

Département Informatique
R2.05 – TP Apache/PHP/MySQL/CMS
Responsable : X.Roirand

Durée : 90mn machine

Le but de ce TP va être d'installer un ensemble constitué d'un serveur web (apache), d'un langage de programmation (PHP), d'une base de donnée mysql et d'un CMS (Joomla).

Introduction :

Un serveur web est un logiciel permettant à des clients d'accéder à des pages web, c'est-à-dire en réalité des fichiers au format HTML à partir d'un navigateur (aussi appelé browser) installé sur leur ordinateur distant.

Un serveur web est donc un « simple » logiciel capable d'interpréter les requêtes HTTP arrivant sur le port associé au protocole HTTP (par défaut le port 80), et de fournir une réponse avec ce même protocole.

Les principaux serveurs web sur le marché sont entre autres :

- Apache
- Nginx
- Microsoft IIS (Internet Information Server)

Apache :

Apache (www.apache.org) est le serveur le plus répandu sur Internet. Il s'agit d'une application fonctionnant à la base sur les systèmes d'exploitation de type Unix, mais il a désormais été porté sur de nombreux systèmes, dont Microsoft Windows.

Apache (prononcez à la française ou bien pour les puristes à l'anglophone « Apatchy ») tire son nom de la façon dont il a été mis au point (« A patchy server » traduisez « un serveur rafistolé ») car il est le fruit d'une multitude de correctifs logiciels afin d'en faire une solution très sûre. En effet Apache est considéré comme sûr dans la mesure où peu de vulnérabilités le concernant sont publiées.

Ainsi, dès qu'un bug ou une faille de sécurité est décelée, ceux-ci sont rapidement corrigés et une nouvelle version de l'application est éditée.

Apache possède désormais de nombreuses fonctionnalités dont la possibilité de définir une configuration spécifique à chaque fichier ou répertoire partagé, ainsi que de définir des restrictions d'accès grâce aux fichiers htaccess.

Document de rendu :

Créez un document qui va contenir les réponses que vous allez donner au fur et à mesure de l'avancement dans le TP (ce document sera à rendre dans la zone de rendu « TP APACHE/PHP/MYSQL/CMS » sur Moodle dans l'UE R 2.05).

Toutes les questions auxquelles vous devez répondre dans le document de rendu sont de la forme :

Question x) cherchez comment ...

Pour ce TP, vous allez vous servir d'une seule machine qui fera tourner un serveur Apache, du PHP, une base de données MySQL et un CMS.

Connexion sur la première machine distante

Vous aller commencer par vous connecter sur une machine distante appelée « Sandbox », qui est un point d'entrée sur l'ensemble des machines du réseau mis à disposition des étudiants pour ces TP. Une fois connecté sur cette sandbox, vous pourrez ensuite vous connectez sur votre machine finale sur laquelle faire le TP. Cette double étape sera à répéter à chaque TP, pensez-donc bien à la noter, afin de gagner du temps pour les prochains TP.

Pour se connecter sur la sandbox, vous avez besoin d'un client ssh (voir début de ce TP) d'une adresse ip (celle de la sandbox), d'un utilisateur et d'un mot de passe, voici ces informations :

Ip : 198.27.82.182
User : student
Mdp : ?Student_56

Une fois sur cette sandbox, connectez-vous sur votre machine distante finale, pour cela vous aller utiliser un client ssh et l'adresse ip de votre machine finale.

Pour se connecter en ssh sur votre machine depuis la sandbox, il faut connaître le numéro que le professeur vous a donné, et utiliser ssh comme suit :

ssh 192.168.0.X ou X est votre numéro

Exemple : si votre numéro est le 23 alors il faut faire :

ssh 192.168.0.23

Ensuite le login et mot de passe sur votre machine distante sont :

login : student
mot de passe : ?<votre numéro>Student_56

Exemple : si votre numéro est le 13 alors il faut faire un ssh sur la machine 192.168.0.13 avec les informations suivantes :

login : student
mot de passe : ?13Student_56

Une fois sur votre machine distante finale, vous pouvez l'administrer totalement car le user student sur cette machine peut faire des commandes sudo.

ATTENTION : la 1^{ère} fois que vous vous connectez sur votre machine finale, changez votre mot de passe de l'utilisateur student. Si vous ne le faites pas, toute personne connaissant votre numéro pourrait se connecter en tant que student sur votre machine finale et en faire ce qu'elle veut !!!

1) Installation du serveur Apache2 sur votre machine distante

Installer le serveur apache2 sur votre machine.

Question 1) indiquez ce que vous avez fait pour trouver le bon package apache2 à installer et la commande pour l'installer.

Tester que le serveur apache fonctionne. Pour cela, utiliser telnet en local ou depuis la sandbox.

Une fois connecté, tapez :

GET / HTTP/1.0 puis <entrée> et encore <entrée>

Il doit vous renvoyer une page HTML du genre:

```
HTTP/1.0 200 OK
Date: Mon, 06 Oct 2014 21:48:07 GMT
Server: Apache/2.2.22 (Debian)
[...]  
</body></html>
```

Question 2) copiez/collez la réponse de votre commande "GET / ..." donc la page HTML qui vous a été renvoyé par le serveur Apache2.

Tester ensuite avec un navigateur sur votre PC ou le PC de l'IUT, avec l'adresse <http://198.27.82.182:180XX/> ou XX est le numéro qui vous a été donné par le professeur. Si votre numéro est inférieur à 10 alors il faut le mettre sous la forme 0X.

Vous devez normalement retrouver la page HTML d'Apache2.

Question 3) faites un screenshot de votre navigateur lorsque vous avez chargé la page <http://198.27.82.182:180XX>.

Question 4) trouvez et indiquez (en vous aidant d'internet) l'endroit où est stocké le fichier index.html qui correspond à ce qui a été renvoyé lors des commandes précédentes. Ce répertoire est nommé le root directory du serveur web.

Nous allons maintenant découvrir PHP (<https://apprendre-php.com/tutoriels/tutoriel-19-introduction-au-langage-php.html>):

PHP a été inventé à l'origine pour le développement d'applications web dynamiques qui constituent encore le cas d'utilisation le plus courant et son point fort. Cependant, les évolutions qui lui ont été apportées jusqu'à aujourd'hui assurent à PHP une polyvalence non négligeable. PHP est par exemple capable d'interagir avec Java, de générer des fichiers PDF, d'exécuter des commandes Shell, de gérer des objets (au sens programmation orientée objet), de créer des images ou bien de fournir des interfaces graphiques au moyen de PHP GTK.

Dans cette présentation du langage, nous introduirons tout d'abord les caractéristiques de PHP, puis nous verrons en quoi il est particulièrement adapté aux développements

d'applications web. Nous synthétiserons ensuite les autres types d'applications possibles avec PHP avant de terminer sur les limites que l'on peut lui reprocher.

Les caractéristiques du langage PHP

Licence

PHP est tout d'abord un langage de script interprété (en réalité précompilé en Opcode), gratuit, OpenSource et distribué sous une licence autorisant la modification et la redistribution.

Portabilité

PHP est supporté sur plusieurs systèmes d'exploitation. C'est par exemple le cas des versions Microsoft Windows™, mais aussi des systèmes reposant sur une base UNIX (Apple MAC OS X™, distributions Linux ou encore Sun Solaris).

Il sera alors très facile de déplacer une application écrite en PHP d'un serveur Windows d'origine vers un serveur Linux sans avoir à la modifier (ou très peu).

Exécution

D'un point de vue exécution, PHP a besoin d'un serveur Web pour fonctionner. Toutes les pages demandées par un client seront construites par le serveur Web, en fonction des paramètres transmis, avant d'être retournées au client. Le schéma ci-dessous illustre le principe de fonctionnement de PHP.



Note : il aurait été possible d'ajouter un serveur de bases de données (local ou distant) sur cette illustration. Nous aurions eu alors deux étapes supplémentaires qui sont l'interrogation de la base de données par PHP et la récupération des résultats en provenance du serveur SQL.

Apprentissage de PHP

PHP est un langage dit de « haut niveau » dont la syntaxe est très proche du langage C. Cette syntaxe proche du langage naturel lui assure un apprentissage rapide et peu de contraintes d'écriture de code. Néanmoins, la maîtrise rapide de sa syntaxe ne signifie pas la maîtrise de ses fonctionnalités et de ses concepts. Une bonne connaissance et une utilisation avancée de la programmation PHP nécessite un temps d'apprentissage relativement long.

Richesse du langage PHP

Une des forces du langage PHP est sa richesse en terme de fonctionnalités. En effet, il dispose à l'origine de plus de 3 000 fonctions natives prêtes à l'emploi garantissant aux développeurs de s'affranchir de temps de développement supplémentaires et parfois fastidieux. Ces fonctions permettent entre autre de traiter les chaînes de caractères, d'opérer mathématiquement sur des nombres, de convertir des dates, de se connecter à un système de base de données, de manipuler des fichiers présents sur le serveur...

PHP puise aussi sa richesse dans le dynamisme de sa communauté de développeurs. Celle-ci était estimée à 500 000 personnes environ en 2003 mais il est très probable qu'elle ait dépassé le million maintenant. Les profils de développeurs de la communauté sont très divers. Il y'a ceux qui apportent de nouvelles fonctionnalités et librairies de version en version, ceux qui traduisent la documentation en plusieurs langues ou encore les programmeurs ayant des compétences plus modestes qui réalisent des applications Opensources prêtes à l'emploi. Parmi les plus connues, nous pouvons citer les CMS (Joomla, SPIP, Dotclear, Wordpress...), les systèmes de ventes en ligne (OSCommerce), les forums (PHPBB, IPB, VBulletin), les frameworks (Zend Framework, Symfony, CakePHP, Jelix)...

PHP, un langage fiable et performant

Le langage est maintenant devenu un langage fiable, performant et viable. Il est capable de supporter des sites qui sollicitent des millions de requêtes journalières. De nombreuses entreprises de renommée nationale et internationale lui font confiance pour le développement de leur site Internet. Nous pouvons parmi elles citer TF1, IBM, Le Monde, Le Figaro, Club-Internet, Orange, Pages Jaunes... Un récent rapport daté de novembre 2006 indique que 87% des entreprises du CAC40 utilisent PHP. [Lire l'interview de Perrick Penet \(AFUP\).](#)

PHP, un langage conçu pour les applications web dynamiques

Le langage PHP a la principale fonction d'être spécialement conçu pour la réalisation d'applications web dynamiques. Par définition, une « application (ou page) dynamique » est un programme capable de générer une page unique en fonction de paramètres qui lui sont transmis.

Un script PHP peut donc être intégré directement à l'intérieur d'un code html.

Installez maintenant PHP sur votre machine distante.

Question 4) indiquez la version de PHP qui a été installé sur votre machine.

Vous allez tester que votre serveur web Apache2 est bien capable de travailler avec du PHP, pour cela, créez une page test.php dans le root directory du serveur web et dans laquelle vous écrivez :

```
<?php  
phpinfo();  
?>
```

Testez que le chargement de la page test.php affiche bien le résultat de phpinfo() dans votre navigateur, ça doit donner un truc du genre:

Question 5) Copiez/Collez le contenu de la page renvoyée.

Question 6) Pourquoi la page ne renvoie pas ce qu'on attend ? Cherchez sur internet et donnez comme réponse l'explication qui explique qu'on voit un problème.

Question 7) Expliquez comment corriger le problème et faire que la page test.php renvoie bien ce qu'on attend.

Question 8) Une fois le problème corrigé, faites un screenshot de la page du navigateur avec le bon contenu affiché de test.php.

MySQL :

MySQL est ce qu'on appelle un système de gestion de bases de données. De manière très schématique, c'est un programme qui va nous permettre de manipuler simplement nos bases de données.

En effet, les bases de données sont des systèmes très complexes. Nous utilisons un système de gestion de bases de données pour cacher cette complexité et effectuer simplement les opérations dont nous avons besoin sur nos bases de données.

MySQL est loin d'être le seul système de gestion de bases de données ; il en existe bien d'autres. Parmi les plus connus, on peut ici notamment citer SQL Server, Postgresql, MS Access ou encore Oracle. Chacun de ces systèmes de gestion de bases de données fonctionne de manière similaire (ils permettent d'envoyer des instructions SQL) et propose des fonctionnalités relativement équivalentes.

Dans ce TP, nous utiliserons MySQL pour des raisons simples : il est gratuit, simple d'utilisation, utilise du SQL standard et le PHP supporte son usage.

Installez MySQL (client et serveur)

ATTENTION: notez bien le mot de passe administrateur demandé à l'installation, car il vous sera utile ensuite.

Question 9) indiquez les commandes que vous avez tapé pour installer MySQL serveur et client.

Question 10) indiquez le ou les commandes que vous avez tapé pour vérifier que le serveur MySQL et le client MySQL ont bien été installés et sont fonctionnel.

Nous allons maintenant installer phpmyadmin pour pouvoir facilement utiliser MySQL, quelques mots sur phpmyadmin;

Le projet phpMyAdmin est l'un des projets les plus populaires afin de manipuler efficacement la base de données MySQL en ligne. Il est souvent inclus avec divers projets comme XAMPP par exemple et chez de nombreux hébergeurs de site Web. Pour quelle raison ? Il permet d'explorer le contenu des bases de données, des tables, de connaître leurs structures ainsi que d'exécuter des requêtes SQL. Enfin, il offre aussi la possibilité la possibilité d'effectuer la recherche de données dans l'ensemble d'une base de données.

Installez phpmyadmin sur votre machine distante.

Question 11) comment testez vous que phpmyadmin fonctionne ?.

Faites ce qu'il faut pour que le message d'erreur ne soit plus affiché.

Est-ce que la page phpmyadmin fonctionne maintenant ?

Question 12) indiquez ce que vous avez fait pour faire fonctionner phpmyadmin du coup ?

Question 13) installer maintenant Wordpress en téléchargeant le tarball de la page <https://wordpress.org/download/> et faites le fonctionner ?