

R1.02 - Interfaces Web

TD1 Serveurs Web

DUT info Vannes

G. Kerbellec 2022

Objectif

Cette séance a pour but :

- 1- mettre en place un serveur web sous windows
- 2- mettre en place un serveur web sous linux
- 3- connaître la configuration d'Apache
- 4- savoir lire les logs d'Apache

1- Serveur local sous Windows

Vous allez transformer votre machine en serveur web.

Vous pourrez donc accéder à un répertoire de votre machine par un navigateur.

Sur votre machine vous pourrez tester le bon fonctionnement localement en allant sur l'URL *http://localhost* ou encore *http://127.0.0.1*

Depuis une autre machine de votre réseau, vous pourrez également entrer l'adresse IP de votre machine dans le navigateur (ex: *http://192.168.1.18*) . Vous serez peut-être bloqué par un pare-feu ou une mauvaise configuration. C'est optionnel.

Dans un premier temps, vous testerez l'installation de WAMP.

Cette installation se fait sous Windows.

Téléchargez la plateforme WAMP ici : <https://www.wampserver.com>

Installez WAMP

WAMP contient le service Apache qui est le programme permettant de donner accès à des pages web, vérifiez que le service Apache fonctionne par clics gauches/droits sur l'icône de Wamp (à droite sur votre barre d'icônes windows).

Créez un fichier texte *toto.txt* dans lequel vous écrirez quelques mots (utilisez notepad++ ou atom ou geany ou nano ou n'importe quel éditeur de texte)

Placez ce fichier dans le répertoire utilisé par WAMP

Vérifiez que vous accédez à ce fichier en utilisant votre navigateur

A présent, créez un répertoire, et répartissez plusieurs fichiers, puis testez leur disponibilité via l'interface web.

Enfin, testez l'interprétation HTML en créant un fichier *test.html* contenant des balises simples :

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>MON SUPER SERVEUR</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
MA <STRONG>SUPER</STRONG> PAGE WEB
```

```
</BODY>
```

```
</HTML>
```

Astuce, si vous nommez votre page *index.html*, il ne sera pas nécessaire de compléter votre URL avec le nom de la page, elle sera utilisée par défaut.

Il est conseillé de créer une page *index.html* par répertoire, même une vide, cela évite de laisser apparaître toute votre arborescence. Faites le test !

2- Serveur local sous Linux (Ubuntu)

Soyez certain que votre distribution Ubuntu est à jour :

```
sudo apt update
```

```
sudo apt upgrade
```

Puis installez le package apache2 (serveur web apache) :

```
sudo apt install apache2
```

Regardez votre URL <http://localhost> depuis un navigateur

Démarrez le serveur web :

```
sudo service apache2 start
```

Regardez à nouveau l'URL <http://localhost>, vous devriez voir une page d'accueil d'Apache

Stoppez le serveur web :

```
sudo service apache2 stop
```

Vérifiez l'URL, et relancez quand vous le souhaitez le serveur, vous pouvez également utiliser :

```
sudo service apache2 restart
```

A présent, vous pouvez tester vos propres fichiers et les déposer dans le répertoire :

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

(utilisez les commandes linux de base pour vous déplacer et travailler dans l'arborescence :
ls cd pwd touch mkdir ou encore l'éditeur *nano*)

3- Configuration d'Apache

a- ports.conf

Par défaut, le port du service apache est 80.

Quand on utilise un navigateur avec l'url *http://localhost*, par défaut ceci est équivalent à donner le port 80 dans l'url *http://localhost:80*

Veuillez indiquer les différentes étapes et tests vous permettant de changer le port de Apache en vous appuyant sur le fichier de configuration */etc/apache2/ports.conf*

b- apache2.conf

Utilisez le fichier de configuration *apache2.conf* pour changer le répertoire de stockage de votre site.

Expliquez en français chacune des options du fichier *apache2.conf*

c- VirtualHost

Recherchez et expliquez le principe des VirtualHosts, présentez un exemple fonctionnel sur votre machine.

<http://httpd.apache.org/docs/2.2/fr/vhosts/examples.html>

d- .htaccess

Expliquez quelles sont les possibilités du fichier *.htaccess*

Mettez en place un contrôle d'accès par mot de passe à votre site, en suivant une procédure semblable :

<https://www.it-connect.fr/proteger-une-page-web-avec-htaccess-et-httpasswd>

4- Logs d'Apache

Décrivez l'emplacement des fichiers de logs d'Apache et l'intérêt / la fréquence pour les utiliser. Indiquez les logs et les erreurs classiques à partir de la documentation.

<https://httpd.apache.org/docs/2.4/fr/logs.html>