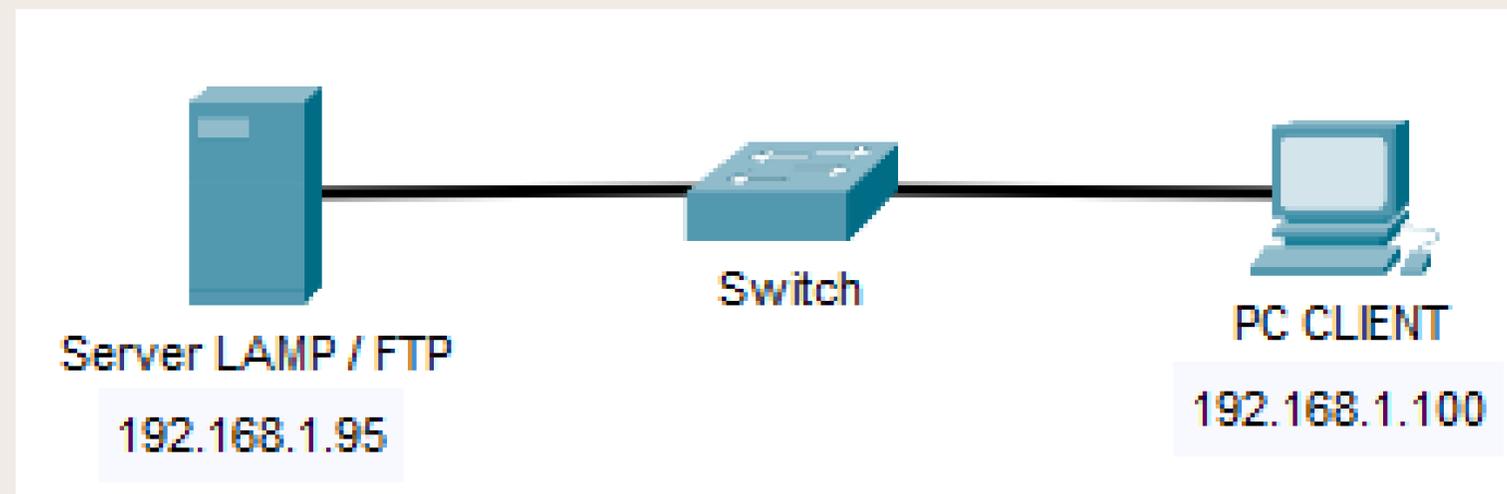


PRÉSENTÉ PAR DRUELLE NICOLAS

TP-SERVEURS

TOPOLOGIE



CAHIER DES CHARGES

- Mise en place d'un serveur LAMP

- Mise en place d'un serveur FTP

- Mise en place des utilisateurs prof et eleve en utilisant les mots de passe donnée

- Administration des droits

- Création de deux pages HTML relié entre elle

- Mise en place d'une authentification sur la page1.html

- Réalisations des différents test de connexion ainsi que des tests des droits sur les dossiers

QU'EST QU'UN SERVER LAMP

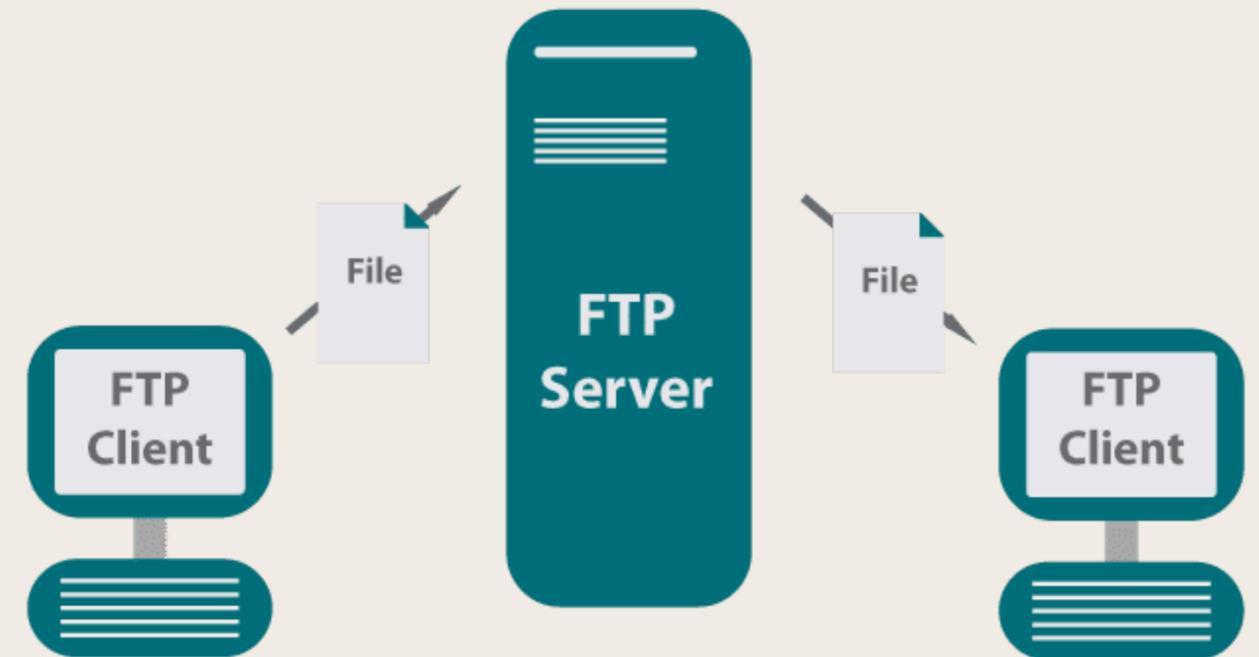
Un serveur LAMP est un ensemble de logiciels utilisée pour héberger des sites web. Il est composé de Linux qui sert de système d'exploitation, Apache qui sert de serveur web, MySQL qui permet de réaliser les bases de données et PHP qui est un langage de scripting. Cette ensemble permet de réaliser des applications web dynamiques.



QU'EST-CE QUE LE FTP ?

FTP signifie File Transfer Protocol (protocole de transfert de fichier en français). Ce protocole de communication est utilisé pour l'échange de fichiers entre un serveur et un client.

Les ports par défaut pour le FTP sont le port 21 pour la connexion de commande et le port 20 pour la connexion de données.



MISE EN PLACE SERVER LAMP

LOGICIELS UTILISÉS

Pour la mise en place du server LAMP je vais utiliser un système d'exploitation linux ici Debian 12. Sur celle-ci je vais par la suite installer divers logiciels comme Apache, MySQL ainsi que PHP afin de pouvoir mettre en place le serveur.

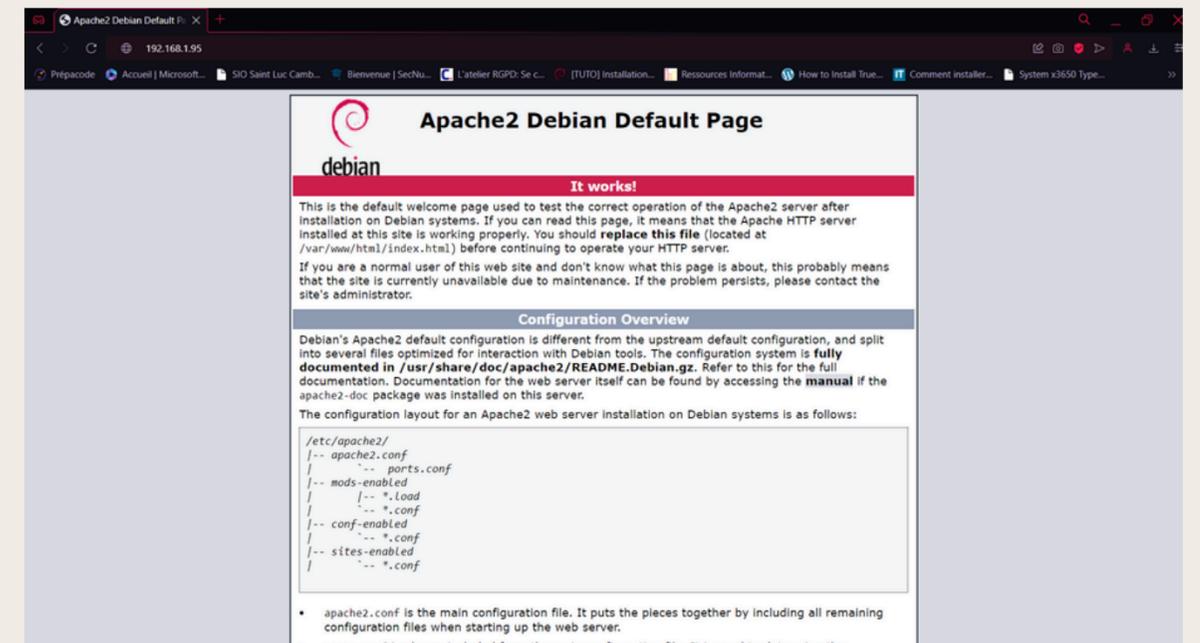


MISE EN PLACE SERVER LAMP

Dans un premier temps je vais installer apache2
qui va nous servir de serveur WEB

Une fois installer je vérifie sur mon navigateur si
le serveur est fonctionnel. Nous pouvons voir
sur le screen que le serveur est fonctionnel

```
root@server:~# apt install apache2
```



MISE EN PLACE SERVER LAMP

Ensuite je vais installer MySQL qui va nous servir pour les bases de données

Une fois MySQL installer je passe à la sécurisation avec la commande “mysql_secure_installation”. Dans un premier temps je mets en place un mot de passe root qui est “Mot2pass”. Ensuite je reponds Yes pour supprimer les utilisateurs anonymes, pour desactiver la connexion root à distance, la suppression des bases de données de test et enfin je recharge les tables de privilèges.

```
root@server:~# apt install mysql-server
```

```
root@server:~# mysql_secure_installation
```

MISE EN PLACE SERVER LAMP

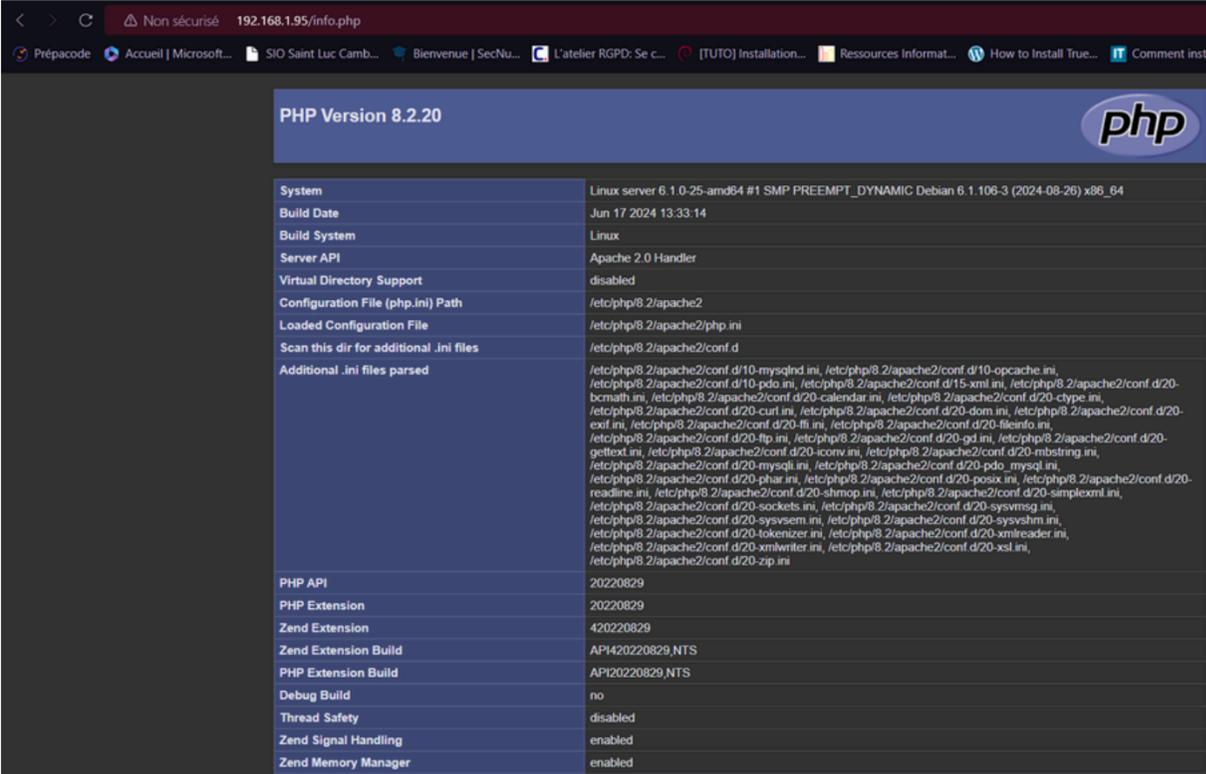
Enfin je procède à l'installation de PHP et met en place le fichier info.php avec la commande “nano /var/www/html/info.php” dans celle-ci je met le code qui permet d'afficher les informations de PHP.

Nous pouvons voir par la suite quand je me connecte à “192.168.1.95/info.php” tout les informations sont présente.

```
root@server:~# apt install php libapache2-mod-php php-mysql
```

```
root@server:~# nano /var/www/html/info.php
```

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



PHP Version 8.2.20	
System	Linux server 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64
Build Date	Jun 17 2024 13:33:14
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.2/apache2
Loaded Configuration File	/etc/php/8.2/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.2/apache2/conf.d
Additional .ini files parsed	/etc/php/8.2/apache2/conf.d/10-mysqld.ini, /etc/php/8.2/apache2/conf.d/10-opcache.ini, /etc/php/8.2/apache2/conf.d/10-pdo.ini, /etc/php/8.2/apache2/conf.d/15-xml.ini, /etc/php/8.2/apache2/conf.d/20-bcmath.ini, /etc/php/8.2/apache2/conf.d/20-calendar.ini, /etc/php/8.2/apache2/conf.d/20-ctype.ini, /etc/php/8.2/apache2/conf.d/20-curl.ini, /etc/php/8.2/apache2/conf.d/20-dom.ini, /etc/php/8.2/apache2/conf.d/20-exif.ini, /etc/php/8.2/apache2/conf.d/20-ftp.ini, /etc/php/8.2/apache2/conf.d/20-gd.ini, /etc/php/8.2/apache2/conf.d/20-gettext.ini, /etc/php/8.2/apache2/conf.d/20-iconv.ini, /etc/php/8.2/apache2/conf.d/20-mbstring.ini, /etc/php/8.2/apache2/conf.d/20-mysql.ini, /etc/php/8.2/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.2/apache2/conf.d/20-phar.ini, /etc/php/8.2/apache2/conf.d/20-posix.ini, /etc/php/8.2/apache2/conf.d/20-readline.ini, /etc/php/8.2/apache2/conf.d/20-shmop.ini, /etc/php/8.2/apache2/conf.d/20-simplexml.ini, /etc/php/8.2/apache2/conf.d/20-sockets.ini, /etc/php/8.2/apache2/conf.d/20-sysmsg.ini, /etc/php/8.2/apache2/conf.d/20-sysvshm.ini, /etc/php/8.2/apache2/conf.d/20-sysvsem.ini, /etc/php/8.2/apache2/conf.d/20-tokenizer.ini, /etc/php/8.2/apache2/conf.d/20-xmlreader.ini, /etc/php/8.2/apache2/conf.d/20-xmlwriter.ini, /etc/php/8.2/apache2/conf.d/20-xsl.ini, /etc/php/8.2/apache2/conf.d/20-zip.ini
PHP API	20220829
PHP Extension	20220829
Zend Extension	420220829
Zend Extension Build	API20220829.NTS
PHP Extension Build	API20220829.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled

MISE EN PLACE SERVER FTP

LOGICIELS UTILISÉS

Pour la mise en place du serveur FTP je vais utiliser vsFTP. Je vais aussi utiliser Filezilla client pour réaliser les différents test.



MISE EN PLACE SERVER FTP

Dans un premier temps j'installe vsftpd.

```
root@server:~# apt install vsftpd
```

Une fois installer je crée un copie du fichier vsftpd.conf qui va me servir de backup.

```
root@server:~# cp /etc/vsftpd.conf /etc/vsftpd.conf.orig
```

MISE EN PLACE SERVER FTP

Une fois la backup faites je vais commencer à configurer le server en faisant “nano /etc/vsftpd.conf”.

Je vais configurer plusieurs choses dedans comme l'autorisation des utilisateurs locaux, empêcher les utilisateurs de naviguer hors de leur répertoire personnel, la désactivation de l'accès anonyme et la configuration des permissions des utilisateurs spécifiques.

```
root@server:~# nano /etc/vsftpd.conf
```

```
local_enable=YES  
write_enable=YES  
  
chroot_local_user=YES  
allow_writeable_chroot=YES  
  
anonymous_enable=NO  
  
user_sub_token=$USER  
local_root=/var/www  
user_config_dir=/etc/vsftpd user conf
```

MISE EN PLACE SERVER FTP

CRÉATION DES UTILISATEURS

Pour la création des utilisateurs j'ai utilisé la commande "adduser" suivi du nom de l'utilisateur une fois cette commande faites on nous demande de mettre un mot de passe pour chaque utilisateur "azerty" pour "eleve" et "qwerty" pour "prof".

```
root@server:~# adduser eleve
```

```
root@server:~# adduser prof
```

MISE EN PLACE SERVER FTP

ATTRIBUTION D'UN REPERTOIRE AUX UTILISATEURS ET DROITS

Une fois les utilisateurs créés je vais attribuer le répertoire “/var/www” aux 2 utilisateurs.

Les droits en 775 accordent les permissions de lecture, d'écriture et d'exécution à l'utilisateur propriétaire et au groupe. Ainsi que des droits de lecture et d'exécution aux autres.

```
root@server:~# usermod -d /var/www/ eleve
```

```
root@server:~# usermod -d /var/www/ prof
```

```
root@server:~# chown -R www-data:www-data /var/www
```

```
root@server:~# usermod -aG www-data prof
```

```
root@server:~# chmod 775 /var/www
```

```
root@server:~# chmod 775 /var/www/html
```

MISE EN PLACE SERVER FTP

CRÉATION DES “VSFTPD_USER_CONF”

Dans un premier temps je vais créer le dossier “vstfpd_user_conf”.

Une fois le dossier créer je vais mettre en place un fichier pour chaque utilisateurs créer précédemment.

Pour l'utilisateur “eleve” je désactive les droits d'utilisateur contrairement à l'utilisateur “prof” à qui je les autorise.

Une fois cela fait je redémarre le serveur ftp.

```
root@server:~# mkdir /etc/vsftpd_user_conf
```

```
root@server:~# nano /etc/vsftpd_user_conf/prof
```

```
GNU nano 7.2
```

```
write_enable=YES
```

```
root@server:~# nano /etc/vsftpd_user_conf/eleve
```

```
GNU nano 7.2
```

```
write_enable=NO
```

```
root@server:~# systemctl restart vsftpd
```

CRÉATION DES 2 PAGES HTML

```
root@server:~# nano /var/www/html/index.html
```

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Accueil</title>
</head>
<body>
  <h1>Bienvenue sur la page d'accueil</h1>
  <p>Cliquez sur le lien ci-dessous pour accéder à la page 1.</p>
  <a href="page1.html">Accéder à la Page 1</a>
</body>
</html>
```

Aperçu du code d'index.html

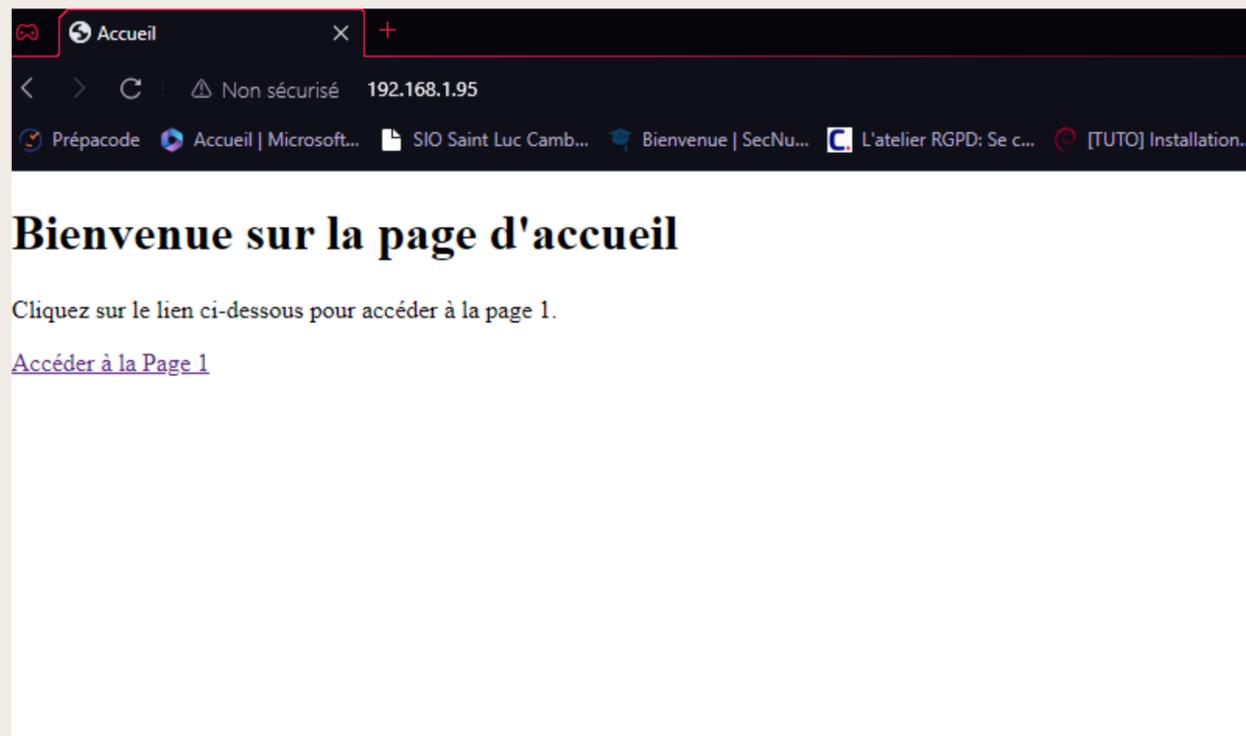
```
root@server:~# nano /var/www/html/page1.html
```

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Page 1</title>
</head>
<body>
  <h1>Ceci est la Page 1</h1>
  <p>Vous avez accédé à cette page depuis l'index.</p>
  <a href="index.html">Retour à la page d'accueil</a>
</body>
</html>
```

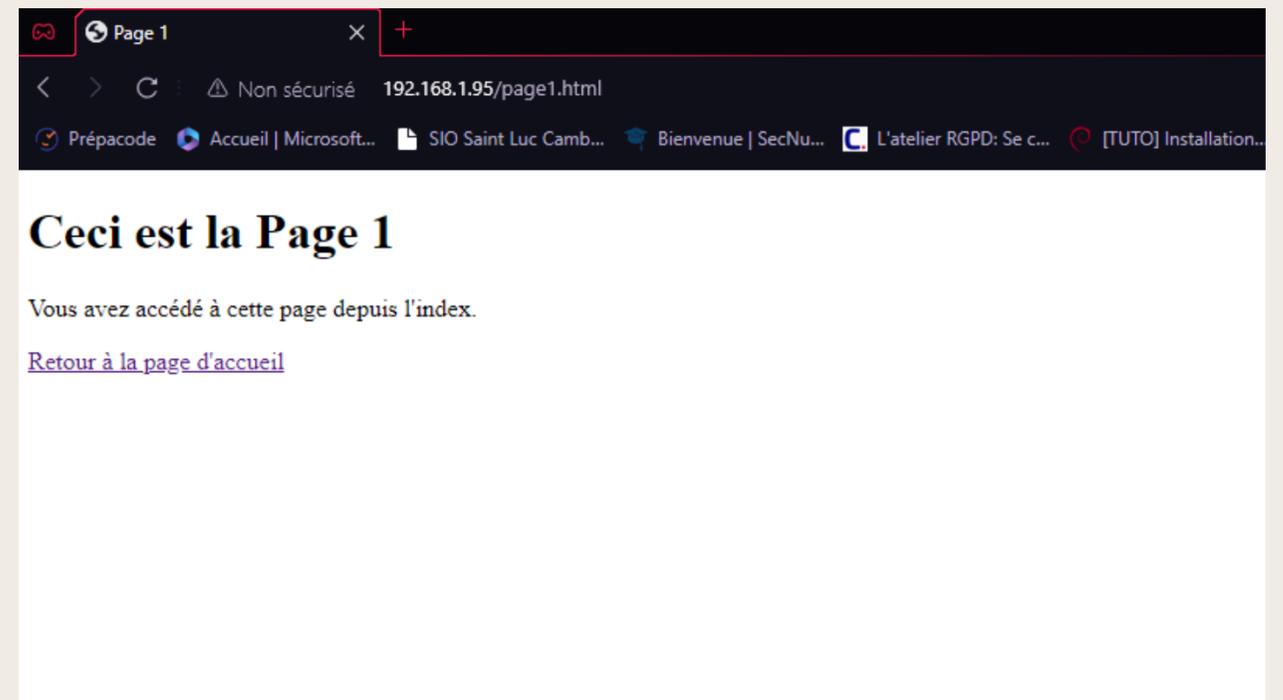
Aperçu du code de page1.html

CRÉATION DES 2 PAGES HTML

APERÇU DES PAGES



Aperçu de la page d'index.html



Aperçu de la page de page1.html

MISE EN PLACE DU HTACCESS

Dans un premier temps nous allons générer le fichier .htpasswd qui contiendra les utilisateurs avec la commande “htpasswd” une fois cette commande faite on va nous demander un mot de passe pour chaque utilisateurs ici “azerty” pour “eleve” et “qwerty” pour “prof”.

```
root@server:~# htpasswd /etc/apache2/.htpasswd eleve
```

```
root@server:~# htpasswd -c /etc/apache2/.htpasswd prof
```

```
root@server:~# nano /etc/apache2/.htpasswd
```

```
GNU nano 7.2
```

```
prof:$apr1$XT4dYON6$S2et.Z03Hw8NmusJj8wvZ.
```

```
eleve:$apr1$BBEVx.vt$Q/bc7x1GSJ9abMiPC1m/5/
```

MISE EN PLACE DU HTACCESS

CRÉATION DU FICHIER .HTACCESS QUI PROTÈGERA “PAGE1.HTML”

Pour la création du fichier .htaccess il y’a plusieurs paramètres comme :

AuthType ici en Basic qui veut dire que l’authentification est en HTTP Basic

Authentification

AuthName qui est le texte dans la boîte de dialogue.

AuthUserFile qui est le chemin vers le fichier contenant les utilisateurs autorisés

Require valid-user qui indique qu’un utilisateur doit être authentifié pour accéder à la page1.html.

```
root@server:~# nano /var/www/html/page1.htaccess
```

```
<Files "page1.html">  
  AuthType Basic  
  AuthName "Restricted Content"  
  AuthUserFile /etc/apache2/.htpasswd  
  Require valid-user  
</Files>
```

MISE EN PLACE DU HTACCESS

DROIT DU “PAGE1.HTACCESS” ET CONFIGURATION APACHE

Le chmod 644 va permettre à apache de pouvoir lire le fichier.

```
root@server:~# chmod 644 /var/www/html/page1.htaccess
```

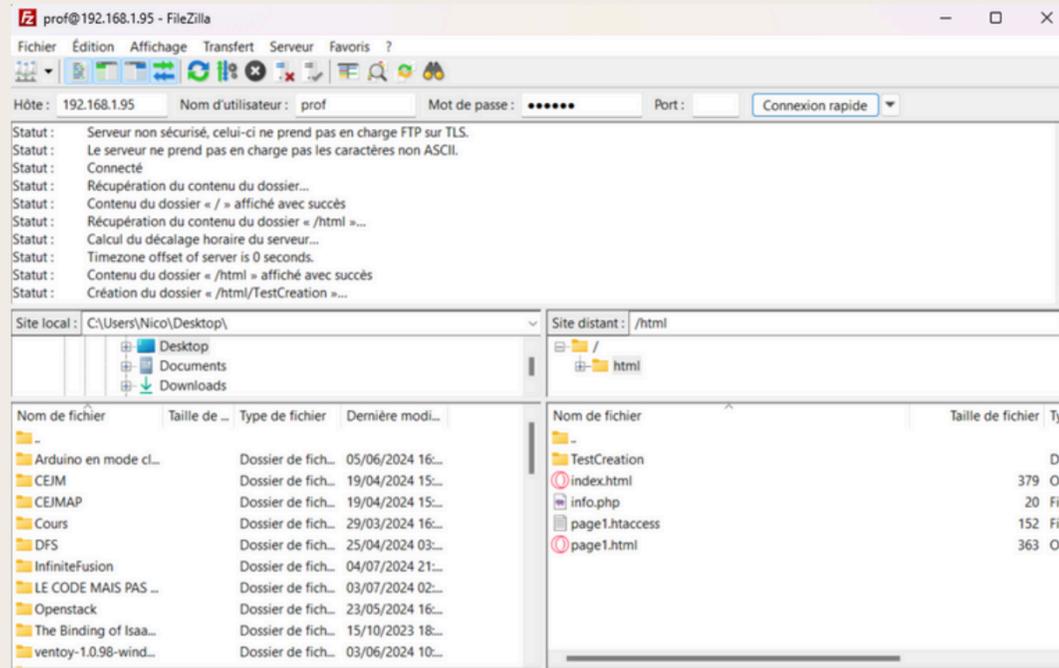
```
root@server:~# nano /etc/apache2/apache2.conf
```

Dans la configuration d'apache 2 je vais changer le “Require all denied” en “Require all granted” ce qui va permettre à apache de pouvoir accéder au fichiers de ses répertoires.

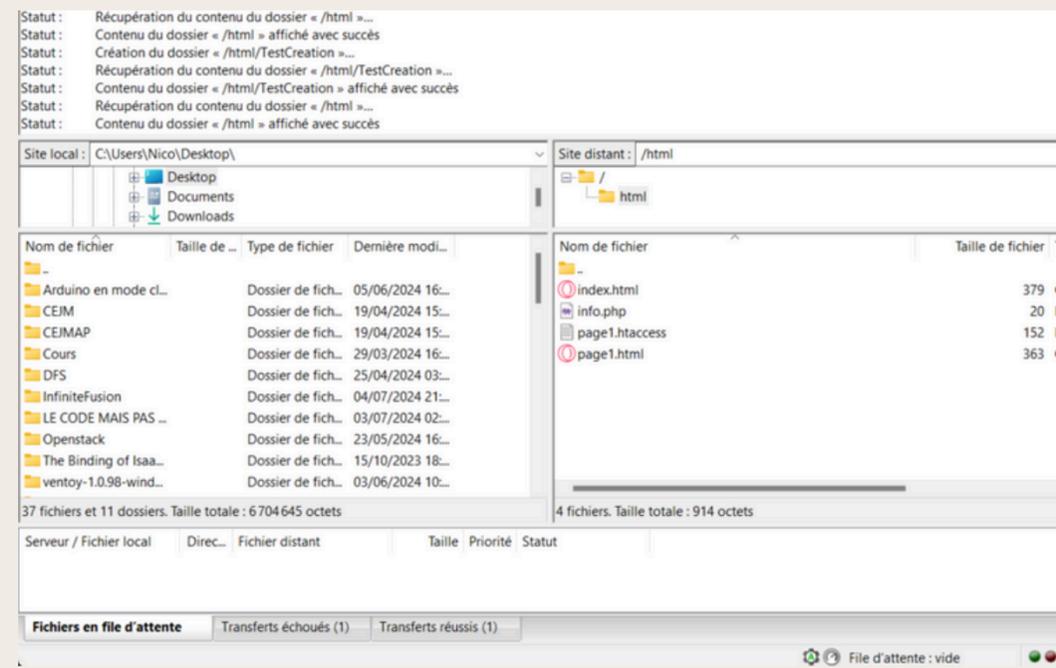
```
<Directory /var/www/>  
    Options Indexes FollowSymLinks  
    AllowOverride AuthConfig  
    Require all granted  
</Directory>  
  
<Directory /var/www/html>  
    Options Indexes FollowSymLinks  
    AllowOverride AuthConfig  
    Require all granted  
</Directory>
```

TEST FTP

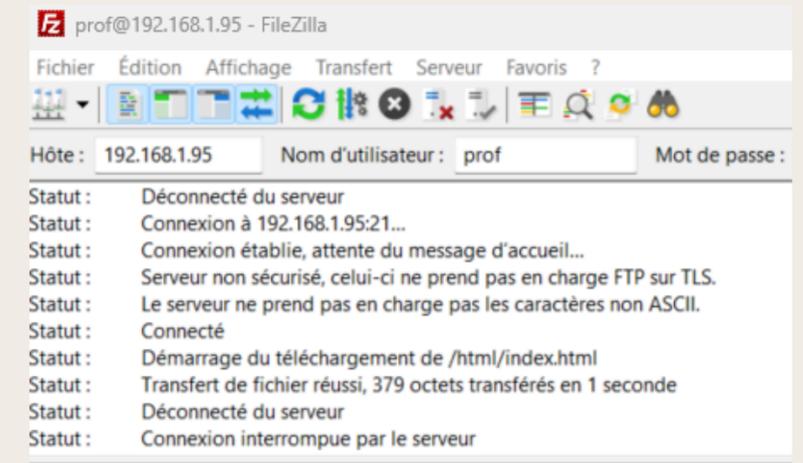
TEST "PROF"



Test création



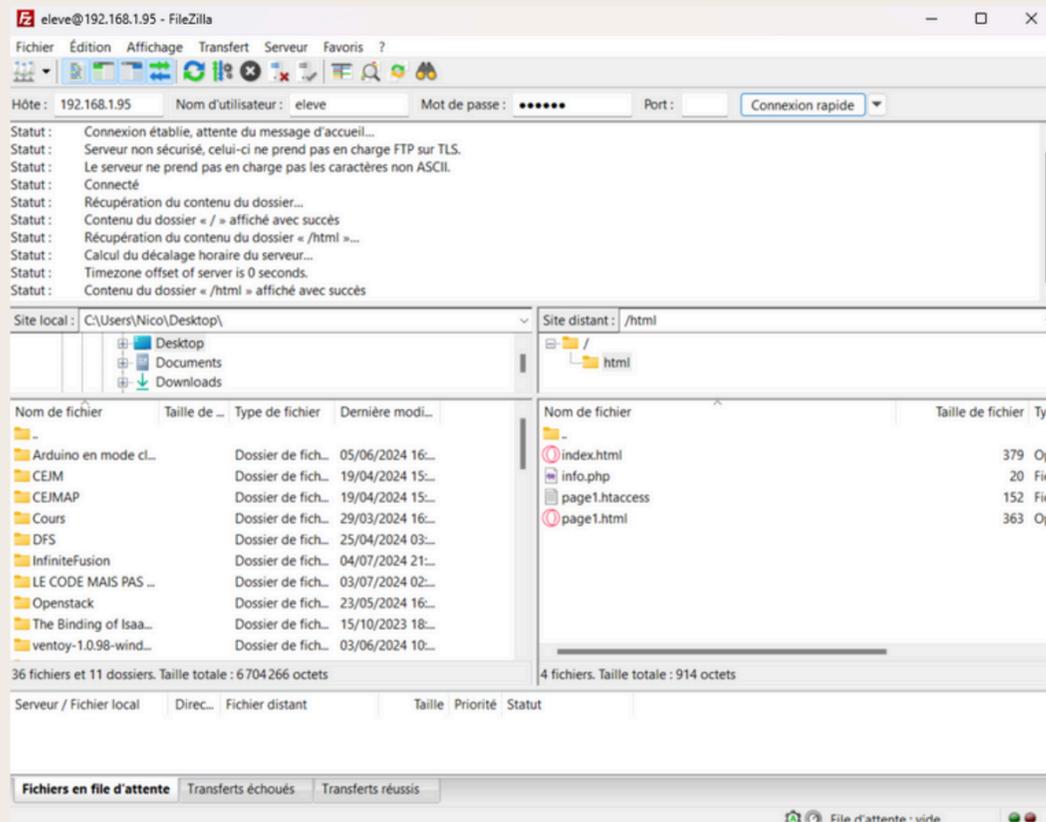
Test suppression



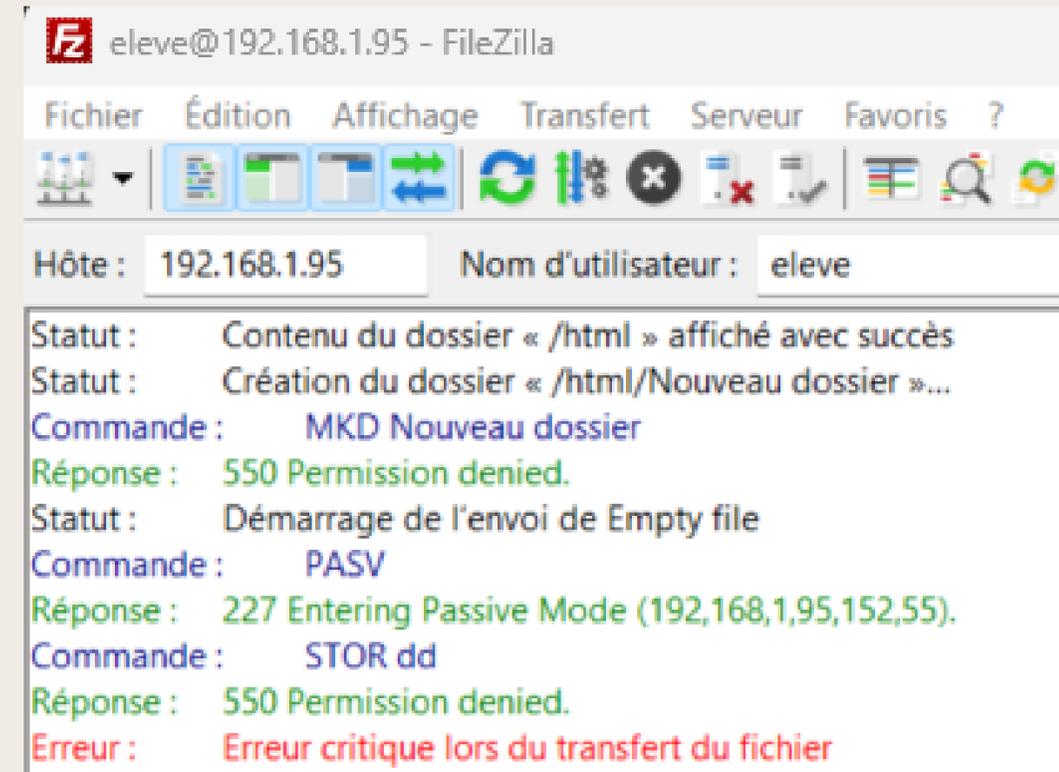
Test transfert

TEST FTP

TEST "ELEVE"

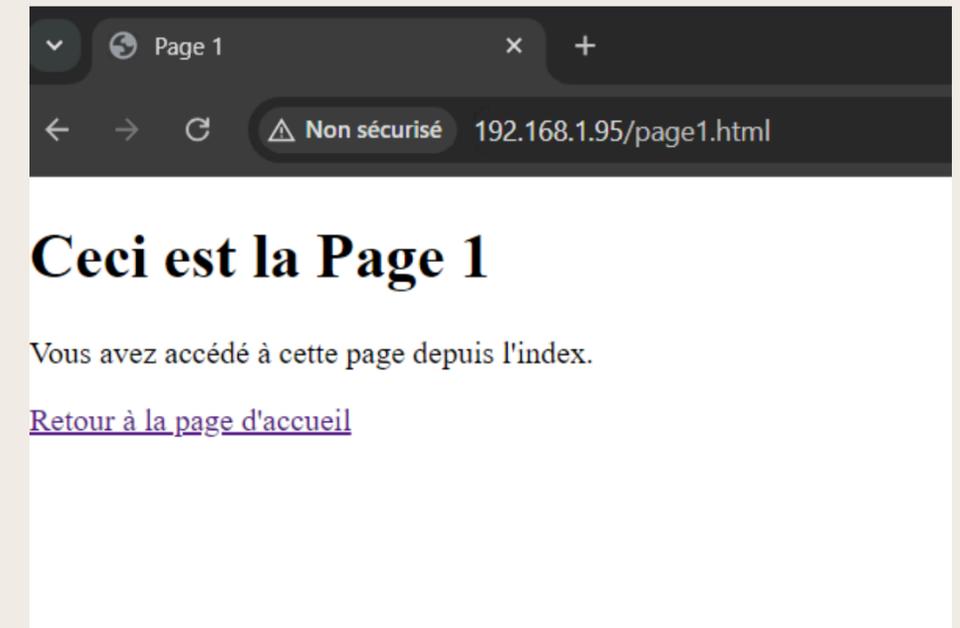
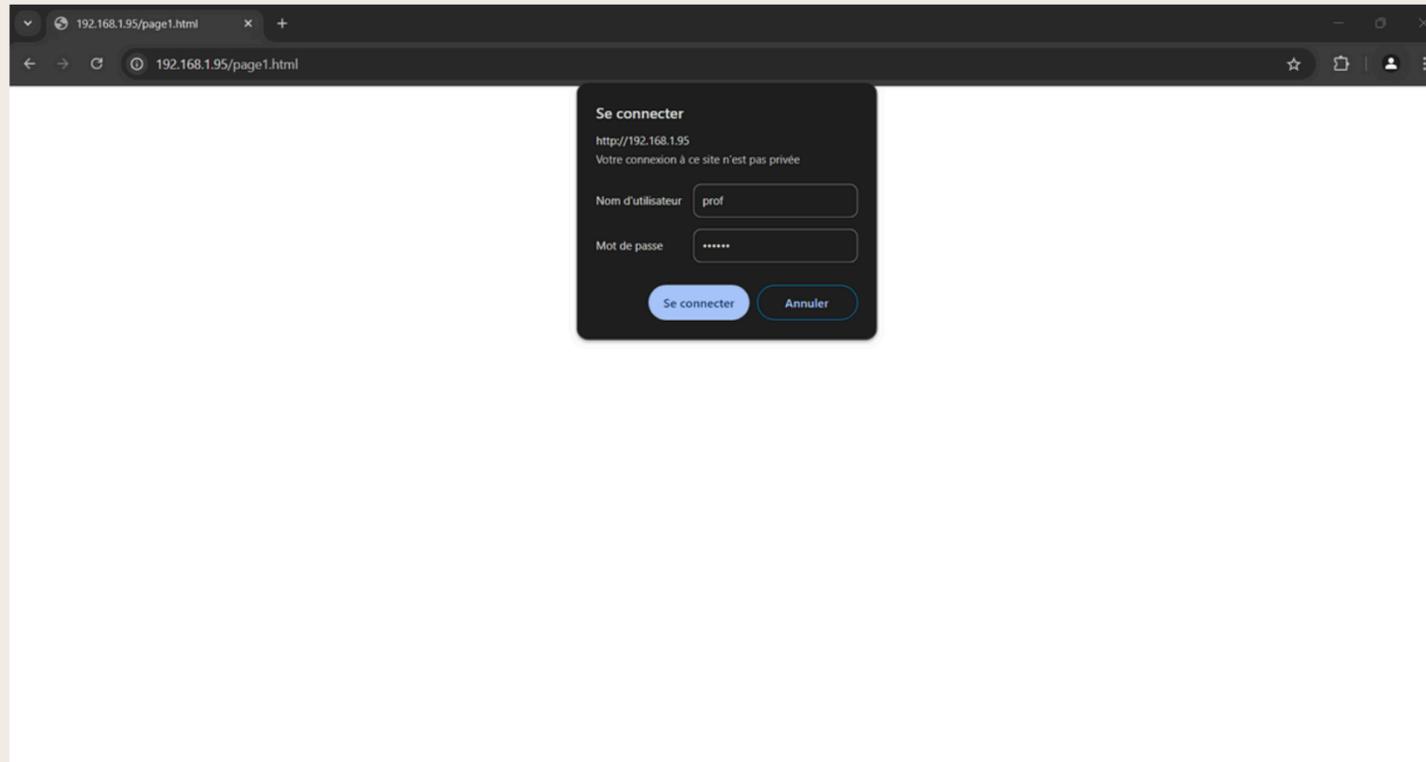


Test accès



Test création

TEST HTACCESS



Test connexion avec
utilisateur prof