

TP SERVEURS



TP SERVEURS

- Dans un premier temps nous allons installer un ssh pour cela il nous suffit d'effectuer la commande suivante: « **apt install openssh-server** »
- Comme nous pouvons le voir le ssh est bien en place

```
192.168.1.28 - PuTTY
login as: root
root@192.168.1.28's password:
Linux romain 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-1
x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 10 19:05:57 2024
root@romain:~#
```



Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

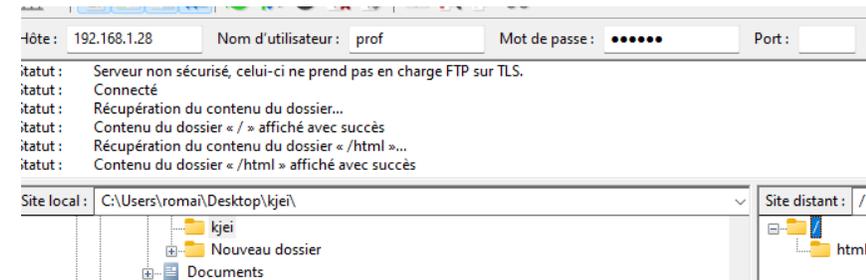
```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf
```

TP SERVEURS

- Ensuite pour le serveur lamp nous allons utiliser apache2, pour se faire nous allons utiliser la commande: ***apt install apache2 -y***. Une fois apache installer nous avons juste à taper l'adresse IP de la Debian et de la rechercher sur internet, ce qui devrait nous donner une page web.

TP SERVEURS

- Ensuite nous allons pouvoir ajouter un ftp dans debian pour se faire nous allons utiliser la commande « **apt get install proftpd** »
- Et nous allons faire en sorte que les utilisateurs créer soit directement diriger dans le dossier /var/www pour se faire nous devons modifier le fichier avec la commande cd /etc/proftpd pour y ajouter « defaultRoot /var/www »



```
# Use this to jail all users in their homes
DefaultRoot /var/www
```

TP SERVEURS

```
root@romain:/var# chown prof /var/www
root@romain:/var# ls -l
total 40
drwxrwxrwx  2 root root  4096  5 sept. 08:40 backups
drwxrwxrwx 22 root root  4096  5 sept. 08:46 cache
drwxrwxrwx 56 root root  4096  5 sept. 08:46 lib
drwxrwsrwx  2 root staff 4096 28 janv. 2024 local
lrwxrwxrwx  1 root root    9  2 sept. 09:30 lock -> /run/lock
drwxrwxrwx 13 root root  4096  5 sept. 08:46 log
drwxrwsrwx  2 root mail  4096  2 sept. 09:30 mail
drwxrwxrwx  2 root root  4096  2 sept. 09:30 opt
lrwxrwxrwx  1 root root    4  2 sept. 09:30 run -> /run
drwxrwxrwx  6 root root  4096  2 sept. 09:35 spool
drwxrwxrwx 11 root root  4096  5 sept. 09:09 tmp
drwxrwxrwx  3 prof root  4096  4 sept. 08:53 www
```

```
root@romain:/var# addgroup eleves
Ajout du groupe « eleves » (GID 1004)...
fait.
root@romain:/var# usermod -G eleves -a eleve
root@romain:/var#
```

```
root@romain:/var# chmod 750 /var/www
root@romain:/var# chgrp -R eleves /var/www
root@romain:/var# chown -R prof /var/www
root@romain:/var# chmod -R 750 /var/www
root@romain:/var# _
```

- Maintenant nous allons définir les droits entre les différents utilisateurs pour se faire nous allons créer un groupe élevé et y ajouter l'utilisateur élevé avec la commande `usermod -G eleves -a eleve`
- Nous allons définir l'utilisateur prof comme le propriétaire avec la commande `chown` ce qui lui permettra de modifier le dossier comme il le souhaite.

Hôte : 192.168.1.27 Nom d'utilisateur : prof Mot de passe : ●●●●●● Port : Connexion rapide ▾

Statut : Connexion établie, attente du message d'accueil...
Statut : Serveur non sécurisé, celui-ci ne prend pas en charge FTP sur TLS.
Statut : Connecté
Statut : Récupération du contenu du dossier...
Statut : Contenu du dossier « / » affiché avec succès
Statut : Création du dossier « /test »...

Hôte : 192.168.1.27 Nom d'utilisateur : eleve Mot de passe : ●●●●●● Port : Connexion rapide ▾

Erreur : Impossible d'établir une connexion au serveur
Statut : Création du dossier « /Nouveau dossier »...
Commande : CWD /
Réponse : 250 Commande CWD exécutée avec succès
Commande : MKD Nouveau dossier
Réponse : 550 Nouveau dossier: Permission non accordée

TP SERVEURS

- Maintenant on va essayer si les droits fonctionnent bien pour se faire nous allons utiliser l'outil Filezilla
- Et comme nous pouvons le voir, l'utilisateur prof peut effectuer des modification et l'utilisateur élevé ne peut que regarder,

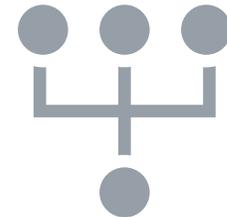
TP SERVEURS



Ensuite nous allons mettre en place phpMyAdmin pour se faire nous allons avoir besoins de `php` et de `mysql`



Pour php nous allons utiliser la commande `apt install php`, pour MySQL utiliser la commande `apt install mariadb-server -y`.



Et pour finir nous allons installer PhpMyAdmin avec la commande : `apt install phpmyadmin -y`

Configuration de phpmyadmin

Le paquet phpmyadmin a besoin d'une base de données installée et configurée avant de pouvoir être utilisé. Ceci peut si nécessaire être géré par dbconfig-common.

Si vous êtes un administrateur de bases de données expérimenté et savez que vous voulez procéder à cette configuration vous-même, ou si votre base de données est déjà installée et configurée, vous pouvez refuser cette option. Des précisions sur la procédure se trouvent dans /usr/share/doc/phpmyadmin.

Autrement, vous devriez choisir cette option.

Faut-il configurer la base de données de phpmyadmin avec dbconfig-common ?

<Oui>

<Non>

Configuration de phpmyadmin

Merci de choisir le serveur Web à reconfigurer automatiquement pour exécuter phpMyAdmin.

Serveur Web à reconfigurer automatiquement :

apache2
 lighttpd

<Ok>

TP SERVEURS

Ensuite nous allons choisir apache2 et autoriser la configuration de la base de données de phpMyAdmin

TP SERVEURS

Nous allons maintenant administrer les droits a un utilisateur pour qu'il accédé à MySQL et phpMyAdmin

Pour se faire nous allons utiliser la commande : GRANT ALL PRIVILEGES ON (etoile) (point) (etoile) TO 'votre utilisateur'@'localhost' IDENTIFIED BY 'votre mot de passe' WITH GRANT OPTION;

Nous allons mettre commande dans mysql pour se faire nous allons taper la commande: mysql

TP SERVEURS

Nous allons définir un mot de passe pour accéder à la page web afin de restreindre l'accès à certains utilisateurs pour ce faire nous allons devoir utiliser un dossier .htaccess

Pour ce faire nous allons créer le fichier avec la commande: `htpasswd -c /var/www/html/.htpasswd prof`. Il faut noter que l'option "-c" (pour créer) n'intervient que si le fichier n'existe pas déjà.

```
root@romain:/var/www/html# htpasswd -c /var/www/html/.htpasswd prof
```

Pour vérifier si tout fonctionne nous allons taper la commande suivante: `nano /var/www/html/htpasswd`

```
prof:$apr1$kZQyEG8x$PsT7PIAmcH78.qVs5h10c0
```

TP SERVEURS

- Dans le fichier htpasswd nous allons définir des autorisations pour effectuer c modification nous pouvons utiliser le ftp puis sélectionner le fichier et l'édité
- AuthUserFile : Désigne le chemin du fichier où seront stockés les identifiants qui seront valident pour la connexion*
- AuthName "Accès restreints - Veuillez-vous authentifier" : C'est ce qui va être affiché sur la boîte de dialogue d'authentification qui apparaîtra sur l'écran de l'utilisateur.
- require valid-user : Désigne les conditions qui doivent être validées pour laisser passer l'utilisateur. Ici il ne s'agit que de la validation des identifiants saisis par l'utilisateur. Tous les utilisateurs valident pourront donc accéder à la ressource en question

```
AuthUserFile /var/www/html/.htpasswd
AuthName "Accès restreints - Veuillez vous authentifier"
AuthType Basic
Require valid-user
```

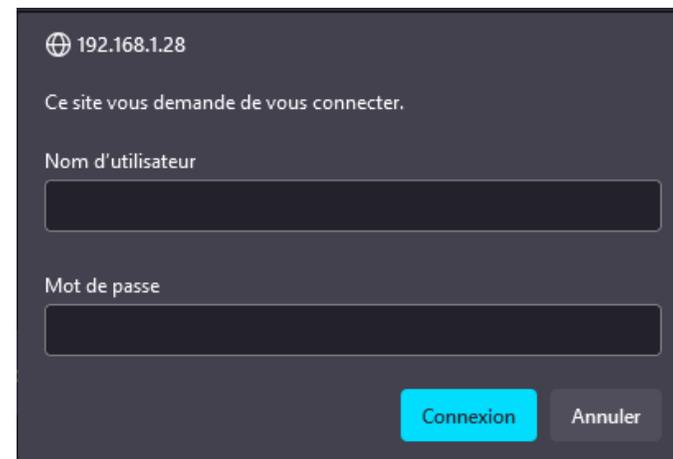
TP SERVEURS

- Nous allons ensuite procéder à la configuration d'apache pour se faire nous allons utiliser la commande suivante `nano /etc/apache/sites-available/000-default.conf`
- Voici ce qu'il faut mettre dans le fichier
- Le paramètre "**AllowOverride**" permet de déterminer quel fichier ou configuration est autorisé à prendre en priorité la configuration d'Apache 2. Si on le paramètre à "**All**", Apache 2 pourra lire les fichier ".**htaccess**"

```
ServerAdmin webmaster@localhost
DocumentRoot /var/www/html
<Directory /var/www/>
AllowOverride all
</Directory>
# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
```

TP SERVEURS

- Comme nous pouvons le voir, les paramètres ont bien été pris en compte



192.168.1.28

Ce site vous demande de vous connecter.

Nom d'utilisateur

Mot de passe

Connexion Annuler