Apache+DSO+mod_ssl+mod_perl+php+mod_auth_nds+mod_auth_mysql+mod_fastcgi

Ray Van Dolson, *rayvd@firetail.org* ;
version française : Xavier FACQ, *xavier.facq@free.fr* . v0.91, 5 avril 2000

Ce document traite de la mise en place d'un serveur Apache intégrant des modules DSO (*Dynamic Shared Object*, NdT : *bibliothèque d'objets partagée*) comme par exemple mod_perl, mod_ssl et php.

Contents

1 Copyright

2 Introduction

Ce document va expliquer l'installation d'Apache et de ses modules sur un serveur du Walla Walla College (www.wwc.edu). Bien qu'il soit spécifique à notre système, il peut servir d'aide précieuse pour faire d'autres installations.

Ce document tentera d'expliquer les étapes exactes utilisées pour installer le serveur. Des remarques seront faites lorsque les choses ont été réalisées différemment, mais la marche à suivre originale sera donnée (en supposant que ça marche).

2.1 Description des composants

L'installation a été réalisée sur une Red Hat 6.1, noyau 2.2.14, fonctionnant sur un Dual PIII 600 avec RAID5 et plein d'autres choses alléchantes.

Le serveur-web est Apache 1.3.12. Les modules suivants ont été ajouté au serveur :

- mod_fastcgi SNAP (et mod_rewrite), utilisé avec Zope ;
- Auth-MySQL 2.20 ;
- mod_ssl 2.6.2 (Open-SSL 0.9.5) ;
- mod_perl 1.21 ;
- PHP 3.0.15 ;
- mod_auth_nds 0.3a.

2.2 Historique

v0.91 (5 avril 2000)

- mise à jour de mod_fastcgi à la version correcte.

v0.9 (4 avril 2000)

- brouillon terminé ;
- correction des erreurs d'orthographe et de grammaire.

v0.1 (mars 2000)

- premier jet.

3 Installation des composants

3.1 Prérequis

Vous aurez besoin des logiciels suivant :

- *Apache 1.3.12* `http://www.apache.org/dist/apache_1.3.12.tar.gz`
- *PHP 3.0.15* `http://www.php.net/distributions/php-3.0.15.tar.gz`
  - GD 1.3 (pour l'utilisation des fichiers GIF)
    - *Source* `http://www.boutell.com/gd/http/gd-1.3.tar.gz`
    - *RPM* `ftp://rpmfind.net/linux/redhat/redhat-6.0/i386/RedHat/RPMS/gd-1.3-5.i386.rpm`
    - *RPM-devel* `ftp://rpmfind.net/linux/redhat/redhat-6.0/i386/RedHat/RPMS/gd-devel-1.3-5.i386.rpm`
  - GD 1.8.1 (pour l'utilisation des fichiers PNG)
    - *Source* `http://www.boutell.com/gd/http/gd-1.8.1.tar.gz`
    - *RPM* `ftp://rpmfind.net/linux/contrib/libc6/i386/gd-1.8.1-1.i386.rpm`
    - *RPM-devel* `ftp://rpmfind.net/linux/contrib/libc6/i386/gd-devel-1.8.1-1.i386.rpm`
  - IMAP 4.5+
    - *Source* `ftp://ftp.cac.washington.edu/imap/imap-4.7b.tar.Z`
    - *RPM* `ftp://rpmfind.net/linux/redhat/redhat-6.1/i386/RedHat/RPMS/imap-4.5-4.i386.rpm`
  - OpenLDAP 1.2.9+
    - *Source* `ftp://ftp.openldap.org/pub/OpenLDAP/openldap-release/openldap-1.2.9.tgz`
    - *RPM* `ftp://rpmfind.net/linux/rawhide/1.0/i386/RedHat/RPMS/openldap-1.2.9-5.i386.rpm`
    - *RPM-devel* `ftp://rpmfind.net/linux/rawhide/1.0/i386/RedHat/RPMS/openldap-devel-1.2.9-5.i386.rpm`

- *mod_perl 1.22+* `http://perl.apache.org/dist/mod_perl-1.22.tar.gz`
  - Modules pour Perl5
  - *MIME::Base64* `http://amaunet.informatik.uni-dortmund.de/cgi-bin/CPAN/authors/id/GAAS/MIME-Base64-2.11.tar.gz`
  - *URI* `http://www.perl.com/CPAN-local/modules/by-module/URI/URI-1.05.tar.gz`
  - *HTML-Parser* `http://www.perl.com/CPAN-local/modules/by-module/HTML/HTML-Parser-3.07.tar.gz`
  - *Digest-MD5* `http://www.perl.com/CPAN-local/modules/by-module/Digest/Digest-MD5-2.09.tar.gz`
  - *libnet* `http://www.perl.com/CPAN-local/modules/by-module/Net/libnet-1.0703.tar.gz`
  - *libwww* `http://www.perl.com/CPAN-local/modules/by-module/WWW/libwww-perl-5.47.tar.gz`
- *mod_ssl 2.6.2+* `http://www.modssl.org/source/mod_ssl-2.6.2-1.3.12.tar.gz`
  - OpenSSL 0.9.5
    - *Source* `http://www.openssl.org/source/openssl-0.9.5.tar.gz`
    - *RPM* `ftp://rpmfind.net/linux/contrib/libc6/i386/openssl-0.9.5-1.i386.rpm`
    - *RPM-devel* `ftp://rpmfind.net/linux/contrib/libc6/i386/openssl-devel-0.9.5-1.i386.rpm`
  - *RSAREF 2.0* `ftp://ftp.wwc.edu/pub/apache/modules/mod_ssl/depend/rsaref20.1996.tar.Z`
  - *MM 1.0.12* `http://www.engelschall.com/sw/mm/mm-1.0.12.tar.gz`
- *MySQL 3.22.32* `http://www.mysql.com/Downloads/MySQL-3.22/mysql-3.22.32.tar.gz`
- *mod_auth_nds 0.4* `ftp://vrooom.at.drew.edu/pub/mod_auth_nds/mod_auth_nds-0.4.tar.gz`
  - *ncpfs 2.2.0.17* `ftp://vrooom.at.drew.edu/pub/mod_auth_nds/ncpfs/ncpfs-2.2.0.17.tgz`
    Note : le noyau doit être compilé avec le support IPX.
- *mod_auth_mysql 2.20* `http://bourbon.netvision.net.il/mysql/mod_auth_mysql/mod_auth_mysql-2.20.tar.gz`
- *mod_fastcgi SNAP Oct06* `http://www.FastCGI.com/dist/mod_fastcgi_SNAP_Oct06.tar.gz`

Voici l'arborescense que j'ai utilisé et que je vous recommande :

```
+ /usr/src
|
+-+ apache
  |
  +-+ apache-1.3.12
  |
  +-+ modules
  | |
  | +-+ mod_perl
  | | |
  | | +- mod_perl-1.21
```

```
| | |
| | +-+ depend
| |   |
| |   +- <perl modules here>
| |
| +-+ mod_ssl
| | |
| | +- mod_ssl-2.6.2-1.3.12
| | |
| | +-+ depend
| |   |
| |   +- openssl-0.9.5
| |   |
| |   +- rsaref-2.0
| |   |
| |   +- mm-1.0.12
| |
| +-+ mod_fastcgi_SNAP
| |
| +-+ php
| | |
| | +- php-3.0.15
| | |
| | +-+ depend
| |   |
| |   +- gd-1.3
| |   |
| |   +- imap-4.5
| |   |
| |   +- openldap-1.2.9
| |
| +-+ mod_auth_nds
| | |
| | +- mod_auth_nds-0.4
| | |
| | +-+ depend
| |   |
| |   +- ncpfs-2.2.0.17
| |
| +-+ mod_auth_mysql
|
+-+ mysql
```

Vérifiez que les logiciels cités ci-dessus sont déjà installés sur votre système. En général, ça ne peut pas faire de mal de tous les télécharger et de les réinstaller au cas où, car vous pouvez avoir une ancienne version sur votre machine.

3.2 mod_ssl

3.2.1 Installation et compilation de OpenSSL

mod_ssl a besoin qu'un moteur SSL soit installé. OpenSSL est le choix « naturel » pour un environnement Linux. Vous pouvez soit l'installer à partir d'un RPM (comme moi), soit compiler les sources. Si vous choisissez de le compiler, il faut ensuite l'installer sur le système dans `/usr/local` par exemple, ou faire pointer les applications qui en ont besoin vers celui-ci.

Les RPM installeront OpenSSL dans le répertoire du système.

3.2.2 Installation et compilation de RSAREF 2.0

Créez un répertoire rsaref-2.0 où vous le souhaitez.
Personnellement, j'ai utilisé `/usr/src/apache/modules/mod_ssl-blah/depend/`. Déplacez vous dans ce répertoire.

```
cd rsaref-2.0
cp -rp install/unix local
cd local
make
mv rsaref librsaref.a
```

Ces commandes doivent compiler la librarie rsaref ! Quand vous aurez besoin d'y faire référence, configurez votre script dans ce répertoire.

3.2.3 Installation et compilation de MM

Décompressez mm-1.0.12 (ou une version plus récente) dans le sous-répertoire `depend` de mod_ssl-blah. Puis suivez les instructions suivantes :

```
cd mm-1.0.12
./configure --disable-shared
make
```

Cela devrait compiler la librarie mm. Comme ci-dessus, référencez le répertoire quand vous en aurez besoin. Vous pouvez aussi installer cette librairie sur le système.

3.2.4 Installation et compilation de mod_ssl

La procédure normale avec apxs est de compiler Apache en premier, puis, en utilisant apxs, de compiler les modules que vous souhaitez utiliser et de les insérer sur le serveur. Cependant, mod_ssl doit être compilé sur le serveur avant que vous ne puissiez l'utiliser via apxs. Une fois que mod_ssl est sur le serveur, vous pouvez alors le mettre à jour via apxs sans recompiler Apache entièrement.

Allez dans le répertoire où vous avez compilé mod_ssl et exécutez les commandes suivantes pour la première compilation :

```
#!/bin/sh

./configure \
        --with-apache=/usr/src/apache/apache_1.3.12 \
```

```
            --with-ssl \
            --with-rsa=../depend/rsaref-2.0/local \
            --with-mm=../depend/mm-1.0.12 \
            --enable-shared=ssl
```

Vous n'avez pas besoin de faire un **make** ou quoi que ce soit d'autre. Quand vous compilerez Apache, tout se fera en même temps.

Ces lignes de configuration donnent deux exemples de comment votre système peut être configuré. Dans mon cas, OpenSSL était déjà installé quelque part (probablement dans `/usr/lib`, `/usr/include`). Par conséquent, je n'ai pas eu besoin de lui passer en paramètre le répertoire, contrairement à rsa and mm qui n'étaient pas déjà présent. J'ai donc du les compiler et laisser les sources dans leur répertoire (ne faites pas de make install ni rien). Dans ce cas, vous devez indiquer lors du « configure » le répertoire adéquate pour qu'il puisse trouver les en-tête et les libraires.

À ce point, à moins que vous ne mettiez à niveau Apache (auquel cas, vous devrez refaire le point précédent pour la nouvelle version), vous pouvez utiliser apxs pour mettre à jour et recompiler mod_ssl. Voici mon script de configuration :

```
./configure \
        --with-apxs=/apps/apache-1.3.12/bin/apxs \
        --with-ssl=../depend/openssl-0.9.4 \
        --with-rsa=../depend/rsaref-2.0/local \
        --with-mm=../depend/mm-1.0.12
```

Ou quelque combinaison de ce qui précède. Puis exécutez :

```
make
make install
make distclean
```

pour mettre fin à l'installation.

Note : *MM n'est pas nécessaire pour compiler mod_ssl. Si vous avez des problèmes, ne le compilez pas et enlevez-le des paramètres de « ./configure ».*

Quand j'ai compilé mod_ssl, j'ai eu des erreurs avec DBM. Pour résourdre cela, j'ai ajouté -lndbm dans le Makefile :

- lancez le script ci-dessus
- déplacez-vous dans pkg.sslmod
- éditez le makefile et ajoutez -lndbm à LIBS_SHLIB. Cela doit ressembler à :
- `LIBS_SHLIB=-lm -lcrypt -lndbdm`

Ainsi, vous ne devriez plus avoir de problème.

3.3 Apache

Désarchivez apache-1.3.12.tar.gz dans `/usr/src/apache` ou autre part. Ensuite, vous devez compiler Apache avec les options suivantes :

- mod_ssl (pour compiler mod_ssl comme une bibliothèque, il doit l'être normalement en premier. Après, il pourra être mis à niveau via apxs.)

- mod_proxy
- mod_so
- mod_rewrite (pour l'utilisez avec Zope)

Voici le fichier de configuration que j'utilise pour compiler Apache :

```
#!/bin/sh

SSL_BASE=../depend/openssl-0.9.4 \
RSA_BASE=../depend/rsaref-2.0/local \
EAPI_MM=../depend/mm-1.0.12 \
./configure \
        --enable-module=ssl \
        --enable-module=proxy\
        --enable-shared=proxy\
        --enable-module=rewrite \
        --enable-shared=rewrite \
        --prefix=/apps/apache-1.3.12 \
        --enable-shared=ssl \
        --enable-rule=SHARED_CORE \
        --enable-rule=SHARED_CHAIN \
        --enable-module=so
```

Puis :

```
make
make certificate
make install
```

Apache doit maintenant être compilé et installé dans le répertoire que vous avez spécifier avec `-prefix`. Vérifez qu'il fonctionne correctement.

`/path/to/apache/bin/apachectl start`

ou

`/path/to/apache/bin/apachectl startssl`

Si tout va bien, Apache doit fonctionner normalement. Sinon, reprennez les manipulations depuis le début, et vérifiez que vous n'avez rien oublié.

3.4 MySQL

php, tout comme mod_auth_mysql et peut-être mod_perl, à besoin que MySQL soit installé et qu'il fontionne. Il est au-delà des intentions de ce document que d'entrer dans les détails de l'installation de MySQL. Mais vous pouvez télécharger l'archive et suivre les instructions des fichiers INSTALL. C'est assez simple de mettre en service MySQL. Ça doit être quelque chose comme :

```
./configure
make
make install
```

Comme vous devriez avoir tout d'installé, vous pouvez maintenant compiler les autres modules d'Apache.

3.5 PHP 3.0.15

Vous allez maintenant compiler php-3.0.15 comme une bibliothèque, ce qui veut dire que c'est un module qui peut être démarré ou arrêté indépendamment du serveur. C'est ainsi plus facile de mettre à niveau php sans avoir à tout recompiler (ce qui peut être pénible si vous utilisez beaucoup de modules avec Apache).

3.5.1 GD

Dans mon installation d'Apache, php utilise gd pour créer des images et d'autres choses. J'utilise une vieille version de gd (installée via RPM) pour faire la liaison avec php. Comme cela, on peut utiliser des fichiers GIF en sortie. Cela n'est peut être pas très bien vis à vis des copyright, ainsi vous devriez utiliser une version supérieure à la 1.3 qui ne supporte que les fichiers PNG.

Vous pouvez l'installer soit à partir d'un RPM (rpm -i gd*.rpm), soit compiler les sources et l'installer sur votre système.

3.5.2 IMAP

Si vous voulez avoir le support IMAP, la procédure est similaire à gd. J'ai aussi utilisé un RPM puisque je suis sur une Red Hat, mais l'installer à partir des sources est une procédure assez simple :

```
./configure
make
make install
```

3.5.3 OpenLDAP

Vous pouvez aussi installer OpenLDAP à partir de RPM ou des sources. J'ai choisi de le faire à partir des sources, car la dernière version n'était pas encore disponible en RPM au moment de mon installation.

```
./configure
make
make install
```

devrait faire l'affaire (ou bien rpm -i openldap*.rpm).

3.5.4 Installation et compilation de PHP 3.0.15

Maintenant que les modules ci-dessus sont installés et qu'ils fonctionnent, vous pouvez compiler php comme une bibliothèque. Le principe est simple et rapide :

```
cd /usr/src/apache/modules/php/php-3.0.15

./configure \
        --with-apxs=/apps/apache/bin/apxs \
        --with-config-file-path=/apps/etc \
        --with-gd \
        --with-imap \
        --with-mysql=/apps/mysql \
        --with-ldap=/apps \
```

```
        --with-zlib \
        --enable-track-vars
```

Soyez sûr que si vos libraries `-with` ne sont pas installées dans `/usr/local` ou `/usr`, vous les rajoutiez sur une ligne =/localisation/, pour que configure puisse trouver les trucs dont il a besoin !

```
make
make install
```

Si tout c'est bien passé, « make install » va utiliser apxs pour installer libphp3.so dans `/apache/libexec/libphp3.so`, ajouter les entrées nécessaires dans httpd.conf et activer php3.

3.6 mod_perl

Cette section explique l'installation de mod_perl comme une bibliothèque d'Apache. Il y a beaucoup de modules perl (en plus, bien sûr, de perl5 que je suppose vous avez déjà installé) qui doivent être ajoutés avant mod_perl pour compiler sans problèmes. Si vous n'installez pas ces modules, mod_perl devrait se plaindre et vous indiquer lesquels manquent.

Il y a un certain ordre à respecter dans l'installation de ces modules. Certains dépendent d'autres, j'ai donc énuméré l'ordre dans lequel je les ai installé sans problème.

3.6.1 Modules perl requis

Les modules perl peuvent être trouvés aux url données plus haut dans ce document. Téléchargez-les et mettez-les où vous le souhaitez ou dans le répertoire que j'ai utilisé (voir aussi 3.1 (arborescense utilisée)). L'installation d'un module est simple. Après l'avoir désarchivé dans le répertoire (normalement avec : tar xvfz), il suffit de faire :

```
perl Makefile.PL
make
make install
```

Si tout ce passe bien, cela devrait configurer, compiler et installer le module perl à votre place. Bien sûr vous pouvez lire les fichiers README en cas d'erreurs.

Voici l'ordre de l'installation des modules nécessaires pour mod_perl :

1. MIME::Base64
2. URI
3. HTML::Parser
4. Digest-MD5
5. libnet
6. libwww

3.6.2 Installation et compilation de mod_perl 1.2x

Après avoir installé les modules perl, vous êtes prêt pour compiler et installer mod_perl dans Apache. Allez dans le répertoire où vous avez extrait mod_perl, et lancez le script suivant :

```
perl Makefile.PL \
        USE_APXS=1 \
        WITH_APXS=/path/to/apache/bin/apxs \
        EVERYTHING=1
```

Cela devrait configurer votre Makefile et dire à mod_perl de se compiler tout seul comme une bibliothèque en utilisant apxs (dans le répertoire que vous devez spécifier). Après, vous n'avez plus qu'à faire :

```
make
make install
```

Ainsi, mod_perl sera déplacé dans le répertoire approprié, et les lignes ajoutées à votre fichier httpd.conf.

3.7 mod_auth_mysql

mod_auth_mysql permet à Apache d'accéder à la base utilisateur MySQL. L'installation comme une bibliothèque n'est pas très bien documentée dans le fichier README, mais c'est possible.

Premièrement, allez dans le répertoire où vous avez extrait mod_auth_mysql. Normalement, vous avez MySQL d'installé (avec les fichiers d'en-tête, etc.). Regardez où sont installées les libraries MySQL et les fichiers d'en-tête. Dans le doute, vérifiez dans `/usr/lib/mysql` et dans `/usr/include/mysql`.

Pour compiler mod_auth_mysql, vous devez renommer le fichier « config.h » en « auth_mysql_config.h ». Je ne sais pas pourquoi ce fichier n'a pas le bon nom, mais faites-le grâce à cette commande :

```
cp config.h auth_mysql_config.h
```

Et pour finir :

```
/path/to/apache/bin/apxs -i -a -I/usr/include/mysql -L/usr/lib/mysql \
        -lmysqlclient -c mod_auth_mysql.c
```

Il faudra peut-être l'exécuter en super-utilisateur (*root*) si vous n'avez pas les droits de lecture/écriture sur le répertoire d'Apache.

3.8 mod_auth_nds

Dans mon école, le réseau Windows choisi est de type Netware. C'est resté comme ça longtemps, et bien qu'il fonctionnait bien, maintenant on l'utilise uniquement pour le partage de fichiers et le courriel. Tous les étudiants ont un compte Netware dans lequel les fichiers personnels, ainsi que les pages web, sont stockés. Chacun peut monter ses répertoires à partir du serveur Linux, ainsi on peut protéger les données avec les informations du compte Netware. Avec ce module, Apache peut faire l'authentification à travers le réseau Netware.

3.8.1 ncpfs

Pour compiler mod_auth_nds, il faut que ncpfs soit installé (ainsi que les fichiers d'en-tête). Avant de compiler ncpfs, vous devez vous assurer que votre noyau supporte IPX. Si c'est le cas, faites juste :

```
./configure
make
make install    # optionnel
```

qui copira (et installera) les libraries.

3.8.2 Installation et compilation de mod_auth_nds

Si ncpfs est installé, en lancant les commandes suivantes, cela devrait compiler mod_auth_nds comme une bibliothèque :

```
/path/to/apache/bin/apxs -c -lncp -L/usr/lib -I/usr/include mod_auth_nds.c
/path/to/apache/bin/apxs -i mod_auth_nds.so
```

Ensuite, ajoutez les lignes suivantes au fichier httpd.conf (à la main) :

```
LoadModule nds_auth_module libexec/mod_auth_nds.so
AddModule mod_auth_nds.c
```

Il ne vous reste plus qu'à redémarrer Apache !

3.9 mod_fastcgi

L'installation de mod_fastcgi est nécessaire si vous voulez accéder à votre serveur Zope avec Apache. C'est plus pratique, car Apache est plus sécurisé et plus configurable que le serveur Zope.

La version stable actuelle de mod_fastcgi est 2.2.2, mais cette version ne fonctionne pas correctement avec Zope. Vous devez donc vous procurer SNAP release qui date du 06 octobre 1999 dont le lien est donné plus haut.

Allez dans le répertoire de mod_fastcgi et lancez ces commandes :

```
/path/to/apache/bin/apxs -o mod_fastcgi.so -c *.c
/path/to/apache/bin/apxs -i -a -n fastcgi mod_fastcgi.so
```

Voir la documentation pour la description de son utilisation.

4 Mots de la fin

La plupart de ces informations peuvent être obtenues en lisant les fichiers README et INSTALL de chacuns des modules. Cependant, ce document est utile dans le cas où ça ne fonctionne pas ou que la documentation n'est pas très explicite. Mais il a surtout l'avantage d'exister et d'être plus simple à suivre et à comprendre que de passer d'un README à un autre.

4.1 Remerciements

Phillip R. Wilson <pwilson@drew.edu>, auteur de mod_auth_nds, pour m'avoir aidé
à compiler et installer cmod_auth_nds avec apxs.

John Ash <jash@wwc.edu>, mon patron, pour son aide et surtout pour mon travail.

Marcus Faure <marcus@faure.de>, auteur de Apache SSL PHP/FI frontpage mini-HOWTO,
dont le document m'a beaucoup aidé.

4.2 Contact

Si vous avez trouvé des fautes de frappe ou de grammaire dans ce document, n'hésitez pas à m'envoyer un courriel.

Ray Van Dolson
Courriel : rayvd@firetail.org
IRC : DALnet, #Bludgeon (nick Variant)

NdT : le lecteur francophone pourra en faire autant...

Xavier Facq
Courriel : xavier.facq@free.fr

4.3 Autre

Tout ce qui est mentionné dans ce document sera éventuellement disponible en ftp sur ftp.wwc.edu/pub/apache. Il y aura tout ce qui est présenté ci-dessus, ainsi que les scripts d'installation.