Services Web

TP n°1 : Préliminaires et Socket

1. Lire le cours en ligne et répondre aux questions en vert :

- a. IPv4 et IPv6 ?? Faire un test avec
telnet www.google.com 80 et
telnet www.univ-paris8.fr 80
Puis ifconfig (sous linux, ou ipconfig sous Windows)
- b. chercher des informations sur nntp et les assembler (sans faire un copié/collé de Wikipedia)
- c. Listez quelques well-known ports.
- d. Qu'est-ce que l'IANA ?
- e. Requêtes HTTP : tester quelques requêtes en utilisant telnet sur le port 80: POST, PUT, DELETE, TRACE, CONNECT. Chercher et expliquer ce qu'elles font en quelques lignes, chacune. (cf. pages 30 à 33 du cours)
- f. TSL (SSL) : donner une description complète
- g. API : détailler en cherchant suffisamment d'informations.

2. Création d'une socket grâce à PHP. Installation, si nécessaire, d'EasyPHP

- a. D'abord, il nous faut un serveur local, sur notre machine. On va utiliser EasyPHP (sous Windows).
Rappel (Wikipedia) : EasyPHP est un environnement comprenant deux serveurs (s. web Apache et s. BD MySQL), un interprète PHP, et une administration SQL phpMyAdmin. Il dispose d'une interface d'administration permettant de gérer les **alias** (dossiers virtuels disponibles sous Apache), et le démarrage/arrêt des serveurs. Il permet donc d'installer en une seule fois tout le nécessaire au développement local du PHP. Par défaut, le serveur Apache crée un **nom de domaine virtuel** (en local) 127.0.0.1. Ainsi, quand on choisit « Web local » dans le menu d'EasyPHP, le navigateur s'ouvre sur cette URL et affiche la page index.php de ce site qui correspond en fait au contenu du répertoire www d'EasyPHP. *EasyPHP peut être utilisé comme une application portable, c'est-à-dire lancé sur une clé USB.*
Il faut le **télécharger** sur easyphp.org et choisir le **DevServer**. Bien lire la documentation.
- b. Une fois fait, nous allons créer **une socket en PHP**. Voir le cours pages 20 à 22 et le site php.net pour davantage de détails : <http://fr.php.net/manual/fr/book.sockets.php>
- c. Il est préférable de séparer le code en 3 fichiers pour éviter les redondances (création en double de la socket, etc.) : fonction.inc.php qui sera appelé dans le fichier dédié au code du serveur et dans le fichier dédié au code du client. Il s'agit donc du code commun au deux. Puis le fichier serveur.php et le fichier client.php. Voir **ce site** pour un suivi pas à pas : http://www.solidtouch-fr.com/development/php_socket
Attention, ne pas oublier d'activer les sockets dans EasyPHP.