

TP 3 de Programmation Web - Master 1

Informatique : PHP

Auteur : Olivier GLÜCK, Université Lyon 1

Objectifs

- Installation et configuration de PHP
- Apprentissage de la programmation Web côté serveur en PHP
- Interfaçage avec une base de données

Pré-requis

Protocole HTTP, formulaires HTML, principes de la programmation web côté serveur, langage PHP

NB

Ce TP se déroule sur 4h30 (moitié séance 4 + séance 5). Il est impératif de travailler de façon importante entre chaque séance. Pour ce faire, il est nécessaire de conserver vos fichiers d'une séance sur l'autre.

1. Installation d'Apache

Dans ce TP, on vous demande d'installer et de configurer un serveur Apache sur votre poste de travail. Pour contacter votre serveur Web, vous utiliserez le navigateur Mozilla ou Netscape avec l'adresse `http://localhost/` ou `http://192.168.9.xx/`.

0- Pensez à faire la remise à zéro de la configuration des machines avant de commencer le TP...

1- Configurez `eth0` sur le réseau du miroir (192.168.9.9) en vous branchant sur le commutateur central ; vous prendrez l'adresse IP **192.168.9.xx** où **xx** est le dernier octet de l'adresse IP inscrite sur l'étiquette de votre machine. Vérifiez que vous arrivez à « ping » le miroir .

2- Installation des packages

Pour ce TP, les packages qui nous intéressent sont :

```
apache          1.3.28-2      Versatile, high-performance HTTP server
apache-common   1.3.28-2      Support files for all Apache webserver
apache-doc      1.3.31-1      Apache webserver docs
php4            4.3.4-4      A server-side, HTML-embedded scripting language
mysql-server    3.23.49-8.4   mysql database server binaries
php4-mysql      4.3.4-4      MySQL module for php4
```

Tapez les commandes suivantes pour installer apache ; vérifiez pour chaque commande qu'il n'y a pas de message d'erreur.

```
ping 192.168.9.9
export ftp_proxy=http://192.168.9.9:3128/
export http_proxy=http://192.168.9.9:3128/
rm -Rf /etc/apache /var/www;
apt-get update; apt-get remove --purge apache apache-doc apache-common;
apt-get install nom_pkg
```

3- Pour vérifier que le package est correctement installé, faire un `dpkg -l` et vérifiez que la ligne correspondant au package commence par `i`

Des fichiers (comme par exemple les logos permettant la construction de votre site) sont disponibles sur le serveur de la salle TP. Vous pouvez les récupérer en montant **192.168.9.9:/partage** par NFS (en lecture seule).

A la fin de la séance, n'oubliez pas de sauvegarder vos fichiers pour la séance suivante, soit en utilisant une disquette ou clé USB, soit en copiant vos fichiers dans /root et en utilisant restore.

2. Mise en place de PHP et premiers tests

Manipulation

Installez les packages permettant l'utilisation de apache, php4 et mysql (cf. 1). Configurez le serveur apache pour qu'il interprète correctement les fichiers ayant pour extension .php. Vous pourrez utiliser l'utilitaire apache-modconf pour permettre le chargement du module php4.

Manipulation

Créez un répertoire /root/PHP dans lequel vous placerez l'ensemble de vos fichiers pour ce TP. Utilisez la directive Alias pour que http://localhost/tpphp corresponde au répertoire /root/PHP. Créez, dans ce répertoire, un premier fichier phpinfo.php permettant d'afficher tous les paramètres de la configuration actuelle de PHP sur votre serveur. Vérifiez que tout fonctionne correctement.

Question et manipulation

Quelles sont les informations qui sont listées quand vous invoquez phpinfo.php ? Où se trouve le fichier de configuration de PHP ? Regardez le source de la page affichée dans votre navigateur après une requête vers phpinfo.php. Qu'en concluez-vous ?

Question et manipulation

Quel processus exécute le code PHP sur le serveur ? Combien de processus de ce type sont en cours d'exécution sur votre serveur ? Ajoutez une boucle infinie dans phpinfo.php et vérifiez avec top votre réponse à la première question. Au bout de combien de temps le processus arrête-t-il d'exécuter la boucle infinie ? Que se passe-t-il alors dans le navigateur ? Où cela se configure-t-il (donnez le nom de la directive correspondante) ?

3. Une petite application de gestion d'Images

On vous propose de réaliser une petite application de gestion et d'affichage d'images (ou photos). Cette application, principalement écrite en PHP, sera interfacée avec une base de données mysql nommée photos_base contenant deux tables :

- une table mes_photos contenant les informations relatives à chaque photo publiée sur votre site :
 - o nom_photo (nom relatif du fichier) de type VARCHAR(20) BINARY NOT NULL,
 - o categorie (catégorie à laquelle la photo appartient) de type VARCHAR(20) BINARY NOT NULL,
 - o titre (permet d'associer un titre à la photo) de type VARCHAR(50) BINARY,
 - o hauteur (hauteur en pixels de la photo) de type SMALLINT NOT NULL,
 - o largeur (largeur en pixels de la photo) de type SMALLINT NOT NULL,
 - o date (date de la prise photo) de type DATE DEFAULT '0000-00-00',
 - o liste_mots (mots-clés associés à la photo avec comme séparateur le caractère espace) de type VARCHAR(100)
- une table mes_categories contenant les informations relatives aux catégories d'appartenance des photos :
 - o categorie (nom de la catégorie) de type VARCHAR(20) BINARY NOT NULL,
 - o chemin (chemin absolu du répertoire contenant les photos de la catégorie) de type VARCHAR(50) BINARY NOT NULL,
 - o passwd (code d'accès aux photos de la catégorie) de type VARCHAR(20) BINARY DEFAULT NULL

L'application devra :

- proposer un formulaire à l'utilisateur lui permettant de choisir les photos qu'il souhaite voir afficher dans son navigateur en fonction soit d'une catégorie, soit d'une liste de mots-clés, soit de la date de prise de la photo, soit d'une combinaison de ces trois critères ;
- gérer l'authentification lorsque l'accès à une catégorie nécessite un mot de passe ;
- gérer l'affichage des photos demandées en utilisant éventuellement le titre et la date de la photo ;

Manipulation

Créez la base `photos_base` et les tables `mes_photos` et `mes_categories`. La clé primaire de la table `mes_photos` sera le couple (`nom_photo`, `categorie`) et celle de `mes_categories` sera (`chemin`). Il est très fortement conseillé de mettre les commandes SQL de création de chacune des tables dans un fichier `.sql` afin de ne pas avoir à tout retaper s'il faut recommencer...

Récupérez sur le serveur `192.168.9.9` le fichier `BD_TP_PHP.tgz` (par NFS ou ssh). Importez le fichier `photos.txt` dans la table `mes_photos` et ajoutez les catégories correspondantes dans la table `mes_categories`. Pour l'importation de `photos.txt`, utilisez la commande suivante :

```
mysql> LOAD DATA LOCAL INFILE "photos.txt" INTO TABLE mes_photos;
```

Créez un utilisateur `php` de `mysql` qui vous permettra de vous connecter à la base avec tous les droits ; pour cela, vous taperez les commandes suivantes :

```
bash$ mysql mysql
```

```
mysql> GRANT ALL PRIVILEGES ON *.* TO php@localhost IDENTIFIED BY 'mot_de_passe' WITH GRANT OPTION;
```

Manipulation

Dans un fichier, nommé `photos.php`, créez un formulaire contenant deux boutons de soumission portant le même nom mais une valeur distincte. La soumission du formulaire doit provoquer l'exécution de `photos.php` (vous pourrez utiliser `$_SERVER['PHP_SELF']`). Dans ce script, écrivez une fonction `aff_table()` permettant d'afficher le contenu d'une table et prenant en paramètres le nom d'une base, le nom d'une table, le nom d'hôte hébergeant la base de données, le nom de connexion et le mot de passe permettant l'accès à la base. Complétez le script pour que les deux boutons du formulaire permettent chacun l'affichage d'une table (resp. `mes_photos` et `mes_categories`). Les affichages liés à une soumission du formulaire devront se faire en-dessous de celui-ci.

Manipulation

Ajoutez au formulaire précédent :

- un menu déroulant contenant la liste des noms de catégories ; ce menu devra être construit après une interrogation de la table `mes_categories` ;
- un bouton de soumission (portant toujours le même nom mais avec comme valeur « Voir les photos »)

Manipulation

Modifiez le script pour qu'un click sur « Voir les photos » (après la sélection d'une catégorie) affiche l'ensemble des images de la catégorie sélectionnée. Pour l'instant, ne tenez pas compte du mot de passe éventuel d'accès à la catégorie. Vous afficherez les photos dans un tableau sur 4 colonnes en renseignant correctement les attributs de la balise `` (`src`, `alt`, `width`, `height`) ; l'attribut `alt` devra contenir le titre de la photo ainsi que sa date. Testez votre script en utilisant la base de `photos` qui vous a été fournie. Voici un aperçu de ce que vous devez obtenir :

Formulaire d'affichage d'images

Voir table photos

Voir table categories

Sélectionnez une catégorie

Animaux
Bouffe
Famille
Monuments

Voir les photos

Voici les photos que vous avez sélectionnées !

Il y a 4 photos dans la catégorie Animaux (BD_TP_PHP/Animaux, toto1) :



BD_TP_PHP/Animaux/DSC00011.jpg "Perroquet" 2004-06-06 h=200 l=150 "perroquet oiseau pirate"
BD_TP_PHP/Animaux/DSC01212.jpg "Chien" 2003-06-06 h=200 l=150 "chien lydie"
BD_TP_PHP/Animaux/DSC01422.jpg "Canard" 2004-06-06 h=200 l=150 "canard versailles"
BD_TP_PHP/Animaux/DSC01446.jpg "Chevre" 2004-06-06 h=200 l=150 "chevre versailles"

Manipulation

Ajoutez maintenant la gestion de l'authentification si un code d'accès à la catégorie est nécessaire.

Manipulation

Utilisez une session PHP pour conserver les mots de passe des catégories déjà accédées et ainsi éviter à l'utilisateur de ressaisir systématiquement le mot de passe. S'il vous reste du temps, vous pouvez améliorer votre application en enrichissant le formulaire pour que le choix des photos puisse également se paramétrer en fonction de la date de prise de la photo, des mots-clés associés aux photos, ... Vous pourrez ajouter un bouton permettant de voir la liste des mots-clés disponibles (pour faire du « exact_match »). Vous pourrez également proposer une interface permettant d'ajouter des photos dans la table mes_photos... Donnez libre court à votre imagination.