

TDM05 de Technologies Web

Serveur Web et protocole HTTP

ISIA3 - INSA Rouen
CORRECTION

1 Serveur Web Apache HTTP

Vous avez à votre disposition une configuration virtual box contenant un serveur Web de type Apache (voir documentation : <http://wiki.insa-rouen.fr/doku.php?id=insa:asi:apache:start>).

Ce serveur est configuré pour exécuter les scripts PHP dans tous les répertoires accessibles (sauf restrictions particulières) et inclus le SGBD SQLite. Parmi les alias disponibles sur ce serveur web, vous trouverez :

- la racine du serveur (<http://SERVEUR/>) pointant vers `/var/www/`;
- <http://SERVEUR/cgi-bin/> : répertoire d'exécution des scripts CGI pointant vers `/usr/lib/cgi-bin/`;

1. Déposez des ressources statiques à la racine du serveur et accédez-y à l'aide d'un navigateur.
2. Créez une hiérarchie de répertoires dans la racine du serveur et vérifiez le comportement de la navigation dans cette hiérarchie depuis votre navigateur.
3. Déposez des ressources dynamiques côté client à la racine du serveur et testez-le à l'aide d'un navigateur. Vérifiez le comportement lors de l'inclusion de scripts définis dans un second fichier présent sur le serveur, ainsi que pour des scripts externes (e.g. jQuery en local ou sur un CDN).
4. Corrigez le script `index.php` situé à la racine (il manque l'accolade fermante) et testez-le en vérifiant le code source de la page reçu par votre navigateur. Il s'agit d'un script PHP (dynamique côté serveur).

Correction

Pour pouvoir accéder aux homes, il faut bien vérifier les droits d'accès sur le groupe auquel appartient l'agent Apache (attention, il peut changer d'une année sur l'autre en fonction de la version d'Apache). Les étudiants doivent donc :

- donner accès en exécution à leur répertoire personnel (`chmod g+x~`)
- donner accès en exécution, lecture et écriture à leur répertoire personnel (`dhmod g+x~public_html`)

Pour les scripts CGI, une moulinette crée les liens symboliques depuis `/var/www/cgi-bin/LOGIN` vers `/home/pnom/cgi-bin`. Vérifier que le script a bien tourné avant le TP, avec les étudiants de la bonne année. La moulinette a pour pré-requis que les comptes étudiants appartiennent au groupe `asi4` (et dorénavant `asi3`). Les scripts CGI seront alors accessibles via l'URL racine <http://asi-technoweb.insa-rouen.fr/cgi-bin/LOGIN/>

livredor-get.html (GET externe)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Formulaire d'inscription au forum ASI</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <link href="Ressources/style.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
    <header id="top">
      
      <h1>Forum ASI</h1>
    </header>
    <div id="main">
      <p id="probleme"></p>
      <form name="formulaire" action="/cgi-bin/inscription-get.cgi" method="get">
        <fieldset>
          <legend>Formulaire d'nscription</legend>
          <label for="name">Nom : </label><input placeholder="Saisissez votre nom" id="name" name="name" type="text" size="30" />
          <label for="surname">Prénom : </label><input placeholder="Saisissez votre prénom" id="surname" name="surname" type="text" size="30" />
          <label for="email">E-mail : </label><input placeholder="Saisissez votre email" id="email" name="email" type="text" size="30" />
        </fieldset>
      </form>
    </div>
  </body>
</html>
```

```

    <label>Statut : </label>
    <input type="radio" id="enseignant" name="statut" value="enseignant" /><label class="labelradio"
for="enseignant"> Enseignant</label>
    <input type="radio" id="etudiant" name="statut" value="etudiant" /> <label class="labelradio" for
="etudiant"> Etudiant</label><br />
    <label for="annee">Si étudiant, année : </label>
    <select name="annee" id="annee">
    <option value="1">1</option>
    <option value="2">2</option>
    <option value="3">3</option>
    <option value="4">4</option>
    <option value="5">5</option>
    </select><br />
    <label for="photo">Photo :</label> <input type="file" name="maPhoto" id="photo" />
</fieldset>
    <input type="submit" />
    <input type="reset" value="Effacer"/>
</form>
</div>
<footer>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.4/jquery.min.js"></script>
    <script src="Scripts/script-jQuery.js"></script>
</footer>
</body>
</html>

```

GET externe

```
<form name="formulaire" action="http://asi-technoweb.insa-rouen.fr/cgi-bin/inscription-get.cgi">
```

GET interne

```
<form name="formulaire" action="/cgi-bin/inscription-get.cgi" method="get">
```

GET home

```
<form name="formulaire" action="/cgi-bin/apauchet/inscription-get.cgi" method="get">
```

POST externe

```
<form name="formulaire" action="http://asi-technoweb.insa-rouen.fr/cgi-bin/inscription-post.cgi" method="post" enctype
="application/x-www-form-urlencoded">
```

POST interne

```
<form name="formulaire" action="/cgi-bin/inscription-post.cgi" method="post" enctype="application/x-www-form-urlencoded"
">
```

POST home

```
<form name="formulaire" action="/cgi-bin/apauchet/inscription-post.cgi" method="post" enctype="application/x-www-form
-urlencoded">
```

2 Serveur Node.js

Sur les machines de l'INSA est également installé un serveur Node.js.

1. Depuis le répertoire `Node/hello`, lancez un serveur Node.js (commande `node app.js`) et effectuez une requête depuis un navigateur afin de récupérer le texte généré dynamiquement.
2. Modifiez le code Javascript pour que celui produise maintenant du code HTML.
3. Afin de gérer le routage des ressources statiques, Express sera utilisé. Lancez maintenant un serveur depuis `expressApp`. Ajoutez des ressources statiques dans le répertoire public et testez différentes routes.

3 Protocole Http

Références : <http://www.w3.org/Protocols/rfc2616/rfc2616.html>

Pour cette partie, vous testerez vos requêtes auprès de la Virtual Box (serveur Apache HTTP), d'un serveur Node.js et d'un serveur extérieur.

Remarque : cette partie du TP nécessite une connexion par telnet (`telnet asi-technoweb.insa-rouen.fr 80`) ou netcat sur le port 80.

1. Repérer la date de dernière modification d'une page. Effectuer une requête sur la page :
 - test 1 : la rapatrier si elle a été modifiée depuis une certaine date ;
 - test 2 : la rapatrier si elle n'a pas été modifiée depuis une certaine date ;
2. Effectuer une requête ne récupérant que les caractères 2 à 5 d'une page web.
3. Écrire une requête effectuant une négociation sur le type de média demandé :
 - test 1 : privilégier les images au format jpeg par rapport au format png ;
 - test 2 : l'inverse.

4. Écrire une requête avec négociation de contenu sur la langue.
5. Écrire une requête HTTP/1.1 sans connexion persistante
6. Écrire 2 requêtes retournant les codes suivants : 404, 501.

Correction

1. *Test 1 :*

```
GET /~apauchet/index.html HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
If-Modified-Since: ...
```

Test 2 :

```
GET /~apauchet//index.html HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
If-Unmodified-Since: ...
```

Remarque : le format de la date "Sun, 09 Sep 2007 17 :00 :00 GMT" par exemple.

- 2.

```
GET /~apauchet/index.html HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
Range: bytes=1-4
```

Remarque : le premier caractère est à 0.

3. *Test 1 :*

```
HEAD /~apauchet/image HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
Accept: image/jpeg; q=1, image/png; q=0.5
```

Test 2 :

```
HEAD /~apauchet/image HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
Accept: image/jpeg; q=0.5, image/png; q=1
```

Remarque : 2 fichiers coexistent, image.png et image.jpg.

- 4.

```
GET /~apauchet/index.html HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
Accept-Language: en;q=0.5, fr;q=1
```

Remarque : 2 fichiers coexistent, index.fr.html et index.en.html.

- 5.

```
HEAD /~apauchet/index HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
Connection: close
```

Remarque : Par défaut pour un telnet, la connexion n'est pas persistante avec un GET ; il faut donc tester avec un HEAD. Pour netcat, la connexion est persistante avec un GET.

6. *Code 404, pour une page inconnue :*

```
HEAD /~apauchet/Fichier-N-Existant.pas HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
```

Code 501, pour une commande inconnue :

```
COMMANDE-INCONNUE /~apauchet/index HTTP/1.1
Host: asi-technoweb.insa-rouen.fr
```

4 Utilisation de scripts CGI

Reprenez le Serveur Apache HTTP du premier exercice.

1. Testez le script CGI `hello-perl.cgi` fourni en annexe.
2. Déposez sur la racine du serveur web la page d'inscription au forum ASI réalisée lors des TDM précédents et testez-la depuis votre navigateur. Modifiez cette page pour que le traitement du formulaire soit effectué en GET par le CGI `inscription-get.cgi`. Vous testerez ce script par une requête externe au serveur Web, puis par une requête interne.
3. Modifiez la page d'inscription au forum pour que le traitement du formulaire soit maintenant effectué en POST. Effectuez les mêmes tests que précédemment avec le CGI `inscription-post.cgi`.

Remarques

1. Continuez à vérifier vos pages (<http://validator.w3.org/> ou <http://www.xmlvalidation.com/>).
2. Afin de faciliter le débogage de vos scripts et si vous utilisez Firefox comme navigateur, n'hésitez pas à utiliser l'outil de Développement Web.
3. À l'issue de la séance, vous aurez accès à la correction de ce TDM au format PDF.
4. **Déposez votre compte-rendu sur moodle sous la forme d'un fichier PDF nommé `TDM05-NomPrenom.pdf`, chez chacune des 2 personnes du binôme.**