, créer un fichier `.htaccess`

Exemple de fichier `.htaccess` :

```
AuthUserfile /etc/apache2/users
AuthName "Acces privé"
AuthType Basic
Require valid-user
```

```
GNU nano 7.2                               .htaccess *
AuthUserfile /etc/apache2/users
AuthName "Acces privé"
AuthType Basic
require valid-user
```

Creation d'un utilisateur Apache:

```
Cd /etc/apache2
Htpasswd -c users toto
Verifier le fichier users
Tester l'accès au dossier
```

```
jack@raspberrypi:/etc/apache2 $ sudo htpasswd -c users toto
New password:
Re-type new password:
Adding password for user toto
jack@raspberrypi:/etc/apache2 $
```

Dans le fichier `users` on voit l'utilisateur « toto »

```
jack@raspberrypi:/etc/apache2 $ cat users
toto:$apr1$j9sX5USe$t7g0X6N25pkn4uXLULZFj1
```

Ensuite dans apache2.conf on ajoute cette ligne

```
<Directory /var/www/html/prive>
    Options Indexes FollowSymLinks
    AllowOverride AuthConfig
    Require all granted
</Directory>
```

Maintenant lorsqu'on lance <http://192.168.0.201/prive> et on se connecte avec le nom d'utilisateur toto et mot de passe qwerty

Lorsqu'on lance une trame wireshark http on intercept le nom d'utilisateur et mot de passe

```
▼ Hypertext Transfer Protocol
  ▶ GET /prive/ HTTP/1.1\r\n
    Host: 192.168.0.201\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:129.0) Gecko/20100101 Firefox/129.0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8\r\n
    Accept-Language: en-GB,en;q=0.5\r\n
    Accept-Encoding: gzip, deflate\r\n
    DNT: 1\r\n
    Sec-GPC: 1\r\n
    ▼ Authorization: Basic dG90b2pxd2VydHk=\r\n
      Credentials: toto:qwerty
```

6) Hôtes virtuels

Une machine peut héberger plusieurs sites différents dans des dossiers différents.

NameVirtualHost @IP_serveur

<VirtualHost @IP_serveur>

DocumentRoot /var/www/html/site1

ServerName site1.fr

</VirtualHost>

J'ai créer les fichier site1 et site2

```
jack@raspberrypi:/var/www/html $ mkdir site1 site2
```


Dans site1 et site2 j'ai créé un fichier « index.htm »

```
jack@raspberrypi:/var/www/html/site1 $ sudo touch index.htm
```

```
jack@raspberrypi:/var/www/html/site2 $ sudo touch index.htm
```

Dans les fichiers index.htm

```
GNU nano 7.2 index.htm *
<html>
<body>
<h1>Coucou site1</h1>
</body>
</html>
```

```
GNU nano 7.2 index.htm *
<html>
<body>
<h1>Coucou site2</h1>
</body>
</html>
```

Ensuite dans apache2.conf on ajoute cette configuration :

```
NameVirtualHost @192.168.0.201
<VirtualHost @192.168.0.201>
    DocumentRoot /var/www/html/site1
    ServerName site1.fr
</VirtualHost>

NameVirtualHost @192.168.0.201
<VirtualHost @192.168.0.201>
    DocumentRoot /var/www/html/site2
    ServerName site2.fr
</VirtualHost>

<Directory /var/www/html/site1>
    Options Indexes FollowSymLinks
    Order Deny, Allow
    Deny from all
    Allow from localhost 192.168.0.201/255.255.255.0
</Directory>

<Directory /var/www/html/site2>
    Options Indexes FollowSymLinks
    Order Deny, Allow
    Deny from all
    Allow from localhost 192.168.0.201/255.255.255.0
</Directory>
```

7) HTTPS: le HTTP sécurisé

Par défaut, Apache utilise le protocole http sur le port 80. Or nous avons vue que les mots de passe circulaient en clair ! Donc nous voulons passer le serveur en HTTPS sur le port par défaut 443. Apache contient deux sites préconfigurés qui pointent tous les deux vers le dossier racine /var/www/html. Le premier, « default », est actif par défaut et permet d'accéder a la page « It Works » Le second, « default-ssl », est désactiver par défaut.

Commandes d'activation :

`A2enmode ssl`

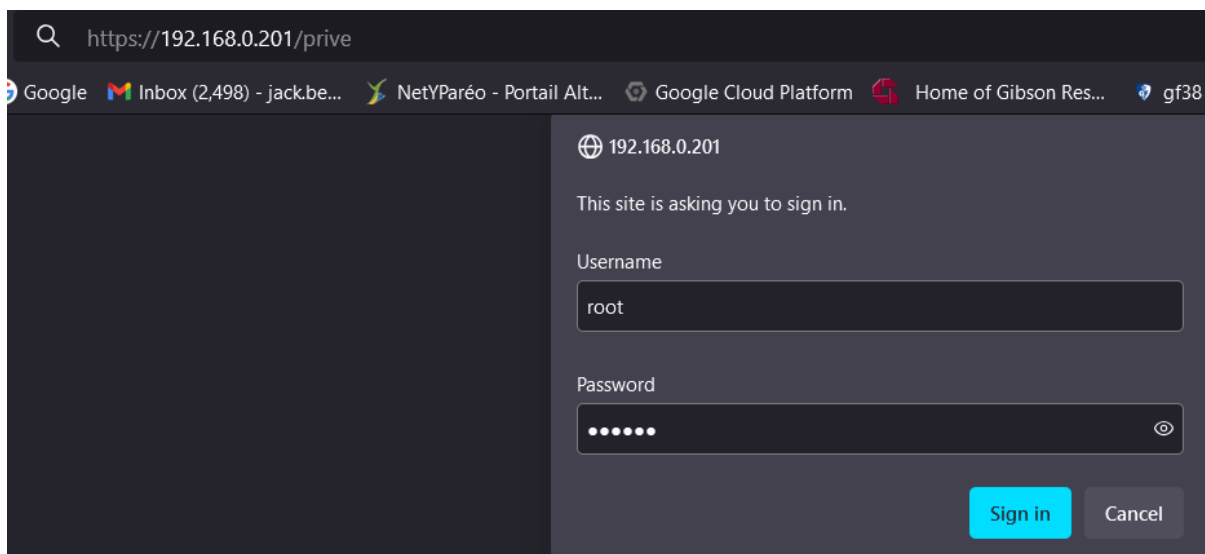
`A2ensite default-ssl`

`Service apache2 restart`

J'ai fait les commandes ci-dessus

Ensuite j'me suis connecter sur le lien web <https://192.168.0.201/prive> et j'ai renseigner mon mot de passe et nom d'utilisateur, lorsque j'ai fait cela j'ai lancer une capture de Wireshark sur le réseau local avec le protocol TLS et j'ai capturer le trame « Application Data » après les échanges « Hello » entre le client et le serveur. Je retrouve le mot de passe et nom d'utilisateur haché.

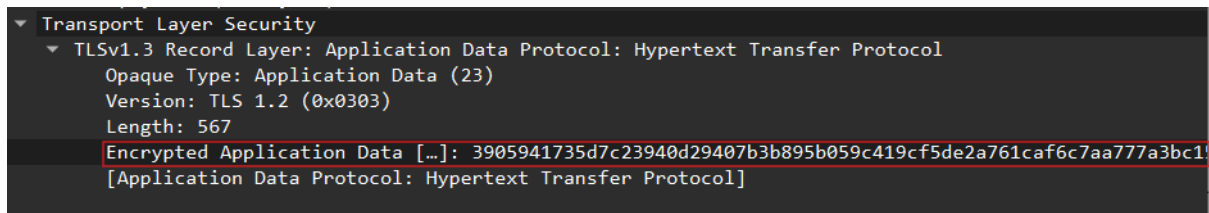
Recherche HTTPS :



Coter Wireshark :

No.	Time	Source	Destination	Protocol	Length	Info
2	0.669410	3.125.208.255	192.168.0.248	TLSv1.2	95	Application Data
6	1.167664	3.125.208.255	192.168.0.248	TLSv1.2	93	Application Data
11	2.166606	3.125.208.255	192.168.0.248	TLSv1.2	98	Application Data
17	2.287559	192.168.0.248	162.159.130.234	TLSv1.2	105	Application Data
19	2.396765	162.159.130.234	192.168.0.248	TLSv1.2	88	Application Data
44	4.172564	3.125.208.255	192.168.0.248	TLSv1.2	93	Application Data
60	6.228929	192.168.0.248	192.168.0.201	TLSv1.2	626	Application Data
61	6.229291	192.168.0.248	192.168.0.201	TLSv1.2	78	Application Data
72	6.362999	192.168.0.248	192.168.0.201	TLSv1.3	693	Client Hello
74	6.516988	192.168.0.201	192.168.0.248	TLSv1.3	1412	Server Hello, Change Cipher Spec, App
75	6.526596	192.168.0.248	192.168.0.201	TLSv1.3	118	Change Cipher Spec, Application Data
76	6.526801	192.168.0.248	192.168.0.201	TLSv1.3	626	Application Data
77	6.534272	192.168.0.201	192.168.0.248	TLSv1.3	133	Application Data
78	6.534272	192.168.0.201	192.168.0.248	TLSv1.3	133	Application Data
80	6.564288	192.168.0.201	192.168.0.248	TLSv1.3	803	Application Data

Mot de passe :



Certificat par défaut dans /etc/ssl/private

```

root@raspberrypi:/etc/ssl/private# cat ssl-cert-snakeoil.key
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFAASCbKYYwggsiAgEAAoIBAQDJDruBf0/sn5jd
0ZCBmNBhXm/+50yi9YeqGtULS653dY2cFB+8zqzzPvroPMqmhHsEibAGUaOgylD5
HEBZQ3HH6vshDy4xqm3hh8A1YmjPzajJkms6Cicz9kwaBu7/7FIYg8Z0Z66ux+Xm
anxVv5am6QtjFa6lKq1koB4Ph420h1qBujr5YQQRNPGAZ0bXOL36AFbzdGrTxY0r
udoZ64SHV8iAvbL3jMhgU1UATu08KLt3z0TXZ7zHK6+tC8yEvxjhK+B3GMCQ6A4h
pcPlSLJkwtJ9PD51R7Y3tiB5SuupstP2pTZcJUrjm3EDRHGD9WvLOSFpoPmn6LMB
ziLaPlyZAgMBAAECggEACrHLlFTfFWuOPSxCgYI9Y1CAqJBIVgpNYxD2UBmAeD46
gBjZ71MXBAFPM1bQeLhOGtmwEUX6bC0WzYlHkt5ZiQbIwONr05x7J8nuwWgwpw0L
c1lkcrvjhernJJoFoD6TGQSORQs5LDkr3wZ/caTs9qjcvMeAPYB27B2U09C2ALDtxcH
5rwdmmOVgjSjQe2Y9G3ghaFcvffm+boU3hc170sGvgJasXUA0yz+1LgSY2AKNrUp
NKsIDk21lKikWfQHndnr+edMjJS35LMKVzROFKDc0zMW7E9TimpPoi5pXJD1VR74J
npef50SxyEB3ds0+S8ekZZ3BWCT/S1apM9WkRT/dAQKBgQDors7IynrLJ9kVWkog
jNXN9gLi1THR0eoEjxm2Y/n/uU2Cn6GjyRv+UBsQcDlntUsDhaoR2jw6o0zCZORk
DLI7Q/2tTAkoWzvtz/eIFe7/xJo5weLX77F860Gr5ucm6nwPvm+Y4YRTdKnM7wg5
VA2CjeIBQbAABn4EVzr6NKpccQKBgQDdNJBsfrYOL9hm1GQ1tdYpxeDcvJ66X+zu
SD7oV272fJvZy2UUA+d2XdlsZ+008eKz3Q00p0Vd2eRhkiah6JV4bQ8tJ7yTbuuG
2s1GvvpqzNB2s4Y+YD9//WWohblitPufmhUeqdf0iSR63ISXuRcLeGAPr9DVhL1
kdaQ3H+2qQKBgDCite0//tj9WNSRXsDnX3l0hS1x/BKN9gg5aR2YQP9MQZSrg2Xl
fyB0yT2mkjLFp/32FYFfygxyoMnzjYK8nzipd/J717UpQpjaNkGWgaGr+pdtkSSc
hKMPWHMo2LJPWs1V5aU39QgLX5e1jDuZArBms+ubLgDzWbnCEDU8YwShAoGAQ5tV
FVSA0+kvnI4BaQD+qZm+1dj0Jyw2P8V45I4EbU7tN8DL5DZ50VbtFQ3EcA/53UC
Gjudace0k1WV1Ro+xKWgY86nRfA6r+ajjoqGv64BZnk3UE7Ba+AU8Urs8G8qkt0zk
05BZXnwj3qETNTY6otjRmwtpc00XsLi2X+PkbLEcGYB7JRDMUXcfahZ/nIzoaf1t
uyWryt23G9EGYkTrkcSYnG+RKXHS715m0H73TpjmbLqDJxpNpnYJ+y8wxzlitv4U
RhTWpFB/rcp5P/dKHDeLsL7K2IvAlvuV/xumfLu01jxXQVD3WQA0i2U4hXQEDE4V
//hpcA8JKZeytdhv+Qc7SA==
-----END PRIVATE KEY-----

```

Création d'un nouveau certificat possible avec la commande `openssl` et des options. Exemple :

```
OpenSSL req -x509 -nodes -days 365 -newkey rsa:2048 -sha256 -out /etc/apache2/server.crt -keyout /etc/apache2/server.key
```

[illegible]

On regarder dans les fichiers `/etc/apache2/server.crt` et `/etc/apache2/server.key`


```

root@raspberrypi:~# cd /etc/apache2/
root@raspberrypi:/etc/apache2# cat server.crt
-----BEGIN CERTIFICATE-----
MIIEFzCCAv+gAwIBAgIUT/u6PtM3WqL0vPv2zQvxvEuILbQwDQYJKoZIhvcNAQEL
BQAwgZoxCzAJBgNVBAYTAkZSMQ4wDAYDVQQIDAVJc2VyZTERMA8GA1UEBwwIR3Jl
bm9ibGUxEzARBgNVBAoMCKNoYXJtaWxsZXMsZXMxEDAOBgNVBAsMB0JUUYBTSU8xETAP
BgNVBAMMCFdJTEJxJQU1TMS4wLAYJKoZIhvcNAQkBFh9qYWNrLndpbGxpYW1zQGxlc
y1jaGFyYbWlsbGVzLmZyMB4XDTE0MDkyNDA4NTE0M1oXDTI1MDkyNDA4NTE0M1ow
gZoxCzAJBgNVBAYTAkZSMQ4wDAYDVQQIDAVJc2VyZTERMA8GA1UEBwwIR3Jlbm9i
bGUxEzARBgNVBAoMCKNoYXJtaWxsZXMsZXMxEDAOBgNVBAsMB0JUUYBTSU8xETAPBgNV
BAMMCFdJTEJxJQU1TMS4wLAYJKoZIhvcNAQkBFh9qYWNrLndpbGxpYW1zQGxlc
y1jaGFyYbWlsbGVzLmZyMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAI
DzjD6Z6EHGtLCs/xW6yX0ZaoUy5kiGrc9EjZB9SJ8pPmcri0Uj/lJEQ5YQ5mbxz1q5D
w7/jCfG+Vt5CzwtKkt/EtGztPj7FyaMvL2F/LC7mqNz8npPkr32ocbUUKm3Ith4P
+mQfZLLD7csGrfLxcF0y/lfZi8fjghCZPx5oYe1ntsWcAjHLuF1j11M3s4Ks6M4v
9sFYLBFv1MbJCR2g977eNvQ9IZZWsgDffwqFgQKGqVavGNJTA4tmaUGrECx9TDL3

root@raspberrypi:/etc/apache2# cat /etc/apache2/server.key
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQDUPOMPpnoQca0s
Kz/FbrJc5lqhTlMsiatz0SNKH1Inyk+ZyuLRSP+UkRDlhDmZvHPWrkPDv+MJ+D5W
3kLPC2Qq38S0b00+PsXJoy8vYX8sLuao3Pyek+RHfahxtRSSbci2Hg/6ZB9mUsPt
ywat8vFwXTL+V9mLx+OCEJk/Hmhh7We2xZwCMcu4XWPXUzezgqzoz/2wVgsEW/U
xskJHaD3vt429D0hl1layAN9/CoWBAopVq8Y0LMDi2ZpQasQLH1MOXcmFZtW3br2
X8DK4eYKXm5ftpw5qStX/qTwjZhBTcAc19iiAMsijSKu9uy6WwWZz0Fw8t0PLeIb
fVVY+EgvAgMBAAECggEABGS7m+f/ADbPi5SG821Wio8c3S3wwXz/nEbTYWU3QX+c
pOW1oE4kjEzHJBktSr4Yn8m9DrZQMDVQBQ2rewxp/GvkGeAPYTR8ylnIcPCnE0u9
QU2hz8k79obmum6HMyjL+N59HNp/E+fZX84jde0Li8nsyPsZ6Rza3CcVJQtaGdIo
poNjbyGjagvVoHUfEmEMWtoSztQXNURSbNcz5CNCldUId+qSFcdXthLC/ypv045
V469se1c6C8U8C3pos8Ylt11HsKewc0a+1cmGBSJxfGdLQ3NfE5ZjBLkrfBlEiAj

```

Deux manieres pour forcer le HTTPS :

- Désactiver le http : [a2disite default](#)
- Rediriger le HTTPS : on remplace le le fichier /etc/apache2/sites-available/default.conf
[Redirect permanent / https://localhost](#) et redémarrer

5. Utilisation de MySQL (MariaDB)

1) Introduction

MySQL est un SGBD (Système de gestion de bases de données) libre, complémentaire de PHP, pour créer des sites web dynamique.

Autres SGBD libre : Postgresql

Autres SGBD propriétaires : Oracle, Microsoft SQL Server

2) Installation

Déjà faite, vérification avec <http://localhost/phpinfo.php>

3) Fonctionnement

Architecture client/serveur

Serveur `mysqld`

4) Commande mysql

Entrée dans l'éditeur : `mysql -u root -p`

Sortie : `exit`

Création d'une base Films :

```
CREATE DATABASE Films ;  
GRANT ALL PRIVILEGES ON Films.* TO adminFilm@localhost  
IDENTIFIED BY 'admin';
```

Sortie: `exit`

Reconnexion: `mysql -u adminFilms -p Films`

Création d'une table :

```
CREATE TABLE FilmSimple (  
    Titre VARCHAR(30)  
    Année INTEGER  
);
```

Remplissage

```
INSERT INTO FilmSimple (titre,annee) VALUES  
    ('Pulp Fiction', 1994),  
    ('Alien', 1979),  
    ('Titanic', 1997);
```

Interrogation : exemple

```
SELECT titre FROM FilmSimple WHERE annee=1997;
```

```

MariaDB [Films]> DESC FilmSimple;
+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| titre | varchar(30)   | YES  |     | NULL    |       |
| annee | int(11)       | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+
2 rows in set (0,003 sec)

MariaDB [Films]> SELECT * FROM FilmSimple;
Empty set (0,001 sec)

MariaDB [Films]> INSERT INTO FilmSimple (titre,annee) VALUES
-> ('Pulp Fiction',1994),
-> ('Alien',1979),
-> ('Titanic',1997);
Query OK, 3 rows affected (0,007 sec)
Records: 3 Duplicates: 0 Warnings: 0

MariaDB [Films]> SELECT * FROM FilmSimple;
+-----+-----+
| titre      | annee |
+-----+-----+
| Pulp Fiction | 1994  |
| Alien       | 1979  |
| Titanic     | 1997  |
+-----+-----+
3 rows in set (0,001 sec)

MariaDB [Films]>

```

Ajouter, supprimer, modifier des films

Ajouter un film dans la table :

```

MariaDB [Films]> INSERT INTO FilmSimple (titre,annee) VALUES
-> ('Batman',1989);
Query OK, 1 row affected (0,007 sec)

MariaDB [Films]> SELECT * FROM FilmSimple;
+-----+-----+
| titre      | annee |
+-----+-----+
| Pulp Fiction | 1994  |
| Alien       | 1979  |
| Titanic     | 1997  |
| Batman      | 1989  |
+-----+-----+
4 rows in set (0,001 sec)

MariaDB [Films]>

```

Supprimer le film « Batman »


```
MariaDB [Films]> DELETE FROM FilmSimple WHERE titre = 'Batman' AND annee = '1989';
Query OK, 1 row affected (0,006 sec)

MariaDB [Films]> SELECT * FROM FilmSimple;
+-----+-----+
| titre      | annee |
+-----+-----+
| Pulp Fiction | 1994 |
| Alien       | 1979 |
| Titanic     | 1997 |
+-----+-----+
3 rows in set (0,001 sec)

MariaDB [Films]> 
```

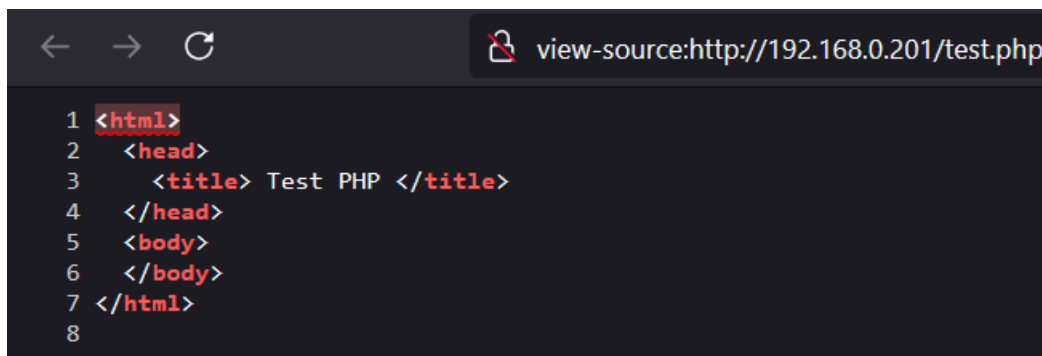
Tester des requêtes

1) Lien avec PHP

On crée un fichier test.php dans /var/www/html :

```
<?php
    echo "<html>\n";
    echo " <head>\n";
    echo "  <title> Test PHP </title>\n";
    echo " </head>\n";
    echo " <body>\n";
    echo " </body>\n";
    echo "</html>\n";
?>
```

Tester dans un navigateur : page blanche avec un titre ctrl + u pour vérifier la qualité du code



```
view-source:http://192.168.0.201/test.php

1 <html>
2   <head>
3     <title> Test PHP </title>
4   </head>
5   <body>
6   </body>
7 </html>
8
```

On ajoute ensuite dans le body ce code

```
test.php
<?php
    echo "<html>\n";
    echo "  <head>\n";
    echo "    <title> Test PHP </title>\n";
    echo "  </head>\n";
    echo "  <body>\n";
    $connexion = mysqli_connect("localhost", "adminFilms", "root", "Films");
    $requete = "select * from FilmSimple";
    $resultat = mysqli_query($connexion,$requete);
    while ($film =mysqli_fetch_assoc($resultat)) {
        echo "<P>".$film['titre']." paru en ".$film['annee']."</P>\n";
    }
    mysqli_close();
    echo "  </body>\n";
    echo "</html>\n";
?>
```

Ce qui nous ajoute cela sur la page web :



Pulp Fiction paru en 1994

Alien paru en 1979

Titanic paru en 1997

Batman paru en 1989

2) Administration MySQL

- Ajouter un nouvel utilisateur sur la base Films avec tous les droits, puis enlevez lui les droits d'écriture. Testez !

Supprimer le droit d'écriture :

```
MariaDB [(none)]> REVOKE INSERT, UPDATE, DELETE ON Films.* FROM 'toto'@'localhost';
Query OK, 0 rows affected (0,004 sec)
```

Créer un nouvel utilisateur qui a uniquement les droits de sélection.

Tester !

```
MariaDB [(none)]> CREATE USER 'test'@'localhost' IDENTIFIED BY 'test'
-> ;
Query OK, 0 rows affected (0,009 sec)

MariaDB [(none)]> GRANT SELECT ON Films.* TO 'test'@'localhost';
Query OK, 0 rows affected (0,005 sec)

MariaDB [(none)]> 
```

Connecter sur le compte test qu'on a créé (Test) :

```
jack@raspberrypi:~ $ mysql -u test -p Films
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor.  Commands end with ; or \g.
```

Tester avec un SELECT :

```
MariaDB [Films]> SELECT * FROM FilmSimple;
+-----+-----+
| titre      | annee |
+-----+-----+
| Pulp Fiction | 1994  |
| Alien       | 1979  |
| Titanic     | 1997  |
| Batman      | 1989  |
+-----+-----+
4 rows in set (0,009 sec)

MariaDB [Films]> 
```

- Quelles sont les options de la commande mysqladmin ?

La commande mysqladmin est utilisé pour exécuter les commandes MySQL dans le CLI Linux.

Exemple d'utilisation

```
jack@raspberrypi:~ $ mysqladmin -u root -p status
Enter password:
Uptime: 22325  Threads: 1  Questions: 13  Slow queries: 0  Opens: 21  Open tables: 14  Querie
s per second avg: 0.000
jack@raspberrypi:~ $ 
```

- Utiliser la commande mysqldump pour sauvegarder votre base dans un SQL.

Vérifier le contenu du fichier


```
jack@raspberrypi:~ $ mysqldump -u root -p Films > sauvegarde_Films.sql
Enter password:
jack@raspberrypi:~ $ cat sauvegarde_Films.sql
-- MariaDB dump 10.19  Distrib 10.11.6-MariaDB, for debian-linux-gnu (aarch64)
--
-- Host: localhost    Database: Films
--
-- Server version      10.11.6-MariaDB-0+deb12u1

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
```

Supprimer la base Films

```
MariaDB [(none)]> DROP DATABASE Films;
Query OK, 1 row affected (0,043 sec)

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| wordpress |
+-----+
5 rows in set (0,008 sec)

MariaDB [(none)]> 
```

Recréer la base Films en important le fichier SQL

```
jack@raspberrypi:~ $ mysql -u root -p -e "CREATE DATABASE Films;"
Enter password:

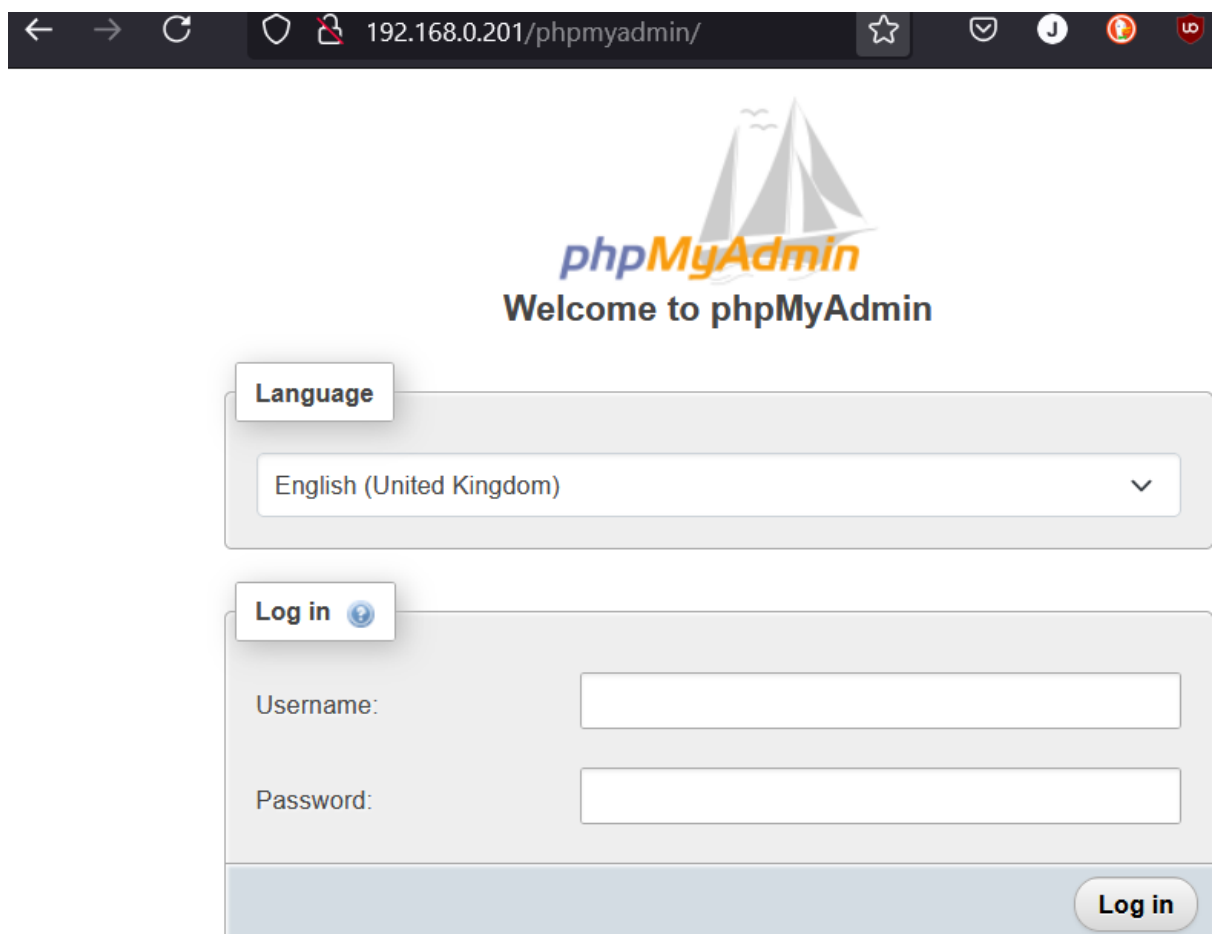
jack@raspberrypi:~ $ mysql -u root -p Films < sauvegarde_Films.sql
Enter password:
jack@raspberrypi:~ $ 
```

Ensuite on regarde si la base Films est de retour :


```
MariaDB [Films]> SELECT * FROM FilmSimple;
+-----+-----+
| titre      | annee |
+-----+-----+
| Pulp Fiction | 1994 |
| Alien       | 1979 |
| Titanic     | 1997 |
| Batman      | 1989 |
+-----+-----+
4 rows in set (0,001 sec)

MariaDB [Films]> 
```

Afficher phpmyadmin dans le navigateur



← → ↻ 🔒 192.168.0.201/phpmyadmin/ ☆ 📁 J 🔄 🔴


Welcome to phpMyAdmin

Language

English (United Kingdom) ▼

Log in ?

Username:

Password:

Log in

- Rajouter un film, le modifier, le supprimer
Je suis aller dans insérer et j'ai ajouter le Film « RAMBO »

titre	annee
Pulp Fiction	1994
Alien	1979
Titanic	1997
Batman	1989
RAMBO	1982

J'ai ajouté une clé primaire qui se nomment « id » pour qu'on puisse supprimer des films ensuite on clique sur « DELETE ».

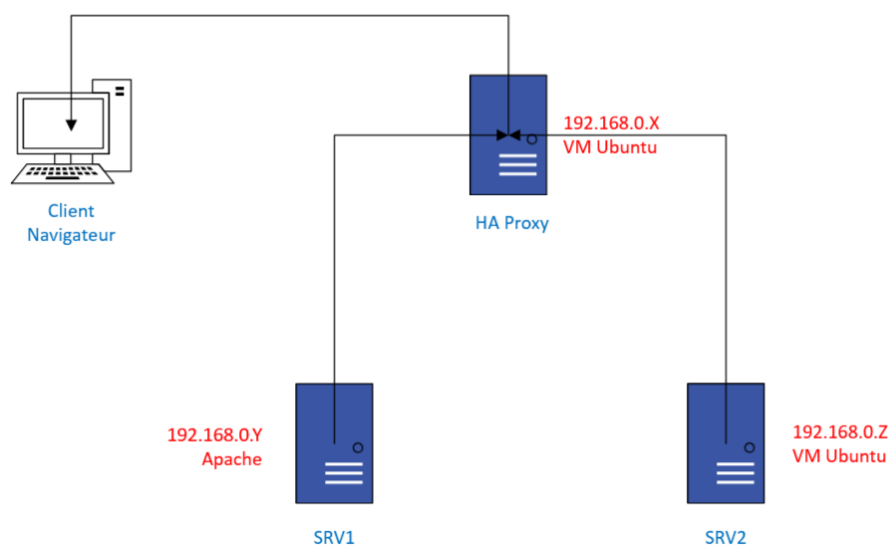
<input type="checkbox"/>	Edit	Copy	Delete	1	Pulp Fiction	1994
<input type="checkbox"/>	Edit	Copy	Delete	2	Alien	1979
<input type="checkbox"/>	Edit	Copy	Delete	3	Titanic	1997
<input type="checkbox"/>	Edit	Copy	Delete	4	Batman	1989
<input type="checkbox"/>	Edit	Copy	Delete	5	RAMBO	1982
<input type="checkbox"/>	Edit	Copy	Delete	6	RAMBO	1982

Ou utiliser « Edit » pour le modifier

- Rajouter un utilisateur avec droit de sélection
- Se connecter avec cet utilisateur et faire une sélection

3) Cluster Apache

HA proxy (High availability Proxy) est un logiciel de balance de charge open-source qui permet de répartir les requetes entre plusieurs serveurs pour assurer la haute disponibilité du service.



Configuration HAProxy :

Sudo apt install haproxy

```
srv1@srv1:~$ sudo apt install haproxy
```

Fichier /etc/haproxy/haproxy.conf → copie .old

```
srv1@srv1:/etc/haproxy$ ls
errors haproxy.cfg haproxy.cfg.old
srv1@srv1:/etc/haproxy$
```

frontend http_front

bind *:80

default_backend http_back

backend http_back

balance roundrobin

server SRV1 192.168.0.201:80 check

server SRV2 192.168.0.125:80 check

Lorsque je me connecte sur le serveur haproxy 192.168.0.149 il m'ouvre soit la fenêtre apache par défaut du Raspberry (192.168.0.201 Debian) ou le Ubuntu (192.168.0.125 Ubuntu)

