

IUT d'Orléans. Département Informatique. Année 2020-2021

Développement Web

Travaux dirigés 1

Publication de contenu Web, HTTP, GET, POST et CGI's

Exercice 1

Apache

- Sur chaque machine de l'IUT tourne un serveur Web Apache
- Connectez-vous sur `http://localhost` pour vous en assurer
 - Créez un répertoire `www` à la racine de votre HOME contenant une page Web simple de test.
 - Procédez à sa "publication" en vérifiant que tous les contenus de ce dossier son accessibles à tout le monde en lecture.
 - Attention, les répertoires doivent aussi être traversables ce qui inclus votre HOME. Les commandes à faire sont donc généralement : `chmod -R a+rX ~/www` et `chmod a+x ~`.
 - Ces commandes sont à faire une seule fois et dépendent de la configuration spécifique du serveur. Si les modules `suexec` et `suphp` d'Apache sont activés, les droits sont plus simples à configurer !
 - Attention : Les fichiers n'ont pas à être exécutables ! C'est une faille de sécurité que de rendre tout exécutable ou de tout mettre en écriture pour tout le monde. ne le faites pas !
 - Visualisez votre site dans un navigateur à l'adresse `http://localhost/~vous` où `vous` est votre nom de login.
 - Consultez les fichiers de configuration d'Apache dans `/etc/apache2/`.
 - En cas de souci avec Apache, placez-vous dans le dossier contenant votre code et lancez le serveur Web intégré de PHP :
`php -S localhost:8000` ou celui de Python :
`python3 -m http.server`

Exercice 2

Le protocole HTTP

- Utilisation du protocole HyperText Transfer Protocol. On va maintenant se connecter directement sur un serveur Web.
- Faire un telnet ou un netcat sur le port 80 d'une machine sur laquelle tourne un serveur web et essayez de communiquer avec le serveur par le protocole HTTP.
 - Essayez essentiellement GET /
 - Que constatez-vous ?
 - Que se passe-t-il si vous demandez une page qui n'existe pas (par exemple GET /toto) ?

Exercice 3

Outils de debugage Web

1. Ouvrez les Outils de Développement de votre navigateur
2. Ouvrez une page Web, puis dans l'onglet réseau observez les échanges entre le client et le serveur.
3. Inspirez-vous en pour faire un échange un peu plus riche "à la main" en HTTP avec un serveur Web.
4. Observez également les informations fournies par ces outils pour observer la structure d'une page Web (Arborescence DOM)

Code 1 – Hello World en CGI

```
#!/bin/bash
# -*- shell-script -*-

cat <<EOF
Content-type: text/plain

hello world!
EOF
```

Exercice 4

Scripts CGI

Nous allons maintenant explorer simplement la manière dont un script est lancé sur le serveur et comment ce script récupère les informations provenant du client.

1. Testez le script helloCGI.sh
2. Installez ce script sur votre serveur dans www/cgi-bin et testez. Adaptez les droits de ce script.
3. Puis pour tester, pointez votre navigateur sur <http://localhost:8000/cgi-bin/helloCGI.sh>

Exercice 5

Formulaire simple en CGI

1. Ecrivez un formulaire simple avec saisie d'un nom et d'un prénom dans des champs texte
2. Ecrivez un script bash qui récupérera les valeurs de ces champs et qui affichera les variables d'environnement concernant le client. Le script cgi doit se trouver dans www/cgi-bin et le formulaire dans www.
3. Testez la validation du formulaire par un POST ou un GET

Exercice 6

Un peu plus de CGI

Ajoutez au formulaire précédent un champ de saisie de la date de naissance de la personne et en réponse affichez le nom et le prénom de la personne ainsi que le calendrier du mois de naissance de la personne en utilisant la commande bash cal.

Code 2 – Trace de courbe avec GNUPLOT

```
set terminal svg
set xlabel "axe des x"
set ylabel "axe des y"
set output "dir/courbes.svg"
plot [-5:5] sin(x), cos(x)
```

Exercice 7

Tracer une courbe en CGI

A présent, tracez une courbe avec gnuplot au format svg et affichez la dans une page web. Ecrivez le script bash correspondant en vous aidant du script.sed fourni pour traiter les intervalles.