TP PHP

Olivier AUBERT

27 mars 2002

1 Configuration d'apache

Installez apache à partir des paquetages *Debian* disponibles sur le serveur `134.214.90.7`. Installez de plus les paquetages `php4`, `php4-gd` et `php4-mysql`.

La documentation de php4 est installée également sur le serveur, à l'adresse `http://134.214.90.7/php/`.

Pour activer php sous apache, il faut charger le module correspondant, et ajouter un *handler* permettant de traiter particulièrement les fichiers se terminant par l'extension `.php`.

2 Configuration de php

Le fichier de configuration de php, `/etc/php/apache/php.ini`, est lu par PHP au démarrage. On peut y fixer un certain nombre de paramètres contrôlant le comportement de l'interpréteur php.

3 Premier fichier php

On va tout d'abord tester le bon fonctionnement du module php. Pour cela, créez dans la racine du serveur web un fichier avec l'extension `.php` contenant du code HTML simple, où vous insérerez le code suivant :

```
Vous utilisez <?php echo $HTTP_USER_AGENT; ?>.
```

Les variables d'environnement sont donc accessibles directement en tant que variables de php. Vérifiez que le fichier que vous avez créé présente bien le fonctionnement attendu. En cas de comportement bizarre, pensez à aller consulter les fichiers d'historique `error.log` et `access.log`.

De manière analogue au module `server-info` d'apache, il est possible de demander à php d'afficher des informations sur sa configuration. Trouvez la documentation de la fonction `phpinfo()` et mettez-là en œuvre pour afficher ces informations.

4 Affichage de la date

Nous allons maintenant faire évoluer ce fichier en faisant afficher la date courante du système. Trouvez dans la documentation la syntaxe de la fonction `getdate()` et utilisez la pour afficher la date en français. Il faudra au préalable créer deux listes contenant les noms de jours et les noms de mois en français.

5 Gestion d'un formulaire

Créez un formulaire HTML contenant les champs suivants : `couleur` (qui sera une liste de noms de couleurs), `hauteur` et `largeur` (qui seront deux entiers). Dans un premier temps, le traitement de ce formulaire consistera uniquement à afficher sous le formulaire original un message du type `Vous avez demandé un rectangle de 200 par 100 en rouge`. Écrivez le code php permettant de traiter les données reçues du formulaire et d'afficher le message désiré.

6 Création dynamique d'image

Nous allons maintenant générer l'image aux caractéristiques demandées par l'utilisateur. Pour cela, il faudra utiliser les fonctions de la bibliothèque de traitement d'image GD installée avec php. Trouvez dans la documentation la section `IV.XLI` relative aux fonctions de manipulation d'image. Vous y trouverez les références des fonctions permettant de créer une nouvelle image et de la manipuler.

Créez un nouveau fichier `image.php` qui prendra comme paramètres les valeurs de couleur, hauteur et largeur et générera l'image correspondante. Ce fichier `image.php` indiquera par la commande

```
header ("Content-type: image/png");
```

le type des données qu'il renvoie.

Au niveau du traitement du formulaire, il faudra insérer dans la réponse une balise `<img src...>` appelant le nouveau fichier créé.

7 Authentification

Pour effectuer l'authentification d'un utilisateur, il faut soit configurer cette authentification au niveau d'apache via le fichier `.htaccess`, soit effectuer cette authentification par php si l'on désire effectuer un traitement particulier. Pour cela, on utilise le code suivant :

```
<?php
  if(!isset($PHP_AUTH_USER)) {
    header("WWW-Authenticate: Basic realm=\"My Realm\"");
    header("HTTP/1.0 401 Unauthorized");
    echo "Texte à envoyer si le client appuie sur le bouton d'annulation\n";
    exit;
  } else {
    echo "Bonjour $PHP_AUTH_USER.<P>"
    echo "Vous avez entré le mot de passe $PHP_AUTH_PW.<P>"
  }
?>
```

Testez l'authentification par les deux moyens. Trouvez pour la première manière les variables par lesquelles l'identité de l'utilisateur est accessible et affichez-les.

8 Utilisation des sessions (php4)

php4 définit la notion de session, qui va permettre de conserver les valeurs de variables sur différents documents du site. Pour cela, il faut indiquer quelles sont les variables dont la valeur doit être conservée via la fonction `session_register ("nom-de-variable");`. Ensuite, il faut indiquer `session_start();` dans chaque document php devant correspondre à la session. Faites un test d'affichage de la valeur d'une variable dans deux document php différents, sans utiliser la notion de session tout d'abord afin de voir que les valeurs ne sont pas transmises.

Activez ensuite la gestion des session, et répétez l'opération. Vous pouvez identifier les mécanismes utilisés pour cela en regardant les cookies déclarés sur votre navigateur ainsi que le répertoire indiqué dans le fichier `php.ini` par le nom `session.save_path`.