

Votre serveur PHP à domicile

annexe



Configuration PHP en local

Il est bien sûr possible de tester PHP chez soit, sans se préoccuper d'Internet. Dans ce cas, rendez-vous directement à la section 2.

Avec la démocratisation des accès haut débit, câble ou ADSL, on peut être tenté d'héberger directement son application web à domicile, sur son ordinateur personnel. Si cette solution vous tente, voici quelques éléments clés avant de démarrer.

Avantages et inconvénients

Avant toute chose il est important de bien peser le pour et le contre d'un hébergement à domicile. Transformer son salon en salle machine n'est évidemment pas sans conséquence.

Cette boutade met en avant un premier point, souvent négligé. Un serveur web, avec ou sans PHP, doit rester opérationnel 24 heures sur 24. Cela implique que l'ordinateur qui sera utilisé pour héberger l'application devra rester allumé en permanence.

Cet état de fait a comme première conséquence notable une consommation électrique non négligeable. Songez que le seul processeur peut consommer jusqu'à 100 W ! De plus le corollaire de cette forte consommation est la chaleur et par voie de conséquence l'importante nécessité de refroidissement de l'ordinateur. Quiconque s'est approché d'un micro-ordinateur aura pu constater que le bruit est une réelle nuisance. Naturellement, il est possible d'améliorer les choses, mais dans la plupart des cas il faudra apprendre à vivre avec ce doux ronronnement.

Si ces nuisances ne constituent pas vraiment un point en faveur de cette solution, elles mettent en avant l'un de ses avantages clés : le serveur est à portée de main pour réaliser toutes les opérations de mise à jour et de modification. Cette proximité facilite énormément le travail tout en procurant une impression de contrôle et de sécurité certaine.

Cette latitude en termes de maintenance se retrouve évidemment au niveau de la configuration logicielle. Il devient possible de construire des environnements logiciels bien plus complexes ou moins « tendances » que ceux disponibles en standard chez les hébergeurs de services mutualisés.

Compte tenu de la jeunesse de PHP 5, l'utilisation d'un ordinateur personnel peut notamment permettre de disposer d'un hébergement sans attendre la migration nécessairement progressive des hébergeurs.

On le voit, la liberté est totale, mais comme souvent, elle s'accompagne d'une responsabilité élargie, notamment en termes de sécurité. Il vous appartiendra d'assurer les mises à jour, la sécurisation quotidienne de votre serveur, en appliquant les patchs de sécurité par exemple. Ce travail quotidien peut être rapidement fastidieux, d'autant qu'il ne constitue pas l'objectif premier de l'installation.

Enfin, dernier point, à ce jour les liaisons ADSL et câble sont asymétriques, c'est-à-dire que (dans le cas des solutions grand public) le débit en réception est sensiblement plus important que le débit en émission. Ce détail technique n'a que peu d'importance pour un usage classique. En effet, nous téléchargeons tous bien plus de données que nous n'en émettons.

Cependant, les possibilités s'inversent quand on se place du point de vue du visiteur d'un site hébergé à domicile. En effet, celui-ci est limité par la capacité d'émission, le plus souvent de 256 Kbit/s, alors qu'en téléchargement la capacité dépasse souvent 2 Mbit/s, soit 10 fois plus !

Si la mise à disposition de fichiers était dans vos intentions, sachez que cela risque d'être très laborieux, pour vos utilisateurs mais aussi pour vous, car le moindre téléchargement saturera votre capacité d'émission, et même les quelques éléments émis pour obtenir une page web seront paralysés, sauf action adéquate. Votre propre consommation Internet risque donc de souffrir.

Malgré l'ensemble des contraintes qui peuvent de prime abord paraître conséquentes, l'amélioration constante de la qualité des connexions haut débit rend réellement viable l'hébergement d'un service classique à domicile, pour peu que les téléchargements soit déportés sur un autre site.

Adresse IP et nom

Avant même de vous lancer à la conquête de PHP 5 et du Web, il faudra vous préoccuper des problèmes d'adressage et de nom.

Principe

En règle générale, pour accéder à un serveur, nous faisons appel à son nom : cela nous paraît logique et naturellement le plus adapté. Cependant, au sein du réseau, les serveurs ne sont pas désignés par ce nom, mais par une adresse numérique plus simple à manipuler. Cette adresse est utilisée pour trouver la route entre un serveur et votre propre ordinateur. Pour assurer la correspondance entre les deux, le réseau Internet dispose donc d'un système d'annuaire, le DNS (*Domain Name Server*).

Jusque-là tout est simple. Hélas, cette mécanique est lourde et suppose que la correspondance entre un nom et son adresse ne change que très rarement. Or, avec l'explosion du nombre d'Internaute, les besoins en adresses sont de plus en plus difficiles à satisfaire (celles-ci sont en effet composées de quatre nombres compris entre 0 et 255, le nombre d'adresses disponibles est donc limité). Pour résoudre ce problème, les fournisseurs d'accès mutualisent leur pool d'adresses entre tous leurs clients. Après tout, nous ne nous connectons pas tous au même instant.

Cette pratique ne pose aucun problème en soit, notamment pour les connexions par modem, cependant un service, pour être contacté doit pouvoir être localisable de manière fiable à chaque instant. Le changement périodique d'adresse imposé par le fournisseur d'accès est donc problématique.

Naturellement, de plus en plus de fournisseurs commencent à proposer une adresse IP fixe. Néanmoins, cette option reste encore minoritaire. Que faire alors si vous n'êtes pas l'heureux possesseur d'une adresse IP fixe ?

Il existe par chance des services simples qui proposent un accès à leur annuaire (et donc à un nom) et un outil pour effectuer automatiquement les mises à jour en cas de changement. Ces annuaires, configurés pour ce type de fonctionnement sauront donc à chaque instant vous retrouver, vous et votre adresse IP.

Selon que vous soyez sous Linux ou Windows (il existe aussi des outils équivalents sur Mac) différents outils permettent cette mise à jour dynamique. La plupart du temps, ceux-ci peuvent se connecter à plusieurs services d'annuaire. N'hésitez donc pas à explorer les possibilités de chacun. Dans tous les cas, les logiciels de mise à jour doivent être lancés dès le démarrage et rester actifs.

SERVICE DynDNS

Les services d'annuaire dynamique sont légions, le plus connu est DynDNS à tel point que ce terme est devenu générique et désigne globalement un service d'annuaire dynamique.

Certaines sociétés proposent une prestation plus complète et vous permettront d'obtenir un nom plus séduisant, mais le principe reste le même.

▶ www.dyndns.org

Microsoft Windows

Bali DynDNS est un logiciel simple et efficace, il est disponible en français et supporte plusieurs services parmi les plus connus.

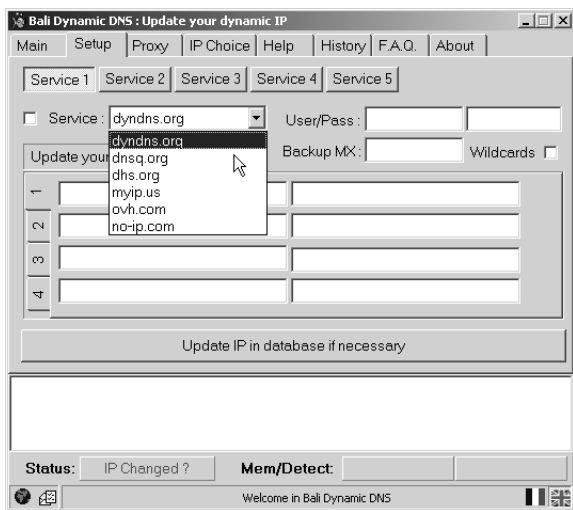


Figure A-1 Fenêtre de configuration de Bali DynamicDNS

Plusieurs services peuvent être utilisés simultanément mais la plupart du temps un seul sera utilisé. Lorsque le logiciel est lancé une icône est accessible dans la barre des tâches.

► <http://www.baliciel.com/software/baliddns.htm>

Linux

Sous Linux, le logiciel ez-ipupdate est particulièrement efficace. Il peut être lancé simplement en arrière-plan et gère tout seul. Le fichier de configuration est simple et le logiciel livré avec des exemples.

```
service-type=easydns
user=XXXX
host=mon-nom-a-moi.com
interface=eth1
wildcard=on
cache-file=/tmp/ez-ipupdate.cache
# uncomment this once you have everything working how
# you want and you are
# ready to have ez-ipupdate running in the background
# all the time. to stop it
# you can use "killall -QUIT ez-ipupdate" under linux.
daemon
```

Il est aussi possible de ne pas lancer ez-update sous forme de démon et le coupler aux scripts appelés par le client DHCP directement lorsque l'adresse est attribuée. La procédure dépend alors de votre client.

► <http://ez-ipupdate.com/>

Installation proprement dite

Une fois votre machine correctement accessible par tous, encore faut-il avoir un serveur web installé, deux installations types sont proposées, sous Microsoft Windows et sous Linux.

Sous Microsoft Windows

Installer Apache

L'installation d'Apache sous Windows est réellement un jeu d'enfant. Un installateur est disponible et l'installation se résume à une séquence de boîtes de dialogue élémentaires.

► <http://httpd.apache.org>



Figure A-2
Démarrage de l'installation d'Apache sous Microsoft Windows



Figure A-3
Fin de l'installation d'Apache et sélections des options

En phase de test, le plus simple est cependant de ne pas utiliser la possibilité offerte d'intégrer Apache en tant que service et de se contenter de démarrer celui-ci à la main.

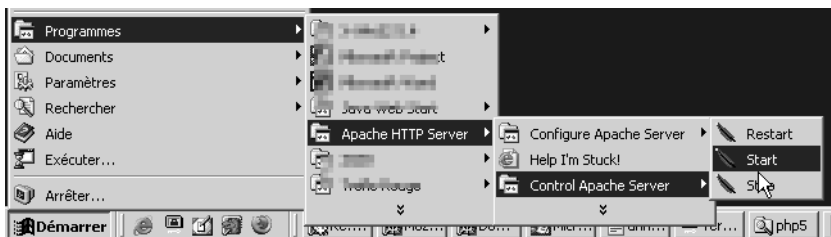


Figure A-4
Lancement d'Apache depuis le menu Démarrer

À RETENIR Localhost

On a vu précédemment que pour pouvoir nommer une machine il faut un annuaire. Toutefois, il existe un nom, disponible par défaut et qui désigne la machine locale, que celle-ci soit connectée au réseau ou non. Ce nom est `localhost`, il est utilisé par tous les systèmes d'exploitation.

L'installation réalisée, il est possible de vérifier immédiatement le bon fonctionnement du serveur avec n'importe quel navigateur. Une page par défaut apparaît alors. La racine du site web est positionnée sur le répertoire `c:\Program files\Apache Group\Apache\htdocs`.

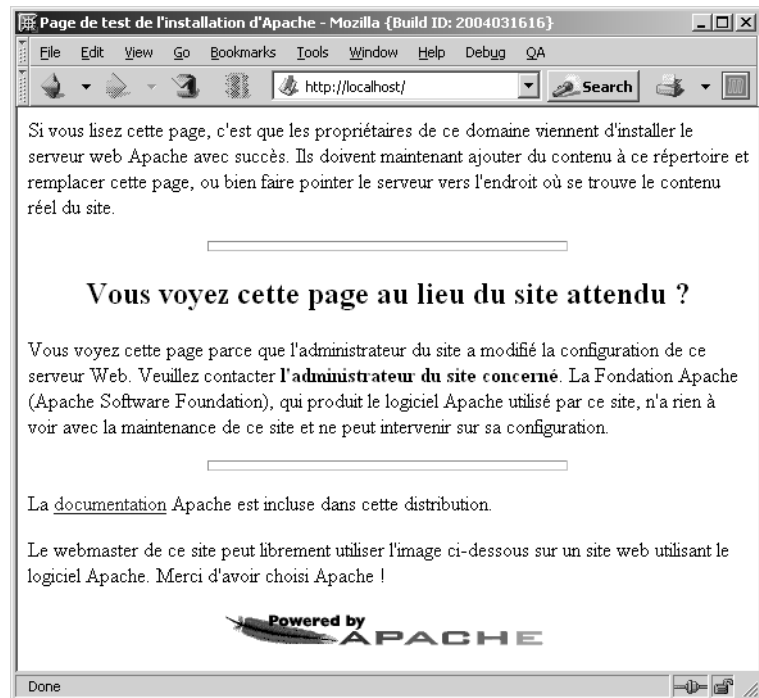


Figure A-5 Test d'Apache sous Microsoft Windows

Installer PHP 5

PHP dispose lui aussi d'un installateur. Celui-ci n'est pas disponible pour PHP 5, mais surtout, l'installation réalisée n'est pas satisfaisante. Il est préférable de télécharger le fichier `.zip` comprenant l'intégralité des extensions, le programme PHP en ligne de commande et les modules PHP.

La procédure consiste à décompresser cette archive dans un répertoire. Contrairement à certaines pratiques décrites sur Internet, il est vraiment déconseillé d'utiliser la racine de votre disque. Le répertoire d'installation d'Apache est beaucoup plus indiqué. Il suffit donc de créer un sous-répertoire `php` à l'intérieur puis de tout décompresser.

Là encore on évitera de suivre les modes d'emploi fantaisistes parfois proposés. Aucune des bibliothèques PHP n'est à copier dans le répertoire système de votre ordinateur. Seul le fichier `php.ini-recommended` doit être copié dans ce répertoire (sous Windows 2000 et XP, il s'agit de `c:\winnt`) et renommé `php.ini`.

► <http://www.php.net>

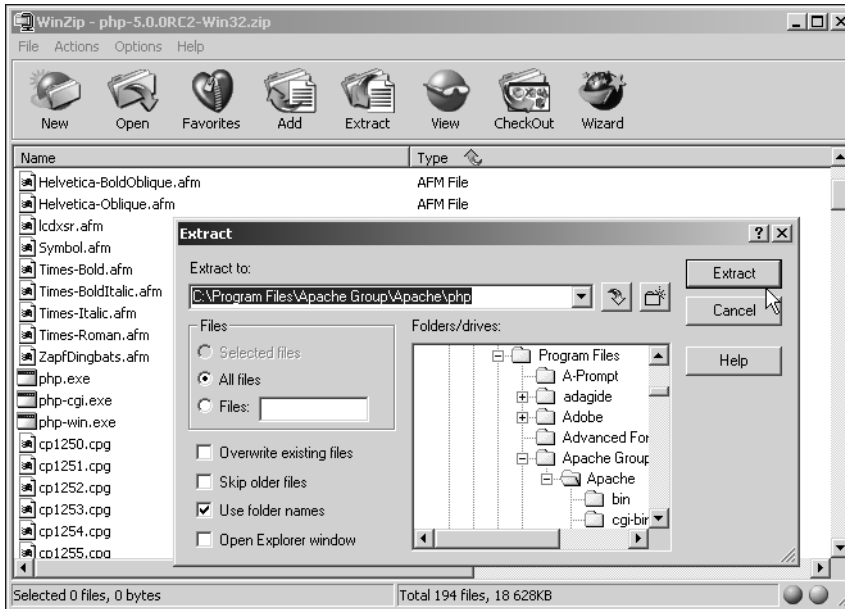


Figure A-6
Extraction des binaires PHP
pour Microsoft Windows

Avant de tester notre installation, quelques modifications doivent être apportées, d'une part au fichier de configuration Apache et d'autre part au fichier de configuration de PHP.

Dans le premier cas, il s'agit de s'assurer qu'Apache saura utiliser PHP. Voici les quatre modifications clés à apporter au fichier `httpd.conf` (celui-ci est situé dans le répertoire `C:\Program Files\Apache Group\Apache\conf` et peut être modifié et testé très simplement en utilisant les raccourcis du menu Démarrer) :

- 1 Ajouter la ligne suivante à la liste des modules :

```
LoadModule php5_module php/php5apache.d11
```

- 2 De même pour :

```
AddModule mod_php5.c
```

- 3 Déclarer l'interpréteur PHP pour les fichiers avec l'extension PHP dans la section `<IfModule mod_mime.c>`

```
AddType application/x-httpd-php .php
```

- 4 Ajouter `index.php` comme alternative dans la section `<IfModule mod_dir.c>`

```
DirectoryIndex index.html index.php
```

Enfin pour `php.ini`, il s'agit de préciser le répertoire où sont situées les extensions. Dans PHP Saloon, nous utiliserons `gd`, `sqlite`, `dom` et `xml` entre autres. Pour simplifier, nous allons demander le chargement automatique de celles-ci, ce qui nous évitera d'utiliser `d1()` dans le code de PHP Saloon. Pour ce faire, il suffit de modifier le chemin des extensions :

```
extension_dir = "c:/program files/apache group/apache/php/ext"
```

Puis de lister les extensions à charger par défaut :

```
extension = php_gd2.dll
extension = php_xml.dll
```

DOM et SQLite sont chargés d'emblée, il suffit donc d'ajouter GD et les transformations XSLT. Ces quelques opérations réalisées, PHP est prêt à être testé. Il ne reste qu'à démarrer Apache et tester avec un fichier tout bête :

```
<? phpinfo() ?>
```

On constate alors l'affichage du détail de la configuration PHP et avec soulagement, qu'il s'agit bien de la version 5 !

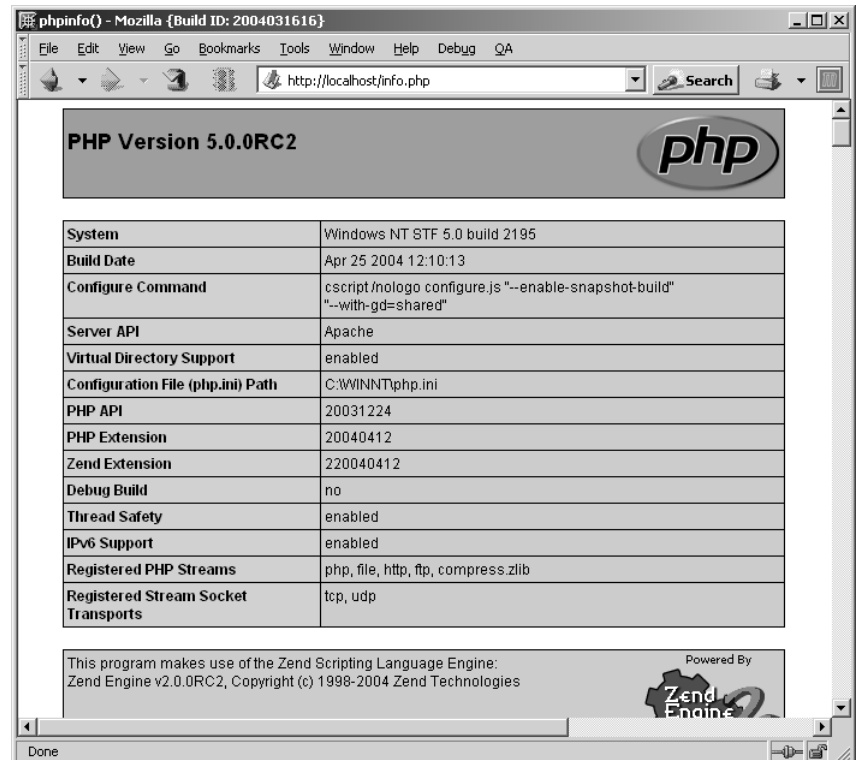


Figure A-7
Test d'intégration
entre Apache et PHP 5

Sous Linux

Installer Apache

Le plus souvent, Apache est déjà installé par les distributions Linux. Si tel n'est pas le cas, la procédure sera encore plus simple et expéditive que sous Microsoft Windows.

Si vous utilisez Debian, la commande suivante suffira :

```
apt-get install apache
```

Vous pouvez aussi utiliser synaptic.

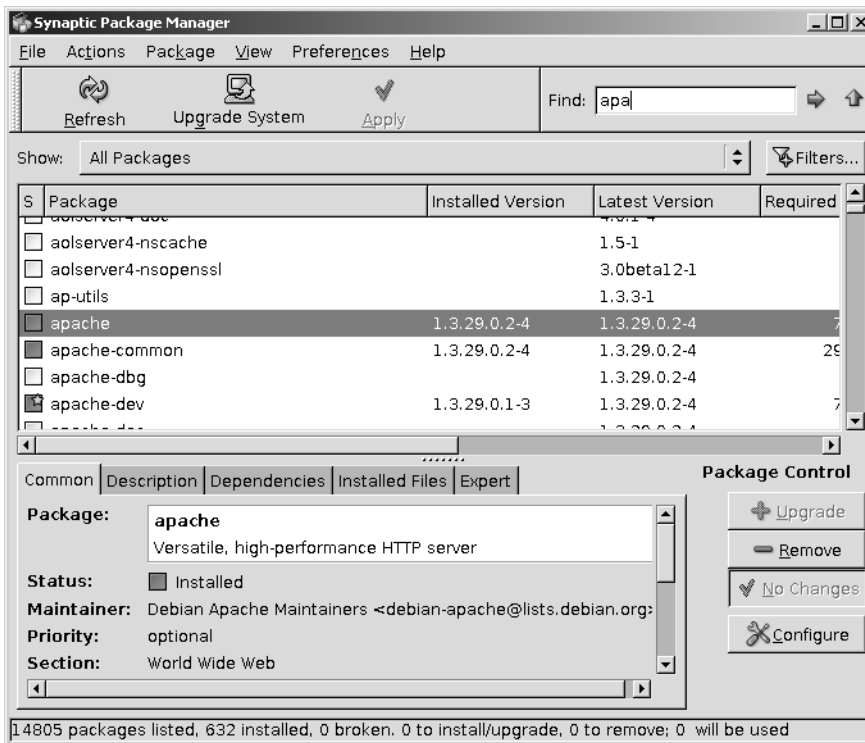
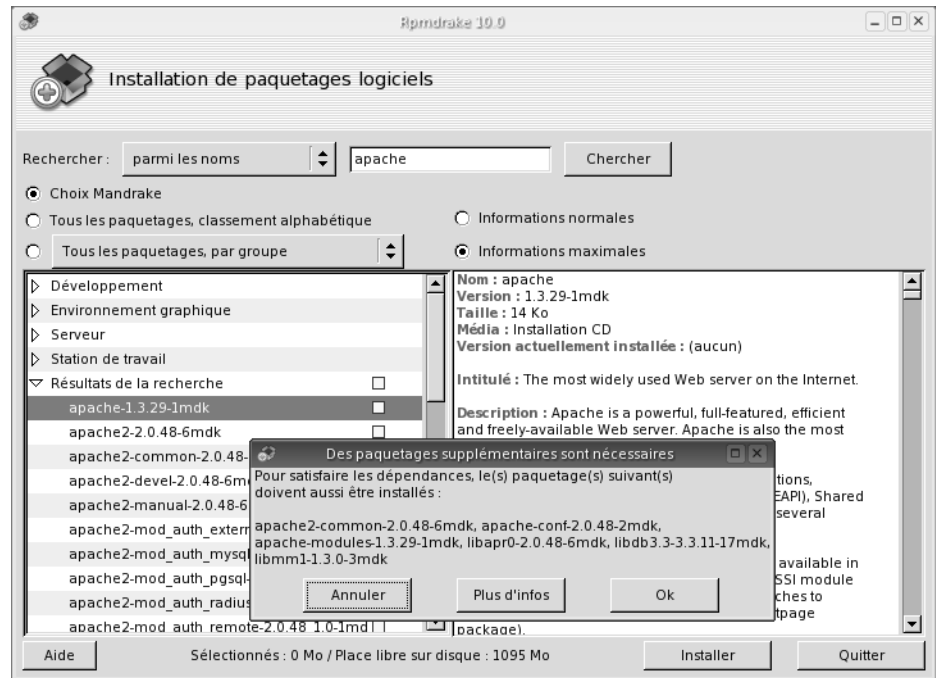


Figure A-8
Installation d'Apache
sous Debian avec synaptic

Pour la distribution Mandrakelinux, l'utilisation de `rpm` et une simple case à cocher devraient résoudre le problème. La plupart du temps, la racine du site web est positionnée sur le répertoire `/var/www`.

Figure A-9
Installation d'Apache
avec `rpm`
sous Mandrakelinux



Installer PHP 5

Pour PHP 5, les choses sont un peu moins simples, car à ce jour aucune distribution n'intègre encore le package associé à PHP 5. La solution la plus simple consiste alors à compiler soit même.

L'opération n'est pas complexe en soi, mais faute d'habitude on peut être amené à tâtonner un peu pour retrouver les outils nécessaires à la compilation ou à l'activation des extensions PHP souhaitées. Il s'agit bien sûr des outils de développements traditionnels (compilateur, en-têtes systèmes) mais les en-têtes de bibliothèques déjà installées seront le plus souvent nécessaires (par exemple pour GD si l'on souhaite obtenir le support des images JPEG, PNG...). Toutefois, dans la plupart des cas l'outil « configure » livré avec PHP saura indiquer les éléments manquants.

La procédure de compilation en elle-même est sans surprise, voici la séquence d'opération utilisée pour PHP Saloon :

```
./configure --with-sqlite --with-apxs --with-xsl --with-gd --with-zlib
--with-jpeg-dir=/usr
make
```

D'autres extensions peuvent naturellement être ajoutées. Pour plus de détail, consulter le fichier `INSTALL` ou l'aide de `configure`. Au terme de la compilation, l'installation est automatisée, il suffit d'invoquer la commande :

```
make install
```

La configuration d'Apache est automatiquement modifiée et PHP est prêt à être utilisé. Comme sous Windows, on pourra s'en assurer avec un simple `phpinfo()`. Il peut toutefois être utile d'ajouter `index.php` comme alternative à `index.html` (voir l'installation Windows).

Tester PHP Saloon

Que faire une fois ces installations réalisées ? Une toute dernière opération : copier le code de PHP Saloon à la racine de votre site web. Quelques ajustements en termes de droits sur les fichiers peuvent être nécessaires sous Linux mais, sauf complication imprévue, PHP Saloon devrait être directement opérationnel.