Tutorial d'installation d'APACHE / OPENSSL / PHP sous Windows

Etant donné qu'il y a très peu de documentation sur cette procédure, qui plus est en français, mais surtout sous Windows (vous ne trouverez pas le module SSL pour Windows, ni sa documentation sur le site du développeur www.openssl.org !!!), je me suis décidé à faire ce tutorial pour aider tous ceux désirent installer Apache avec le module de cryptage SSL, en simplifiant et synthétisant le plus possible chaque étape. Si vous désirez installer d'autres modules comme MySQL ou PhpMyAdmin, sachez que d'autres tutoriaux existent sur http://php.developpez.com/. Je me suis inspiré du tutorial présent sur PHPFrance.com et à celui de J.F. Moreau afin d'arriver à mes fins, pour cela je les remercie grandement.

Pour tout contact : XzoF

I - Pré-requis :

1. Télécharger Apache 1.3.29 sans SSL ici : http://www.apachefrance.com/Telechargement/download.php?url=http://apache.belnet.be/dist/httpd/binaries/win32/apache_1.3.29-win32-x86-no_src.exe
2. Télécharger Apache avec SSL ici : http://tor.ath.cx/~hunter/apache/Apache_1.3.29-Mod_SSL_2.8.16-Openssl_0.9.7c-Win32.zip
3. Télécharger le binaire zippé de PHP ici : http://fr.php.net/get/php-4.3.4-Win32.zip/from/this/mirror
4. Télécharger le module SSL ici : http://hunter.campbus.com/Openssl-0.9.7c-Win32.zip
5. Terminer tous les processus d'Apache en cours sur votre système

II - Installation et configuration de PHP

1. Créer un répertoire « **serveur** » à la racine de votre disque dur, tous les modules serveurs y seront installés
2. Créer un répertoire « **php** » dans ce répertoire « serveur » et dé zippez-y tout le contenu du fichier du dossier « **php-4.3.4-Win32** » situé à l'intérieur du zip
3. Copier le fichier « **php.ini-dist** » de votre répertoire « **c:\serveur\php** » dans le répertoire racine de votre système « **c:\windows** » ou « **c:\winnt** » s'il s'agit d'un système NT et renommez le en « **php.ini** ». Copier aussi la dll « **php4ts.dll** » dans le répertoire « **system32** » du répertoire windows.
4. Ouvrez le « **php.ini** » et éditez le comme suit :
    - A la rubrique « **Paths and Directories** » (Enlever le « ; » en début de chaque propriété pour l'activer)

```
« include_path = .
doc_root =
user_dir =
upload_max_filesize = 2097152
extension_dir = c:\serveur\php\extensions
enable_dl = On »
```

- A la rubrique « **Windows Extensions** » : Vous enlèverez plus tard le « ; » en début de la propriété « extension=php_openssl.dll » afin de l'activer, lorsque nous installerons le module **SSL**.

III – Installation et configuration d'Apache

1. Installer Apache sans **SSL**. Lors de la demande d'information du serveur, mettez :
   - **Network Domain** : votre domaine sur lequel est situé votre ordinateur (Mettez localhost si vous utilisez ce serveur en local)
   - **Server Name** : mettez l'IP locale soit « 127.0.0.1 »
   - **Administrator Email** : mettez votre adresse Internet.
   - Sélectionnez l'option « **Run as service** » pour l'exécuter en tant que service de Windows
2. Mettez ensuite comme répertoire d'installation **« c:\serveur** »
3. A la fin de l'installation, revenez dans le répertoire « c:\serveur » et créez un répertoire « **www** » qui contiendra vos fichiers web.
4. Configurer le fichier « **httpd.conf** » du répertoire « **conf** » d'Apache avec les instructions suivantes concernant le fonctionnement d'Apache :
   - Rechercher la variable « **DocumentRoot** » et mettez le répertoire : « **c:/serveur/www** » (Attention, dans le fichier de configuration apache, tous les anti-slashs « \ » doivent être transformés en slashs « / »)
   - Recherchez ensuite les balises juste en dessous « **<Directory />…</Directory>** » et modifiez comme suit :
     « <Directory "c:/serveur/www">
     Options All
     AllowOverride All
     Order allow,deny
     Allow from all
     </Directory> »
   - Rechercher ensuite les balises :
     « <IfModule mod_dir.c>
     DirectoryIndex index.html
     </IfModule>» et ajouter « index.php » après « DirectoryIndex index.html » séparé d'un simple espace. Cela permet d'exécuter ce fichier lorsqu'un répertoire est saisi directement dans le navigateur.
5. Cette étape de configuration concerne le fonctionnement de PHP avec Apache. Toujours dans « **httpd.conf** » :
   - Ajouter les lignes suivantes (elles permettent l'interprétation des fichiers avec les extensions suivantes) :
     « AddType application/x-httpd-php .php
     AddType application/x-httpd-php .php3
     AddType application/x-httpd-php .phtml »
   - Ajouter ensuites ces autres lignes :
     « ScriptAlias /php/ c:/serveur/php/
     Action application/x-httpd-php /php/php.exe » Cela permet l'exécution de PHP lors de la rencontre de fichiers PHP.

- Dans la partie des « **LoadModule** » ajouter la ligne suivante :
  « LoadModule php4_module c:/serveur/php/sapi/php4apache.dll » Cette ligne appelle le module PHP.

6. Voila, la configuraiton d'Apache avec PHP est terminée, veuillez tester la configuration actuelle avant de passer à la suite. Pour cela, lancer « apache.exe » du répertoire « **c:\serveur\apache** ». Si vous avez des erreurs, retrouvez les dans les le fichier « **error.log** » du répertoire « **logs** » d'Apache, et repassez par les étapes précédentes.
   Si le serveur se lance sans problème, créer un fichier « index.php » dans le répertoire « www » avec le code « **<? phpinfo() ; ?>** » qui affiche toutes les informations sur PHP.
   Lancer enfin un navigateur et mettez, l'IP locale c'est-à-dire 127.0.0.1 avec le protocole http soit « http://127.0.0.1 »
   Normalement, vous devriez voir apparaître une liste d'informations sur PHP. Si ce n'est pas le cas re-vérifiez le fichier de configuration, sinon passez à la suite.
   Pour arrêter Apache, faites un « **Ctrl - C** » sur la fenêtre DOS.

IV – Installation de mod_ssl

1. Nous allons rééditer le « **httpd.conf** » d'Apache :
   - Chercher « **Port 80** » et mettez le en commentaire en le précédant d'un « **#** »
   - Ajouter « **Listen 80** » si IIS n'est pas en supplément
   - Ajouter « **Listen 443** » correspondant au port de communication d'**SSL**.
2. Maintenant, créer un nouveau répertoire nommé « **ssl** » dans le répertoire racine du serveur « **c:\serveur** » et décompressez le contenu du fichier « **Openssl-0.9.7c-Win32.zip** »
3. Copier les 2 DLL extraites « **libeay32.dll** » et « **ssleay32.dll** » dans le répertoire « **system32** » du répertoire « **Windows** »
4. Récupérer le fichier « **openssl.cnf** » du répertoire « **openssl** » de **PHP**. (Rq : l'extension *.cnf ne s'affiche pas dans l'explorateur de fichier) et copiez le dans le répertoire « **ssl** » que nous venons de créer.

V - Création d'un certificat temporaire

1. Nous allons maintenant créer un certificat temporaire et une clé privée à l'aide d'Openssl. Pour cela ouvrez une console Windows et positionnez vous dans le répertoire « **c:\serveur\ssl** » (Je rappelle que la fonction de navigation dans les répertoires est « **cd nom_du_repertoire** » pour monter dans l'arborescence et « **cd..** » pour revenir vers la racine. Exécutez les commandes suivantes :
   - « openssl req -config openssl.cnf -new -out nom-du-serveur.csr »

   Cela crée un CSR (Certificat Signing Request) et une clef privée. Lorsque l'on vous demande votre "nom de domaine", donnez le nom de domaine exact de votre serveur Web (par exemple **www.mon-serveur.com**, **195.125.204.24, D16**). Le certificat appartient à ce nom de serveur et les navigateurs se "plaignent" si le nom ne correspond pas.
   Plusieurs informations vous sont ensuite demandées :
   Le « **PEM pass phrase** » : par défaut c'est « **privkey.pem** »
   Le nom du pays, mettez « **FR** »
   L'état : entrez « **France** »
   La ville
   Le nom de votre organisation
   La section de l'organisation
   Votre nom

Votre Email
Un mot de passe de votre choix
Un autre nom de votre organisation

- « openssl rsa -in privkey.pem -out nom-du-serveur.key »
  le « **PEM pass phrase** » vous est encore demandé : par défaut « **privkey.pem** »
  Cela enlève la phrase/motdepasse de la clef privée. Vous DEVEZ comprendre ce que cela signifie; « **nom-du-serveur.key** » doit être lisible seulement par le serveur apache et l'administrateur. Vous devez supprimer le fichier « **.rnd** » parce qu'il contient l'information d'entropie pour créer la clef et pourrait être employé pour des attaques cryptographiques contre votre clef privée.
- « openssl x509 -in nom-du-serveur.csr -out nom-du-serveur.cert -req -signkey nom-du-serveur.key -days 365 »
  Cela crée un certificat signé que vous pouvez employer avant que vous n'en obteniez un "réel" d'une autorité de certification. (Le certificat réel est facultatif; si vous connaissez vos utilisateurs, vous pouvez leur dire d'installer le certificat dans leur(s) navigateur(s).) Notez que ce certificat expire après un an, vous pouvez modifier le délai (-days 365) si vous le désirez.
- Si vous avez des utilisateurs naviguant avec MS Internet Explorer 4.x et vouliez qu'ils soient capables d'installer le certificat dans leur navigateur (en le téléchargeant et en l'ouvrant), vous devez créer une version codée DER du certificat :
  « openssl x509 -in nom-du-serveur.cert -out nom-du-serveur.der.crt -outform DER »

2. Créez un répertoire « **c:\Apache\conf\ssl** »et déplacez-y les fichiers « nom-du-serveur.key » et « nom-du-serveur.cert. » créés dans le répertoire « **c:\serveur\ssl** »

VI – Configuration d'Apache et mod_ssl

1. Premièrement, arrêter le serveur
2. **ATTENTION, TRES IMPORTANT : SAUVEGARDER VOTRE FICHIER DE CONFIGURATION « httpd.conf » dans un répertoire temporaire, car il sera écrasé par la suite.**
3. Dé zippez tout le contenu du fichier « **Apache_1.3.29-Mod_SSL_2.8.16-Openssl_0.9.7c-Win32.zip** » que vous avez téléchargé, dans le répertoire Apache, en acceptant de tout écraser.
4. Ré insérer le « **httpd.conf** » dans le répertoire « **conf** » et éditez le en ajoutant ces lignes :
   - « LoadModule ssl_module modules/mod_ssl.so »
   - « AddModule mod_ssl.c »
   - *« # voir http://www.modssl.org/docs/2.4/ssl_reference.html pour plus d'informations*
     SSLMutex sem
     SSLRandomSeed startup builtin
     SSLSessionCache none

     SSLLog logs/SSL.log
     SSLLogLevel info
     # Vous pouvez plus tard changer "info" en "warn" si tout est OK

<VirtualHost nom-du-serveur:443>
SSLEngine On
SSLCertificateFile C:/Serveur/Apache/conf/ssl/nom-du-serveur.cert
SSLCertificateKeyFile C:/Serveur/Apache/conf/ssl/nom-du-serveur.key
</VirtualHost> »

Testez.

V – Fichiers finaux

Voici les exemples des fichiers de configurations finaux :
« **Httpd.conf** » du répertoire « **c:\serveur\conf** »:

```
#
# Based upon the NCSA server configuration files originally by Rob McCool.
#
# This is the main Apache server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See <URL:http://www.apache.org/docs/> for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here without understanding
# what they do.  They're here only as hints or reminders.  If you are unsure
# consult the online docs. You have been warned.
#
# After this file is processed, the server will look for and process
# C:/Serveur/Apache/conf/srm.conf and then C:/Serveur/Apache/conf/access.conf
# unless you have overridden these with ResourceConfig and/or
# AccessConfig directives here.
#
# The configuration directives are grouped into three basic sections:
#  1. Directives that control the operation of the Apache server process as a
#     whole (the 'global environment').
#  2. Directives that define the parameters of the 'main' or 'default' server,
#     which responds to requests that aren't handled by a virtual host.
#     These directives also provide default values for the settings
#     of all virtual hosts.
#  3. Settings for virtual hosts, which allow Web requests to be sent to
#     different IP addresses or hostnames and have them handled by the
#     same Apache server process.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path.  If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so "logs/foo.log"
# with ServerRoot set to "/usr/local/apache" will be interpreted by the
# server as "/usr/local/apache/logs/foo.log".
#
# NOTE: Where filenames are specified, you must use forward slashes
# instead of backslashes (e.g., "c:/apache" instead of "c:\apache").
# If a drive letter is omitted, the drive on which Apache.exe is located
# will be used by default.  It is recommended that you always supply
# an explicit drive letter in absolute paths, however, to avoid
```

```
# confusion.
#

### Section 1: Global Environment
#
# The directives in this section affect the overall operation of Apache,
# such as the number of concurrent requests it can handle or where it
# can find its configuration files.
#

#
# ServerType is either inetd, or standalone.  Inetd mode is only supported on
# Unix platforms.
#
ServerType standalone

#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
ServerRoot "C:/Serveur/Apache"

#
# PidFile: The file in which the server should record its process
# identification number when it starts.
#
PidFile logs/httpd.pid

#
# ScoreBoardFile: File used to store internal server process information.
# Not all architectures require this.  But if yours does (you'll know because
# this file will be  created when you run Apache) then you *must* ensure that
# no two invocations of Apache share the same scoreboard file.
#
ScoreBoardFile logs/apache_runtime_status

#
# In the standard configuration, the server will process httpd.conf (this
# file, specified by the -f command line option), srm.conf, and access.conf
# in that order.  The latter two files are now distributed empty, as it is
# recommended that all directives be kept in a single file for simplicity.
# The commented-out values below are the built-in defaults.  You can have the
# server ignore these files altogether by using "/dev/null" (for Unix) or
# "nul" (for Win32) for the arguments to the directives.
#
#ResourceConfig conf/srm.conf
#AccessConfig conf/access.conf

#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300
```

```
#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On

#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100

#
# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
#
KeepAliveTimeout 15

#
# Apache on Win32 always creates one child process to handle requests.  If it
# dies, another child process is created automatically.  Within the child
# process multiple threads handle incoming requests.  The next two
# directives control the behaviour of the threads and processes.
#

#
# MaxRequestsPerChild: the number of requests each child process is
# allowed to process before the child dies.  The child will exit so
# as to avoid problems after prolonged use when Apache (and maybe the
# libraries it uses) leak memory or other resources.  On most systems, this
# isn't really needed, but a few (such as Solaris) do have notable leaks
# in the libraries.  For Win32, set this value to zero (unlimited)
# unless advised otherwise.
#
# NOTE: This value does not include keepalive requests after the initial
#       request per connection. For example, if a child process handles
#       an initial request and 10 subsequent "keptalive" requests, it
#       would only count as 1 request towards this limit.
#
MaxRequestsPerChild 0

#
# Number of concurrent threads (i.e., requests) the server will allow.
# Set this value according to the responsiveness of the server (more
# requests active at once means they're all handled more slowly) and
# the amount of system resources you'll allow the server to consume.
#
ThreadsPerChild 50

#
```

```
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
#Listen 3000
#Listen 12.34.56.78:80

#
# BindAddress: You can support virtual hosts with this option. This directive
# is used to tell the server which IP address to listen to. It can either
# contain "*", an IP address, or a fully qualified Internet domain name.
# See also the <VirtualHost> and Listen directives.
#
#BindAddress *

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Please read the file README.DSO in the Apache 1.3 distribution for more
# details about the DSO mechanism and run `apache -l' for the list of already
# built-in (statically linked and thus always available) modules in your Apache
# binary.
#
# Note: The order in which modules are loaded is important.  Don't change
# the order below without expert advice.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
#LoadModule vhost_alias_module modules/mod_vhost_alias.so
#LoadModule mime_magic_module modules/mod_mime_magic.so
#LoadModule status_module modules/mod_status.so
#LoadModule info_module modules/mod_info.so
#LoadModule speling_module modules/mod_speling.so
#LoadModule rewrite_module modules/mod_rewrite.so
#LoadModule anon_auth_module modules/mod_auth_anon.so
#LoadModule dbm_auth_module modules/mod_auth_dbm.so
#LoadModule digest_auth_module modules/mod_auth_digest.so
#LoadModule digest_module modules/mod_digest.so
#LoadModule proxy_module modules/mod_proxy.so
#LoadModule cern_meta_module modules/mod_cern_meta.so
#LoadModule expires_module modules/mod_expires.so
#LoadModule headers_module modules/mod_headers.so
#LoadModule usertrack_module modules/mod_usertrack.so
#LoadModule unique_id_module modules/mod_unique_id.so
LoadModule php4_module c:/serveur/php/sapi/php4apache.dll
LoadModule ssl_module modules/mod_ssl.so

#
```

```
#  Reconstruction of the complete module list from all available modules
#  (static and shared ones) to achieve correct module execution order.
#
# The modules listed below, without a corresponding LoadModule directive,
# are static bound into the standard Apache binary distribution for Windows.
#
# Note: The order in which modules are loaded is important.  Don't change
# the order below without expert advice.
#
# [WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE, UPDATE THIS
TOO!]
ClearModuleList
#AddModule mod_vhost_alias.c
AddModule mod_env.c
AddModule mod_log_config.c
#AddModule mod_mime_magic.c
AddModule mod_mime.c
AddModule mod_negotiation.c
#AddModule mod_status.c
#AddModule mod_info.c
AddModule mod_include.c
AddModule mod_autoindex.c
AddModule mod_dir.c
AddModule mod_isapi.c
AddModule mod_cgi.c
AddModule mod_asis.c
AddModule mod_imap.c
AddModule mod_actions.c
#AddModule mod_speling.c
AddModule mod_userdir.c
AddModule mod_alias.c
#AddModule mod_rewrite.c
AddModule mod_access.c
AddModule mod_auth.c
#AddModule mod_auth_anon.c
#AddModule mod_auth_dbm.c
#AddModule mod_auth_digest.c
#AddModule mod_digest.c
#AddModule mod_proxy.c
#AddModule mod_cern_meta.c
#AddModule mod_expires.c
#AddModule mod_headers.c
#AddModule mod_usertrack.c
#AddModule mod_unique_id.c
AddModule mod_so.c
AddModule mod_setenvif.c
AddModule mod_ssl.c

#
# ExtendedStatus controls whether Apache will generate "full" status
# information (ExtendedStatus On) or just basic information (ExtendedStatus
# Off) when the "server-status" handler is called. The default is Off.
```

```
#
#ExtendedStatus On

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values used by the 'main'
# server, which responds to any requests that aren't handled by a
# <VirtualHost> definition.  These values also provide defaults for
# any <VirtualHost> containers you may define later in the file.
#
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#

#
# Port: The port to which the standalone server listens.  Certain firewall
# products must be configured before Apache can listen to a specific port.
# Other running httpd servers will also interfere with this port.  Disable
# all firewall, security, and other services if you encounter problems.
# To help diagnose problems use the Windows NT command NETSTAT -a
#
#Port 80
listen 80
listen 443

#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed.  This address appears on some server-generated pages, such
# as error documents.
#
ServerAdmin votre_email@hotmail.com

#
# ServerName allows you to set a host name which is sent back to clients for
# your server if it's different than the one the program would get (i.e., use
# "www" instead of the host's real name).
#
# Note: You cannot just invent host names and hope they work. The name you
# define here must be a valid DNS name for your host. If you don't understand
# this, ask your network administrator.
# If your host doesn't have a registered DNS name, enter its IP address here.
# You will have to access it by its address (e.g., http://123.45.67.89/)
# anyway, and this will make redirections work in a sensible way.
#
# 127.0.0.1 is the TCP/IP local loop-back address, often named localhost. Your
# machine always knows itself by this address. If you use Apache strictly for
# local testing and development, you may use 127.0.0.1 as the server name.
#
ServerName 127.0.0.1
```

```
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "C:/Serveur/www"

#
# Each directory to which Apache has access, can be configured with respect
# to which services and features are allowed and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very restrictive set of
# permissions.
#
<Directory "c:/serveur/www">
Options All
 AllowOverride All
 Order allow,deny
 Allow from all
</Directory>


#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#

#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "C:/Serveur/Apache/htdocs">

#
# This may also be "None", "All", or any combination of "Indexes",
# "Includes", "FollowSymLinks", "ExecCGI", or "MultiViews".
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
    Options Indexes FollowSymLinks MultiViews

#
# This controls which options the .htaccess files in directories can
# override. Can also be "All", or any combination of "Options", "FileInfo",
# "AuthConfig", and "Limit"
#
    AllowOverride None

#
```

```
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all
</Directory>

#
# UserDir: The name of the directory which is appended onto a user's home
# directory if a ~user request is received.
#
# Under Win32, we do not currently try to determine the home directory of
# a Windows login, so a format such as that below needs to be used.  See
# the UserDir documentation for details.
#
<IfModule mod_userdir.c>
    UserDir "C:/Serveur/Apache/users/"
</IfModule>

#
# Control access to UserDir directories.  The following is an example
# for a site where these directories are restricted to read-only.
#
#<Directory "C:/Serveur/Apache/users">
#    AllowOverride FileInfo AuthConfig Limit
#    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
#    <Limit GET POST OPTIONS PROPFIND>
#        Order allow,deny
#        Allow from all
#    </Limit>
#    <LimitExcept GET POST OPTIONS PROPFIND>
#        Order deny,allow
#        Deny from all
#    </LimitExcept>
#</Directory>

#
# DirectoryIndex: Name of the file or files to use as a pre-written HTML
# directory index.  Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
    DirectoryIndex index.html index.php
</IfModule>

#
# AccessFileName: The name of the file to look for in each directory
# for access control information.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess files from being viewed by
# Web clients.  Since .htaccess files often contain authorization
```

```
# information, access is disallowed for security reasons.  Comment
# these lines out if you want Web visitors to see the contents of
# .htaccess files.  If you change the AccessFileName directive above,
# be sure to make the corresponding changes here.
#
# Also, folks tend to use names such as .htpasswd for password
# files, so this will protect those as well.
#
<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
    Satisfy All
</Files>

#
# CacheNegotiatedDocs: By default, Apache sends "Pragma: no-cache" with each
# document that was negotiated on the basis of content. This asks proxy
# servers not to cache the document. Uncommenting the following line disables
# this behavior, and proxies will be allowed to cache the documents.
#
#CacheNegotiatedDocs

#
# UseCanonicalName:  (new for 1.3)  With this setting turned on, whenever
# Apache needs to construct a self-referencing URL (a URL that refers back
# to the server the response is coming from) it will use ServerName and
# Port to form a "canonical" name.  With this setting off, Apache will
# use the hostname:port that the client supplied, when possible.  This
# also affects SERVER_NAME and SERVER_PORT in CGI scripts.
#
UseCanonicalName On

#
# TypesConfig describes where the mime.types file (or equivalent) is
# to be found.
#
<IfModule mod_mime.c>
    TypesConfig conf/mime.types
</IfModule>

#
# DefaultType is the default MIME type the server will use for a document
# if it cannot otherwise determine one, such as from filename extensions.
# If your server contains mostly text or HTML documents, "text/plain" is
# a good value.  If most of your content is binary, such as applications
# or images, you may want to use "application/octet-stream" instead to
# keep browsers from trying to display binary files as though they are
# text.
#
DefaultType text/plain

#
```

```
# The mod_mime_magic module allows the server to use various hints from the
# contents of the file itself to determine its type.  The MIMEMagicFile
# directive tells the module where the hint definitions are located.
# mod_mime_magic is not part of the default server (you have to add
# it yourself with a LoadModule [see the DSO paragraph in the 'Global
# Environment' section], or recompile the server and include mod_mime_magic
# as part of the configuration), so it's enclosed in an <IfModule> container.
# This means that the MIMEMagicFile directive will only be processed if the
# module is part of the server.
#
<IfModule mod_mime_magic.c>
    MIMEMagicFile conf/magic
</IfModule>

#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
HostnameLookups Off

#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog logs/error.log

#
# LogLevel: Control the number of messages logged to the error.log.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

#
# The location and format of the access logfile (Common Logfile Format).
# If you do not define any access logfiles within a <VirtualHost>
```

```
# container, they will be logged here.  Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles, transactions will be
# logged therein and *not* in this file.
#
CustomLog logs/access.log common

#
# If you would like to have agent and referer logfiles, uncomment the
# following directives.
#
#CustomLog logs/referer.log referer
#CustomLog logs/agent.log agent

#
# If you prefer a single logfile with access, agent, and referer information
# (Combined Logfile Format) you can use the following directive.
#
#CustomLog logs/access.log combined

#
# Optionally add a line containing the server version and virtual host
# name to server-generated pages (error documents, FTP directory listings,
# mod_status and mod_info output etc., but not CGI generated documents).
# Set to "EMail" to also include a mailto: link to the ServerAdmin.
# Set to one of:  On | Off | EMail
#
ServerSignature On

#
# Apache parses all CGI scripts for the shebang line by default.
# This comment line, the first line of the script, consists of the symbols
# pound (#) and exclamation (!) followed by the path of the program that
# can execute this specific script.  For a perl script, with perl.exe in
# the C:\Program Files\Perl directory, the shebang line should be:

  #!c:/program files/perl/perl

# Note you _must_not_ indent the actual shebang line, and it must be the
# first line of the file.  Of course, CGI processing must be enabled by
# the appropriate ScriptAlias or Options ExecCGI directives for the files
# or directory in question.
#
# However, Apache on Windows allows either the Unix behavior above, or can
# use the Registry to match files by extention.  The command to execute
# a file of this type is retrieved from the registry by the same method as
# the Windows Explorer would use to handle double-clicking on a file.
# These script actions can be configured from the Windows Explorer View menu,
# 'Folder Options', and reviewing the 'File Types' tab.  Clicking the Edit
# button allows you to modify the Actions, of which Apache 1.3 attempts to
# perform the 'Open' Action, and failing that it will try the shebang line.
# This behavior is subject to change in Apache release 2.0.
#
```

```
# Each mechanism has it's own specific security weaknesses, from the means
# to run a program you didn't intend the website owner to invoke, and the
# best method is a matter of great debate.
#
# To enable the this Windows specific behavior (and therefore -disable- the
# equivilant Unix behavior), uncomment the following directive:
#
#ScriptInterpreterSource registry
#
# The directive above can be placed in individual <Directory> blocks or the
# .htaccess file, with either the 'registry' (Windows behavior) or 'script'
# (Unix behavior) option, and will override this server default option.
#

#
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
<IfModule mod_alias.c>

    #
    # Note that if you include a trailing / on fakename then the server will
    # require it to be present in the URL.  So "/icons" isn't aliased in this
    # example, only "/icons/".  If the fakename is slash-terminated, then the
    # realname must also be slash terminated, and if the fakename omits the
    # trailing slash, the realname must also omit it.
    #
    Alias /icons/ "C:/Serveur/Apache/icons/"

    <Directory "C:/Serveur/Apache/icons">
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    # This Alias will project the on-line documentation tree under /manual/
    # even if you change the DocumentRoot. Comment it if you don't want to
    # provide access to the on-line documentation.
    #
    Alias /manual/ "C:/Serveur/Apache/htdocs/manual/"

    <Directory "C:/Serveur/Apache/htdocs/manual">
        Options Indexes FollowSymlinks MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    #
    # ScriptAlias: This controls which directories contain server scripts.
    # ScriptAliases are essentially the same as Aliases, except that
```

```
    # documents in the realname directory are treated as applications and
    # run by the server when requested rather than as documents sent to the client.
    # The same rules about trailing "/" apply to ScriptAlias directives as to
    # Alias.
    #
    ScriptAlias /cgi-bin/ "C:/Serveur/Apache/cgi-bin/"

    #
    # "C:/Serveur/Apache/cgi-bin" should be changed to whatever your ScriptAliased
    # CGI directory exists, if you have that configured.
    #
    <Directory "C:/Serveur/Apache/cgi-bin">
        AllowOverride None
        Options None
        Order allow,deny
        Allow from all
    </Directory>

</IfModule>
# End of aliases.

#
# Redirect allows you to tell clients about documents which used to exist in
# your server's namespace, but do not anymore. This allows you to tell the
# clients where to look for the relocated document.
# Format: Redirect old-URI new-URL
#

#
# Directives controlling the display of server-generated directory listings.
#
<IfModule mod_autoindex.c>

    #
    # FancyIndexing is whether you want fancy directory indexing or standard
    #
    # Note, add the option TrackModified to the IndexOptions default list only
    # if all indexed directories reside on NTFS volumes.  The TrackModified flag
    # will report the Last-Modified date to assist caches and proxies to properly
    # track directory changes, but it does _not_ work on FAT volumes.
    #
    IndexOptions FancyIndexing

    #
    # AddIcon* directives tell the server which icon to show for different
    # files or filename extensions.  These are only displayed for
    # FancyIndexed directories.
    #
    AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip

    AddIconByType (TXT,/icons/text.gif) text/*
    AddIconByType (IMG,/icons/image2.gif) image/*
```

```
AddIconByType (SND,/icons/sound2.gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*

AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core

AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^^DIRECTORY^^
AddIcon /icons/blank.gif ^^BLANKICON^^

#
# DefaultIcon is which icon to show for files which do not have an icon
# explicitly set.
#
DefaultIcon /icons/unknown.gif

#
# AddDescription allows you to place a short description after a file in
# server-generated indexes.  These are only displayed for FancyIndexed
# directories.
# Format: AddDescription "description" filename
#
#AddDescription "GZIP compressed document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar archive" .tgz

#
# ReadmeName is the name of the README file the server will look for by
# default, and append to directory listings.
#
# HeaderName is the name of a file which should be prepended to
# directory indexes.
#
ReadmeName README
HeaderName HEADER

#
```

```
    # IndexIgnore is a set of filenames which directory indexing should ignore
    # and not include in the listing.  Shell-style wildcarding is permitted.
    #
    IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t

</IfModule>
# End of indexing directives.

#
# Document types.
#
<IfModule mod_mime.c>

    #
    # AddType allows you to tweak mime.types without actually editing it, or to
    # make certain files to be certain types.
    #
        AddType application/x-tar .tgz
        AddType application/x-httpd-php .php
        AddType application/x-httpd-php .php3
        AddType application/x-httpd-php .phtml

    #
    # AddEncoding allows you to have certain browsers uncompress
    # information on the fly. Note: Not all browsers support this.
    # Despite the name similarity, the following Add* directives have nothing
    # to do with the FancyIndexing customization directives above.
    #
    AddEncoding x-compress .Z
    AddEncoding x-gzip .gz .tgz
    #
    # If the AddEncoding directives above are commented-out, then you
    # probably should define those extensions to indicate media types:
    #
    #AddType application/x-compress .Z
    #AddType application/x-gzip .gz .tgz

        ScriptAlias /php/ c:/serveur/php/
        Action application/x-httpd-php /php/php.exe

    #
    # AddLanguage allows you to specify the language of a document. You can
    # then use content negotiation to give a browser a file in a language
    # it can understand.
    #
    # Note 1: The suffix does not have to be the same as the language
    # keyword --- those with documents in Polish (whose net-standard
    # language code is pl) may wish to use "AddLanguage pl .po" to
    # avoid the ambiguity with the common suffix for perl scripts.
    #
    # Note 2: The example entries below illustrate that in quite
    # some cases the two character 'Language' abbreviation is not
```

```
# identical to the two character 'Country' code for its country,
# E.g. 'Danmark/dk' versus 'Danish/da'.
#
# Note 3: In the case of 'ltz' we violate the RFC by using a three char
# specifier. But there is 'work in progress' to fix this and get
# the reference data for rfc1766 cleaned up.
#
# Danish (da) - Dutch (nl) - English (en) - Estonian (ee)
# French (fr) - German (de) - Greek-Modern (el)
# Italian (it) - Korean (kr) - Norwegian (no) - Norwegian Nynorsk (nn)
# Portugese (pt) - Luxembourgeois* (ltz)
# Spanish (es) - Swedish (sv) - Catalan (ca) - Czech(cs)
# Polish (pl) - Brazilian Portuguese (pt-br) - Japanese (ja)
# Russian (ru)
#
AddLanguage da .dk
AddLanguage nl .nl
AddLanguage en .en
AddLanguage et .ee
AddLanguage fr .fr
AddLanguage de .de
AddLanguage el .el
AddLanguage he .he
AddCharset ISO-8859-8 .iso8859-8
AddLanguage it .it
AddLanguage ja .ja
AddCharset ISO-2022-JP .jis
AddLanguage kr .kr
AddCharset ISO-2022-KR .iso-kr
AddLanguage nn .nn
AddLanguage no .no
AddLanguage pl .po
AddCharset ISO-8859-2 .iso-pl
AddLanguage pt .pt
AddLanguage pt-br .pt-br
AddLanguage ltz .lu
AddLanguage ca .ca
AddLanguage es .es
AddLanguage sv .sv
AddLanguage cs .cz .cs
AddLanguage ru .ru
AddLanguage zh-TW .zh-tw
AddCharset Big5         .Big5    .big5
AddCharset WINDOWS-1251 .cp-1251
AddCharset CP866        .cp866
AddCharset ISO-8859-5   .iso-ru
AddCharset KOI8-R       .koi8-r
AddCharset UCS-2        .ucs2
AddCharset UCS-4        .ucs4
AddCharset UTF-8        .utf8

# LanguagePriority allows you to give precedence to some languages
```

```
    # in case of a tie during content negotiation.
    #
    # Just list the languages in decreasing order of preference. We have
    # more or less alphabetized them here. You probably want to change this.
    #
    <IfModule mod_negotiation.c>
        LanguagePriority en da nl et fr de el it ja kr no pl pt pt-br ru ltz ca es sv tw
    </IfModule>

    #
    # AddHandler allows you to map certain file extensions to "handlers",
    # actions unrelated to filetype. These can be either built into the server
    # or added with the Action command (see below)
    #
    # If you want to use server side includes, or CGI outside
    # ScriptAliased directories, uncomment the following lines.
    #
    # To use CGI scripts:
    #
    #AddHandler cgi-script .cgi

    #
    # To use server-parsed HTML files
    #
    #AddType text/html .shtml
    #AddHandler server-parsed .shtml

    #
    # Uncomment the following line to enable Apache's send-asis HTTP file
    # feature
    #
    #AddHandler send-as-is asis

    #
    # If you wish to use server-parsed imagemap files, use
    #
    #AddHandler imap-file map

    #
    # To enable type maps, you might want to use
    #
    #AddHandler type-map var

</IfModule>
# End of document types.

#
# Action lets you define media types that will execute a script whenever
# a matching file is called. This eliminates the need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-script/location
```

```
#

#
# MetaDir: specifies the name of the directory in which Apache can find
# meta information files. These files contain additional HTTP headers
# to include when sending the document
#
#MetaDir .web

#
# MetaSuffix: specifies the file name suffix for the file containing the
# meta information.
#
#MetaSuffix .meta

#
# Customizable error response (Apache style)
#  these come in three flavors
#
#    1) plain text
#ErrorDocument 500 "The server made a boo boo.
#  n.b.  the single leading (") marks it as text, it does not get output
#
#    2) local redirects
#ErrorDocument 404 /missing.html
#  to redirect to local URL /missing.html
#ErrorDocument 404 /cgi-bin/missing_handler.pl
#  N.B.: You can redirect to a script or a document using server-side-includes.
#
#    3) external redirects
#ErrorDocument 402 http://www.example.com/subscription_info.html
#  N.B.: Many of the environment variables associated with the original
#  request will *not* be available to such a script.

#
# Customize behaviour based on the browser
#
<IfModule mod_setenvif.c>

    #
    # The following directives modify normal HTTP response behavior.
    # The first directive disables keepalive for Netscape 2.x and browsers that
    # spoof it. There are known problems with these browser implementations.
    # The second directive is for Microsoft Internet Explorer 4.0b2
    # which has a broken HTTP/1.1 implementation and does not properly
    # support keepalive when it is used on 301 or 302 (redirect) responses.
    #
    BrowserMatch "Mozilla/2" nokeepalive
    BrowserMatch "MSIE 4\.0b2;" nokeepalive downgrade-1.0 force-response-1.0

    #
    # The following directive disables HTTP/1.1 responses to browsers which
```

```
    # are in violation of the HTTP/1.0 spec by not being able to grok a
    # basic 1.1 response.
    #
    BrowserMatch "RealPlayer 4\.0" force-response-1.0
    BrowserMatch "Java/1\.0" force-response-1.0
    BrowserMatch "JDK/1\.0" force-response-1.0

</IfModule>
# End of browser customization directives

#
# Allow server status reports, with the URL of http://servername/server-status
# Change the "localhost" to match your domain to enable.
#
#<Location /server-status>
#    SetHandler server-status
#    Order deny,allow
#    Deny from all
#    Allow from localhost
#</Location>

#
# Allow remote server configuration reports, with the URL of
# http://servername/server-info (requires that mod_info.c be loaded).
# Change the "localhost" to match your domain to enable.
#
#<Location /server-info>
#    SetHandler server-info
#    Order deny,allow
#    Deny from all
#    Allow from localhost
#</Location>

#
# There have been reports of people trying to abuse an old bug from pre-1.1
# days.  This bug involved a CGI script distributed as a part of Apache.
# By uncommenting these lines you can redirect these attacks to a logging
# script on phf.apache.org.  Or, you can record them yourself, using the script
# support/phf_abuse_log.cgi.
#
#<Location /cgi-bin/phf*>
#    Deny from all
#    ErrorDocument 403 http://phf.apache.org/phf_abuse_log.cgi
#</Location>

### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them. Most configurations
# use only name-based virtual hosts so the server doesn't need to worry about
# IP addresses. This is indicated by the asterisks in the directives below.
#
```

```
# Please see the documentation at <URL:http://www.apache.org/docs/vhosts/>
# for further details before you try to setup virtual hosts.
#
# You may use the command line option '-S' to verify your virtual host
# configuration.

#
# Use name-based virtual hosting.
#
#NameVirtualHost *:80

#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for requests without a known
# server name.
#
#<VirtualHost *:80>
#    ServerAdmin webmaster@dummy-host.example.com
#    DocumentRoot /www/docs/dummy-host.example.com
#    ServerName dummy-host.example.com
#    ErrorLog logs/dummy-host.example.com-error_log
#    CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>


# voir http://www.modssl.org/docs/2.4/ssl_reference.html pour plus d'informations
SSLMutex sem
SSLRandomSeed startup builtin
SSLSessionCache none

SSLLog logs/SSL.log
SSLLogLevel info
# Vous pouvez plus tard changer "info" en "warn" si tout est OK

<VirtualHost 127.0.0.1:443>
SSLEngine On
SSLCertificateFile C:/Serveur/Apache/conf/ssl/xof.cert
SSLCertificateKeyFile C:/Serveur/Apache/conf/ssl/xof.key
</VirtualHost>
```

Le « **php.ini** » ce trouvant à la racine du répertoire de votre système Windows (comme pour le « http.conf », les modifications spécifiées dans ce tutorial se retrouvent en caractère gras de couleurs :

```
[PHP]



;;;;;;;;;;
```

```
; WARNING ;

;;;;;;;;;;

; This is the default settings file for new PHP installations.

; By default, PHP installs itself with a configuration suitable for

; development purposes, and *NOT* for production purposes.

; For several security-oriented considerations that should be taken

; before going online with your site, please consult php.ini-recommended

; and http://php.net/manual/en/security.php.



;;;;;;;;;;;;;;;;;;;

; About this file ;

;;;;;;;;;;;;;;;;;;;

; This file controls many aspects of PHP's behavior.  In order for PHP to

; read it, it must be named 'php.ini'.  PHP looks for it in the current

; working directory, in the path designated by the environment variable

; PHPRC, and in the path that was defined in compile time (in that order).

; Under Windows, the compile-time path is the Windows directory.  The

; path in which the php.ini file is looked for can be overridden using

; the -c argument in command line mode.

;

; The syntax of the file is extremely simple.  Whitespace and Lines

; beginning with a semicolon are silently ignored (as you probably guessed).

; Section headers (e.g. [Foo]) are also silently ignored, even though

; they might mean something in the future.

;
```

```
; Directives are specified using the following syntax:
; directive = value
; Directive names are *case sensitive* - foo=bar is different from FOO=bar.
;
; The value can be a string, a number, a PHP constant (e.g. E_ALL or M_PI), one
; of the INI constants (On, Off, True, False, Yes, No and None) or an expression
; (e.g. E_ALL & ~E_NOTICE), or a quoted string ("foo").
;
; Expressions in the INI file are limited to bitwise operators and parentheses:
; |       bitwise OR
; &        bitwise AND
; ~        bitwise NOT
; !        boolean NOT
;
; Boolean flags can be turned on using the values 1, On, True or Yes.
; They can be turned off using the values 0, Off, False or No.
;
; An empty string can be denoted by simply not writing anything after the equal
; sign, or by using the None keyword:
;
;  foo =         ; sets foo to an empty string
;  foo = none    ; sets foo to an empty string
;  foo = "none"  ; sets foo to the string 'none'
;
; If you use constants in your value, and these constants belong to a
; dynamically loaded extension (either a PHP extension or a Zend extension),
```

```
; you may only use these constants *after* the line that loads the extension.
;
; All the values in the php.ini-dist file correspond to the builtin
; defaults (that is, if no php.ini is used, or if you delete these lines,
; the builtin defaults will be identical).



;;;;;;;;;;;;;;;;;;;
; Language Options ;
;;;;;;;;;;;;;;;;;;;

; Enable the PHP scripting language engine under Apache.
engine = On

; Allow the <? tag.  Otherwise, only <?php and <script> tags are recognized.
; NOTE: Using short tags should be avoided when developing applications or
; libraries that are meant for redistribution, or deployment on PHP
; servers which are not under your control, because short tags may not
; be supported on the target server. For portable, redistributable code,
; be sure not to use short tags.
short_open_tag = On

; Allow ASP-style <% %> tags.
asp_tags = Off

; The number of significant digits displayed in floating point numbers.
```

```
precision    =  12

; Enforce year 2000 compliance (will cause problems with non-compliant browsers)
y2k_compliance = On

; Output buffering allows you to send header lines (including cookies) even
; after you send body content, at the price of slowing PHP's output layer a
; bit.  You can enable output buffering during runtime by calling the output
; buffering functions.  You can also enable output buffering for all files by
; setting this directive to On.  If you wish to limit the size of the buffer
; to a certain size - you can use a maximum number of bytes instead of 'On', as
; a value for this directive (e.g., output_buffering=4096).
output_buffering = Off

; You can redirect all of the output of your scripts to a function.  For
; example, if you set output_handler to "mb_output_handler", character
; encoding will be transparently converted to the specified encoding.
; Setting any output handler automatically turns on output buffering.
; Note: People who wrote portable scripts should not depend on this ini
;       directive. Instead, explicitly set the output handler using ob_start().
;       Using this ini directive may cause problems unless you know what script
;       is doing.
; Note: You cannot use both "mb_output_handler" with "ob_iconv_handler"
;       and you cannot use both "ob_gzhandler" and "zlib.output_compression".
;output_handler =
```

```
; Transparent output compression using the zlib library
; Valid values for this option are 'off', 'on', or a specific buffer size
; to be used for compression (default is 4KB)
; Note: Resulting chunk size may vary due to nature of compression. PHP
;       outputs chunks that are few hundreds bytes each as a result of
;       compression. If you prefer a larger chunk size for better
;       performance, enable output_buffering in addition.
; Note: You need to use zlib.output_handler instead of the standard
;       output_handler, or otherwise the output will be corrupted.
zlib.output_compression = Off

; You cannot specify additional output handlers if zlib.output_compression
; is activated here. This setting does the same as output_handler but in
; a different order.
;zlib.output_handler =

; Implicit flush tells PHP to tell the output layer to flush itself
; automatically after every output block.  This is equivalent to calling the
; PHP function flush() after each and every call to print() or echo() and each
; and every HTML block.  Turning this option on has serious performance
; implications and is generally recommended for debugging purposes only.
implicit_flush = Off

; The unserialize callback function will called (with the undefind class'
; name as parameter), if the unserializer finds an undefined class
; which should be instanciated.
```

```
; A warning appears if the specified function is not defined, or if the
; function doesn't include/implement the missing class.
; So only set this entry, if you really want to implement such a
; callback-function.
unserialize_callback_func=

; When floats & doubles are serialized store serialize_precision significant
; digits after the floating point. The default value ensures that when floats
; are decoded with unserialize, the data will remain the same.
serialize_precision = 100

; Whether to enable the ability to force arguments to be passed by reference
; at function call time.  This method is deprecated and is likely to be
; unsupported in future versions of PHP/Zend.  The encouraged method of
; specifying which arguments should be passed by reference is in the function
; declaration.  You're encouraged to try and turn this option Off and make
; sure your scripts work properly with it in order to ensure they will work
; with future versions of the language (you will receive a warning each time
; you use this feature, and the argument will be passed by value instead of by
; reference).
allow_call_time_pass_reference = On

; Safe Mode
;
safe_mode = Off
```

```
; By default, Safe Mode does a UID compare check when
; opening files. If you want to relax this to a GID compare,
; then turn on safe_mode_gid.
safe_mode_gid = Off

; When safe_mode is on, UID/GID checks are bypassed when
; including files from this directory and its subdirectories.
; (directory must also be in include_path or full path must
; be used when including)
safe_mode_include_dir =

; When safe_mode is on, only executables located in the safe_mode_exec_dir
; will be allowed to be executed via the exec family of functions.
safe_mode_exec_dir =

; Setting certain environment variables may be a potential security breach.
; This directive contains a comma-delimited list of prefixes.  In Safe Mode,
; the user may only alter environment variables whose names begin with the
; prefixes supplied here.  By default, users will only be able to set
; environment variables that begin with PHP_ (e.g. PHP_FOO=BAR).
;
; Note:  If this directive is empty, PHP will let the user modify ANY
; environment variable!
safe_mode_allowed_env_vars = PHP_

; This directive contains a comma-delimited list of environment variables that
```

```
; the end user won't be able to change using putenv().  These variables will be
; protected even if safe_mode_allowed_env_vars is set to allow to change them.
safe_mode_protected_env_vars = LD_LIBRARY_PATH

; open_basedir, if set, limits all file operations to the defined directory
; and below.  This directive makes most sense if used in a per-directory
; or per-virtualhost web server configuration file. This directive is
; *NOT* affected by whether Safe Mode is turned On or Off.
;open_basedir =

; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names. This directive is
; *NOT* affected by whether Safe Mode is turned On or Off.
disable_functions =

; This directive allows you to disable certain classes for security reasons.
; It receives a comma-delimited list of class names. This directive is
; *NOT* affected by whether Safe Mode is turned On or Off.
disable_classes =

; Colors for Syntax Highlighting mode.  Anything that's acceptable in
; <font color="??????"> would work.
;highlight.string  = #DD0000
;highlight.comment = #FF9900
;highlight.keyword = #007700
;highlight.bg      = #FFFFFF
```

```
;highlight.default = #0000BB

;highlight.html    = #000000




;

; Misc

;

; Decides whether PHP may expose the fact that it is installed on the server

; (e.g. by adding its signature to the Web server header).  It is no security

; threat in any way, but it makes it possible to determine whether you use PHP

; on your server or not.

expose_php = On




;;;;;;;;;;;;;;;;;;;

; Resource Limits ;

;;;;;;;;;;;;;;;;;;;


max_execution_time = 30     ; Maximum execution time of each script, in seconds

max_input_time = 60 ; Maximum amount of time each script may spend parsing request data

memory_limit = 8M      ; Maximum amount of memory a script may consume (8MB)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; Error handling and logging ;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
; error_reporting is a bit-field.  Or each number up to get desired error
; reporting level
; E_ALL             - All errors and warnings
; E_ERROR           - fatal run-time errors
; E_WARNING         - run-time warnings (non-fatal errors)
; E_PARSE           - compile-time parse errors
; E_NOTICE          - run-time notices (these are warnings which often result
;                     from a bug in your code, but it's possible that it was
;                     intentional (e.g., using an uninitialized variable and
;                     relying on the fact it's automatically initialized to an
;                     empty string)
; E_CORE_ERROR      - fatal errors that occur during PHP's initial startup
; E_CORE_WARNING    - warnings (non-fatal errors) that occur during PHP's
;                     initial startup
; E_COMPILE_ERROR   - fatal compile-time errors
; E_COMPILE_WARNING - compile-time warnings (non-fatal errors)
; E_USER_ERROR      - user-generated error message
; E_USER_WARNING    - user-generated warning message
; E_USER_NOTICE     - user-generated notice message
;
; Examples:
;
;   - Show all errors, except for notices
;
;error_reporting = E_ALL & ~E_NOTICE
;
```

```
;   - Show only errors
;
;error_reporting = E_COMPILE_ERROR|E_ERROR|E_CORE_ERROR
;
;   - Show all errors except for notices
;
error_reporting  =  E_ALL & ~E_NOTICE


; Print out errors (as a part of the output).  For production web sites,
; you're strongly encouraged to turn this feature off, and use error logging
; instead (see below).  Keeping display_errors enabled on a production web site
; may reveal security information to end users, such as file paths on your Web
; server, your database schema or other information.
display_errors = On


; Even when display_errors is on, errors that occur during PHP's startup
; sequence are not displayed.  It's strongly recommended to keep
; display_startup_errors off, except for when debugging.
display_startup_errors = Off


; Log errors into a log file (server-specific log, stderr, or error_log (below))
; As stated above, you're strongly advised to use error logging in place of
; error displaying on production web sites.
log_errors = Off


; Set maximum length of log_errors. In error_log information about the source is
```

```
; added. The default is 1024 and 0 allows to not apply any maximum length at all.
log_errors_max_len = 1024

; Do not log repeated messages. Repeated errors must occur in same file on same
; line until ignore_repeated_source is set true.
ignore_repeated_errors = Off

; Ignore source of message when ignoring repeated messages. When this setting
; is On you will not log errors with repeated messages from different files or
; sourcelines.
ignore_repeated_source = Off

; If this parameter is set to Off, then memory leaks will not be shown (on
; stdout or in the log). This has only effect in a debug compile, and if
; error reporting includes E_WARNING in the allowed list
report_memleaks = On

; Store the last error/warning message in $php_errormsg (boolean).
track_errors = Off

; Disable the inclusion of HTML tags in error messages.
;html_errors = Off

; If html_errors is set On PHP produces clickable error messages that direct
; to a page describing the error or function causing the error in detail.
; You can download a copy of the PHP manual from http://www.php.net/docs.php
```

```
; and change docref_root to the base URL of your local copy including the
; leading '/'. You must also specify the file extension being used including
; the dot.
;docref_root = "/phpmanual/"
;docref_ext = .html

; String to output before an error message.
;error_prepend_string = "<font color=ff0000>"

; String to output after an error message.
;error_append_string = "</font>"

; Log errors to specified file.
;error_log = filename

; Log errors to syslog (Event Log on NT, not valid in Windows 95).
;error_log = syslog



;;;;;;;;;;;;;;;;;
; Data Handling ;
;;;;;;;;;;;;;;;;;
;
; Note - track_vars is ALWAYS enabled as of PHP 4.0.3

; The separator used in PHP generated URLs to separate arguments.
```

```
; Default is "&".
;arg_separator.output = "&amp;"

; List of separator(s) used by PHP to parse input URLs into variables.
; Default is "&".
; NOTE: Every character in this directive is considered as separator!
;arg_separator.input = ";&"

; This directive describes the order in which PHP registers GET, POST, Cookie,
; Environment and Built-in variables (G, P, C, E & S respectively, often
; referred to as EGPCS or GPC).  Registration is done from left to right, newer
; values override older values.
variables_order = "EGPCS"

; Whether or not to register the EGPCS variables as global variables.  You may
; want to turn this off if you don't want to clutter your scripts' global scope
; with user data.  This makes most sense when coupled with track_vars - in which
; case you can access all of the GPC variables through the $HTTP_*_VARS[],
; variables.
;
; You should do your best to write your scripts so that they do not require
; register_globals to be on;  Using form variables as globals can easily lead
; to possible security problems, if the code is not very well thought of.
register_globals = Off

; This directive tells PHP whether to declare the argv&argc variables (that
```

```
; would contain the GET information).  If you don't use these variables, you

; should turn it off for increased performance.

register_argc_argv = On


; Maximum size of POST data that PHP will accept.

post_max_size = 8M


; This directive is deprecated.  Use variables_order instead.

gpc_order = "GPC"


; Magic quotes

;


; Magic quotes for incoming GET/POST/Cookie data.

magic_quotes_gpc = On


; Magic quotes for runtime-generated data, e.g. data from SQL, from exec(), etc.

magic_quotes_runtime = Off


; Use Sybase-style magic quotes (escape ' with " instead of \').

magic_quotes_sybase = Off


; Automatically add files before or after any PHP document.

auto_prepend_file =

auto_append_file =
```

```
; As of 4.0b4, PHP always outputs a character encoding by default in
; the Content-type: header.  To disable sending of the charset, simply
; set it to be empty.
;
; PHP's built-in default is text/html
default_mimetype = "text/html"
;default_charset = "iso-8859-1"

; Always populate the $HTTP_RAW_POST_DATA variable.
;always_populate_raw_post_data = On



;;;;;;;;;;;;;;;;;;;;;;;;;
; Paths and Directories ;
;;;;;;;;;;;;;;;;;;;;;;;;;

; UNIX: "/path1:/path2"
;include_path = ".:/php/includes"
;
; Windows: "\path1;\path2"
include_path = .

; The root of the PHP pages, used only if nonempty.
; if PHP was not compiled with FORCE_REDIRECT, you SHOULD set doc_root
; if you are running php as a CGI under any web server (other than IIS)
; see documentation for security issues.  The alternate is to use the
```

; cgi.force_redirect configuration below

doc_root =

; The directory under which PHP opens the script using /~username used only

; if nonempty.

user_dir =

; Directory in which the loadable extensions (modules) reside.

extension_dir = C:\Serveur\php\extensions

; Whether or not to enable the dl() function. The dl() function does NOT work

; properly in multithreaded servers, such as IIS or Zeus, and is automatically

; disabled on them.

enable_dl = On

; cgi.force_redirect is necessary to provide security running PHP as a CGI under

; most web servers. Left undefined, PHP turns this on by default. You can

; turn it off here AT YOUR OWN RISK

; **You CAN safely turn this off for IIS, in fact, you MUST.**

; cgi.force_redirect = 1

; if cgi.force_redirect is turned on, and you are not running under Apache or Netscape

; (iPlanet) web servers, you MAY need to set an environment variable name that PHP

; will look for to know it is OK to continue execution. Setting this variable MAY

; cause security issues, KNOW WHAT YOU ARE DOING FIRST.

; cgi.redirect_status_env = ;

; cgi.fix_pathinfo provides *real* PATH_INFO/PATH_TRANSLATED support for CGI. PHP's

; previous behaviour was to set PATH_TRANSLATED to SCRIPT_FILENAME, and to not grok

; what PATH_INFO is. For more information on PATH_INFO, see the cgi specs. Setting

; this to 1 will cause PHP CGI to fix it's paths to conform to the spec. A setting

; of zero causes PHP to behave as before. Default is zero. You should fix your scripts

; to use SCRIPT_FILENAME rather than PATH_TRANSLATED.

; cgi.fix_pathinfo=0

; FastCGI under IIS (on WINNT based OS) supports the ability to impersonate

; security tokens of the calling client. This allows IIS to define the

; security context that the request runs under. mod_fastcgi under Apache

; does not currently support this feature (03/17/2002)

; Set to 1 if running under IIS. Default is zero.

; fastcgi.impersonate = 1;

; cgi.rfc2616_headers configuration option tells PHP what type of headers to

; use when sending HTTP response code. If it's set 0 PHP sends Status: header that

; is supported by Apache. When this option is set to 1 PHP will send

; RFC2616 compliant header.

; Default is zero.

;cgi.rfc2616_headers = 0

;;;;;;;;;;;;;;;;

; File Uploads ;

;;;;;;;;;;;;;;;;

; Whether to allow HTTP file uploads.

file_uploads = On

; Temporary directory for HTTP uploaded files (will use system default if not

; specified).

;upload_tmp_dir =

; Maximum allowed size for uploaded files.

upload_max_filesize = 2097152

;;;;;;;;;;;;;;;;;;

; Fopen wrappers ;

;;;;;;;;;;;;;;;;;;

; Whether to allow the treatment of URLs (like http:// or ftp://) as files.

allow_url_fopen = On

; Define the anonymous ftp password (your email address)

;from="john@doe.com"

; Define the User-Agent string

; user_agent="PHP"

```
; Default timeout for socket based streams (seconds)
default_socket_timeout = 60


; If your scripts have to deal with files from Macintosh systems,
; or you are running on a Mac and need to deal with files from
; unix or win32 systems, setting this flag will cause PHP to
; automatically detect the EOL character in those files so that
; fgets() and file() will work regardless of the source of the file.
; auto_detect_line_endings = Off



;;;;;;;;;;;;;;;;;;;;
; Dynamic Extensions ;
;;;;;;;;;;;;;;;;;;;;
;
; If you wish to have an extension loaded automatically, use the following
; syntax:
;
;   extension=modulename.extension
;
; For example, on Windows:
;
;   extension=msql.dll
;
; ... or under UNIX:
;
```

```
;   extension=msql.so

;

; Note that it should be the name of the module only; no directory information

; needs to go here.  Specify the location of the extension with the

; extension_dir directive above.



;Windows Extensions

;Note that MySQL and ODBC support is now built in, so no dll is needed for it.

;

;extension=php_bz2.dll

;extension=php_cpdf.dll

;extension=php_crack.dll

;extension=php_curl.dll

;extension=php_db.dll

;extension=php_dba.dll

;extension=php_dbase.dll

;extension=php_dbx.dll

;extension=php_domxml.dll

;extension=php_exif.dll

;extension=php_fdf.dll

;extension=php_filepro.dll

;extension=php_gd2.dll

;extension=php_gettext.dll

;extension=php_hyperwave.dll

;extension=php_iconv.dll
```

```
;extension=php_ifx.dll
;extension=php_iisfunc.dll
;extension=php_imap.dll
;extension=php_interbase.dll
;extension=php_java.dll
;extension=php_ldap.dll
;extension=php_mbstring.dll
;extension=php_mcrypt.dll
;extension=php_mhash.dll
;extension=php_mime_magic.dll
;extension=php_ming.dll
;extension=php_mssql.dll
;extension=php_msql.dll
;extension=php_oci8.dll
extension=php_openssl.dll
;extension=php_oracle.dll
;extension=php_pdf.dll
;extension=php_pgsql.dll
;extension=php_printer.dll
;extension=php_shmop.dll
;extension=php_snmp.dll
;extension=php_sockets.dll
;extension=php_sybase_ct.dll
;extension=php_w32api.dll
;extension=php_xmlrpc.dll
;extension=php_xslt.dll
```

```
;extension=php_yaz.dll

;extension=php_zip.dll



;;;;;;;;;;;;;;;;;;;

; Module Settings ;

;;;;;;;;;;;;;;;;;;;


[Syslog]

; Whether or not to define the various syslog variables (e.g. $LOG_PID,

; $LOG_CRON, etc.).  Turning it off is a good idea performance-wise.  In

; runtime, you can define these variables by calling define_syslog_variables().

define_syslog_variables  = Off


[mail function]

; For Win32 only.

SMTP = localhost


; For Win32 only.

;sendmail_from = me@example.com


; For Unix only.  You may supply arguments as well (default: "sendmail -t -i").

;sendmail_path =


[Java]

;java.class.path = .\php_java.jar
```

```
;java.home = c:\jdk

;java.library = c:\jdk\jre\bin\hotspot\jvm.dll

;java.library.path = .\


[SQL]

sql.safe_mode = Off


[ODBC]

;odbc.default_db    =  Not yet implemented

;odbc.default_user  =  Not yet implemented

;odbc.default_pw    =  Not yet implemented


; Allow or prevent persistent links.

odbc.allow_persistent = On


; Check that a connection is still valid before reuse.

odbc.check_persistent = On


; Maximum number of persistent links.  -1 means no limit.

odbc.max_persistent = -1


; Maximum number of links (persistent + non-persistent).  -1 means no limit.

odbc.max_links = -1


; Handling of LONG fields.  Returns number of bytes to variables.  0 means

; passthru.
```

```
odbc.defaultlrl = 4096


; Handling of binary data.  0 means passthru, 1 return as is, 2 convert to char.
; See the documentation on odbc_binmode and odbc_longreadlen for an explanation
; of uodbc.defaultlrl and uodbc.defaultbinmode
odbc.defaultbinmode = 1


[MySQL]
; Allow or prevent persistent links.
mysql.allow_persistent = On


; Maximum number of persistent links.  -1 means no limit.
mysql.max_persistent = -1


; Maximum number of links (persistent + non-persistent).  -1 means no limit.
mysql.max_links = -1


; Default port number for mysql_connect().  If unset, mysql_connect() will use
; the $MYSQL_TCP_PORT or the mysql-tcp entry in /etc/services or the
; compile-time value defined MYSQL_PORT (in that order).  Win32 will only look
; at MYSQL_PORT.
mysql.default_port =


; Default socket name for local MySQL connects.  If empty, uses the built-in
; MySQL defaults.
mysql.default_socket =
```

```
; Default host for mysql_connect() (doesn't apply in safe mode).
mysql.default_host =

; Default user for mysql_connect() (doesn't apply in safe mode).
mysql.default_user =

; Default password for mysql_connect() (doesn't apply in safe mode).
; Note that this is generally a *bad* idea to store passwords in this file.
; *Any* user with PHP access can run 'echo get_cfg_var("mysql.default_password")
; and reveal this password!  And of course, any users with read access to this
; file will be able to reveal the password as well.
mysql.default_password =

; Maximum time (in secondes) for connect timeout. -1 means no limimt
mysql.connect_timeout = 60

; Trace mode. When trace_mode is active (=On), warnings for table/index scans and
; SQL-Erros will be displayed.
mysql.trace_mode = Off

[mSQL]
; Allow or prevent persistent links.
msql.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
```

```
msql.max_persistent = -1

; Maximum number of links (persistent+non persistent).  -1 means no limit.
msql.max_links = -1

[PostgresSQL]
; Allow or prevent persistent links.
pgsql.allow_persistent = On

; Detect broken persistent links always with pg_pconnect(). Need a little overhead.
pgsql.auto_reset_persistent = Off

; Maximum number of persistent links.  -1 means no limit.
pgsql.max_persistent = -1

; Maximum number of links (persistent+non persistent).  -1 means no limit.
pgsql.max_links = -1

; Ignore PostgreSQL backends Notice message or not.
pgsql.ignore_notice = 0

; Log PostgreSQL backends Noitce message or not.
; Unless pgsql.ignore_notice=0, module cannot log notice message.
pgsql.log_notice = 0

[Sybase]
```

```
; Allow or prevent persistent links.
sybase.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
sybase.max_persistent = -1

; Maximum number of links (persistent + non-persistent).  -1 means no limit.
sybase.max_links = -1

;sybase.interface_file = "/usr/sybase/interfaces"

; Minimum error severity to display.
sybase.min_error_severity = 10

; Minimum message severity to display.
sybase.min_message_severity = 10

; Compatability mode with old versions of PHP 3.0.
; If on, this will cause PHP to automatically assign types to results according
; to their Sybase type, instead of treating them all as strings.  This
; compatability mode will probably not stay around forever, so try applying
; whatever necessary changes to your code, and turn it off.
sybase.compatability_mode = Off

[Sybase-CT]
; Allow or prevent persistent links.
```

```
sybct.allow_persistent = On


; Maximum number of persistent links.  -1 means no limit.

sybct.max_persistent = -1


; Maximum number of links (persistent + non-persistent).  -1 means no limit.

sybct.max_links = -1


; Minimum server message severity to display.

sybct.min_server_severity = 10


; Minimum client message severity to display.

sybct.min_client_severity = 10


[dbx]

; returned column names can be converted for compatibility reasons

; possible values for dbx.colnames_case are

; "unchanged" (default, if not set)

; "lowercase"

; "uppercase"

; the recommended default is either upper- or lowercase, but

; unchanged is currently set for backwards compatibility

dbx.colnames_case = "unchanged"


[bcmath]

; Number of decimal digits for all bcmath functions.
```

```
bcmath.scale = 0


[browscap]

;browscap = extra/browscap.ini


[Informix]

; Default host for ifx_connect() (doesn't apply in safe mode).

ifx.default_host =


; Default user for ifx_connect() (doesn't apply in safe mode).

ifx.default_user =


; Default password for ifx_connect() (doesn't apply in safe mode).

ifx.default_password =


; Allow or prevent persistent links.

ifx.allow_persistent = On


; Maximum number of persistent links.  -1 means no limit.

ifx.max_persistent = -1


; Maximum number of links (persistent + non-persistent).  -1 means no limit.

ifx.max_links = -1


; If on, select statements return the contents of a text blob instead of its id.

ifx.textasvarchar = 0
```

```
; If on, select statements return the contents of a byte blob instead of its id.
ifx.byteasvarchar = 0

; Trailing blanks are stripped from fixed-length char columns.  May help the
; life of Informix SE users.
ifx.charasvarchar = 0

; If on, the contents of text and byte blobs are dumped to a file instead of
; keeping them in memory.
ifx.blobinfile = 0

; NULL's are returned as empty strings, unless this is set to 1.  In that case,
; NULL's are returned as string 'NULL'.
ifx.nullformat = 0

[Session]
; Handler used to store/retrieve data.
session.save_handler = files

; Argument passed to save_handler.  In the case of files, this is the path
; where data files are stored. Note: Windows users have to change this
; variable in order to use PHP's session functions.
; As of PHP 4.0.1, you can define the path as:
;     session.save_path = "N;/path"
; where N is an integer.  Instead of storing all the session files in
```

```
; /path, what this will do is use subdirectories N-levels deep, and
; store the session data in those directories.  This is useful if you
; or your OS have problems with lots of files in one directory, and is
; a more efficient layout for servers that handle lots of sessions.
; NOTE 1: PHP will not create this directory structure automatically.
;         You can use the script in the ext/session dir for that purpose.
; NOTE 2: See the section on garbage collection below if you choose to
;         use subdirectories for session storage
session.save_path = /tmp

; Whether to use cookies.
session.use_cookies = 1

; This option enables administrators to make their users invulnerable to
; attacks which involve passing session ids in URLs; defaults to 0.
; session.use_only_cookies = 1

; Name of the session (used as cookie name).
session.name = PHPSESSID

; Initialize session on request startup.
session.auto_start = 0

; Lifetime in seconds of cookie or, if 0, until browser is restarted.
session.cookie_lifetime = 0
```

```
; The path for which the cookie is valid.
session.cookie_path = /

; The domain for which the cookie is valid.
session.cookie_domain =

; Handler used to serialize data.  php is the standard serializer of PHP.
session.serialize_handler = php

; Define the probability that the 'garbage collection' process is started
; on every session initialization.
; The probability is calculated by using gc_probability/gc_divisor,
; e.g. 1/100 means there is a 1% chance that the GC process starts
; on each request.

session.gc_probability = 1
session.gc_divisor     = 100

; After this number of seconds, stored data will be seen as 'garbage' and
; cleaned up by the garbage collection process.
session.gc_maxlifetime = 1440

; NOTE: If you are using the subdirectory option for storing session files
;       (see session.save_path above), then garbage collection does *not*
;       happen automatically.  You will need to do your own garbage
;       collection through a shell script, cron entry, or some other method.
```

```
;       For example, the following script would is the equivalent of
;       setting session.gc_maxlifetime to 1440 (1440 seconds = 24 minutes):
;          cd /path/to/sessions; find -cmin +24 | xargs rm

; PHP 4.2 and less have an undocumented feature/bug that allows you to
; to initialize a session variable in the global scope, albeit register_globals
; is disabled.  PHP 4.3 and later will warn you, if this feature is used.
; You can disable the feature and the warning seperately. At this time,
; the warning is only displayed, if bug_compat_42 is enabled.

session.bug_compat_42 = 1
session.bug_compat_warn = 1

; Check HTTP Referer to invalidate externally stored URLs containing ids.
; HTTP_REFERER has to contain this substring for the session to be
; considered as valid.
session.referer_check =

; How many bytes to read from the file.
session.entropy_length = 0

; Specified here to create the session id.
session.entropy_file =

;session.entropy_length = 16
```

```
;session.entropy_file = /dev/urandom


; Set to {nocache,private,public,} to determine HTTP caching aspects
; or leave this empty to avoid sending anti-caching headers.
session.cache_limiter = nocache


; Document expires after n minutes.
session.cache_expire = 180


; trans sid support is disabled by default.
; Use of trans sid may risk your users security.
; Use this option with caution.
; - User may send URL contains active session ID
;   to other person via. email/irc/etc.
; - URL that contains active session ID may be stored
;   in publically accessible computer.
; - User may access your site with the same session ID
;   always using URL stored in browser's history or bookmarks.
session.use_trans_sid = 0


; The URL rewriter will look for URLs in a defined set of HTML tags.
; form/fieldset are special; if you include them here, the rewriter will
; add a hidden <input> field with the info which is otherwise appended
; to URLs.  If you want XHTML conformity, remove the form entry.
; Note that all valid entries require a "=", even if no value follows.
url_rewriter.tags = "a=href,area=href,frame=src,input=src,form=,fieldset="
```

```
[MSSQL]
; Allow or prevent persistent links.
mssql.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
mssql.max_persistent = -1

; Maximum number of links (persistent+non persistent).  -1 means no limit.
mssql.max_links = -1

; Minimum error severity to display.
mssql.min_error_severity = 10

; Minimum message severity to display.
mssql.min_message_severity = 10

; Compatability mode with old versions of PHP 3.0.
mssql.compatability_mode = Off

; Connec timeout
;mssql.connect_timeout = 5

; Query timeout
;mssql.timeout = 60
```

```
; Valid range 0 - 2147483647.  Default = 4096.

;mssql.textlimit = 4096


; Valid range 0 - 2147483647.  Default = 4096.

;mssql.textsize = 4096


; Limits the number of records in each batch.  0 = all records in one batch.

;mssql.batchsize = 0


; Specify how datetime and datetim4 columns are returned

; On => Returns data converted to SQL server settings

; Off => Returns values as YYYY-MM-DD hh:mm:ss

;mssql.datetimeconvert = On


; Use NT authentication when connecting to the server

mssql.secure_connection = Off


; Specify max number of processes. Default = 25

;mssql.max_procs = 25


[Assertion]

; Assert(expr); active by default.

;assert.active = On


; Issue a PHP warning for each failed assertion.

;assert.warning = On
```

```
; Don't bail out by default.
;assert.bail = Off

; User-function to be called if an assertion fails.
;assert.callback = 0

; Eval the expression with current error_reporting().  Set to true if you want
; error_reporting(0) around the eval().
;assert.quiet_eval = 0

[Ingres II]
; Allow or prevent persistent links.
ingres.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
ingres.max_persistent = -1

; Maximum number of links, including persistents.  -1 means no limit.
ingres.max_links = -1

; Default database (format: [node_id::]dbname[/srv_class]).
ingres.default_database =

; Default user.
ingres.default_user =
```

```
; Default password.
ingres.default_password =

[Verisign Payflow Pro]
; Default Payflow Pro server.
pfpro.defaulthost = "test-payflow.verisign.com"

; Default port to connect to.
pfpro.defaultport = 443

; Default timeout in seconds.
pfpro.defaulttimeout = 30

; Default proxy IP address (if required).
;pfpro.proxyaddress =

; Default proxy port.
;pfpro.proxyport =

; Default proxy logon.
;pfpro.proxylogon =

; Default proxy password.
;pfpro.proxypassword =
```

```
[Sockets]
; Use the system read() function instead of the php_read() wrapper.
sockets.use_system_read = On

[com]
; path to a file containing GUIDs, IIDs or filenames of files with TypeLibs
;com.typelib_file =
; allow Distributed-COM calls
;com.allow_dcom = true
; autoregister constants of a components typlib on com_load()
;com.autoregister_typelib = true
; register constants casesensitive
;com.autoregister_casesensitive = false
; show warnings on duplicate constat registrations
;com.autoregister_verbose = true

[Printer]
;printer.default_printer = ""

[mbstring]
; language for internal character representation.
;mbstring.language = Japanese

; internal/script encoding.
; Some encoding cannot work as internal encoding.
; (e.g. SJIS, BIG5, ISO-2022-*)
```

```
;mbstring.internal_encoding = EUC-JP


; http input encoding.
;mbstring.http_input = auto


; http output encoding. mb_output_handler must be
; registered as output buffer to function
;mbstring.http_output = SJIS


; enable automatic encoding translation accoding to
; mbstring.internal_encoding setting. Input chars are
; converted to internal encoding by setting this to On.
; Note: Do _not_ use automatic encoding translation for
;       portable libs/applications.
;mbstring.encoding_translation = Off


; automatic encoding detection order.
; auto means
;mbstring.detect_order = auto


; substitute_character used when character cannot be converted
; one from another
;mbstring.substitute_character = none;


; overload(replace) single byte functions by mbstring functions.
; mail(), ereg(), etc are overloaded by mb_send_mail(), mb_ereg(),
```

```
; etc. Possible values are 0,1,2,4 or combination of them.
; For example, 7 for overload everything.
; 0: No overload
; 1: Overload mail() function
; 2: Overload str*() functions
; 4: Overload ereg*() functions
;mbstring.func_overload = 0

[FrontBase]
;fbsql.allow_persistent = On
;fbsql.autocommit = On
;fbsql.default_database =
;fbsql.default_database_password =
;fbsql.default_host =
;fbsql.default_password =
;fbsql.default_user = "_SYSTEM"
;fbsql.generate_warnings = Off
;fbsql.max_connections = 128
;fbsql.max_links = 128
;fbsql.max_persistent = -1
;fbsql.max_results = 128
;fbsql.batchSize = 1000

[Crack]
; Modify the setting below to match the directory location of the cracklib
; dictionary files.  Include the base filename, but not the file extension.
```

```
; crack.default_dictionary = "c:\php\lib\cracklib_dict"


[exif]

; Exif UNICODE user comments are handled as UCS-2BE/UCS-2LE and JIS as JIS.

; With mbstring support this will automatically be converted into the encoding

; given by corresponding encode setting. When empty mbstring.internal_encoding

; is used. For the decode settings you can distinguish between motorola and

; intel byte order. A decode setting cannot be empty.

;exif.encode_unicode = ISO-8859-15

;exif.decode_unicode_motorola = UCS-2BE

;exif.decode_unicode_intel    = UCS-2LE

;exif.encode_jis =

;exif.decode_jis_motorola = JIS

;exif.decode_jis_intel    = JIS


; Local Variables:

; tab-width: 4

; End:
```

Voila c'est terminé vous pouvez utilisez le protocole « **https** » et les fonctions « **OpenSSL** » de PHP. Cependant votre certificat créé ne provient pas d'une autorité de confiance.

L'obtention de tels certificats est soumise à un contrôle strict de l'identité du demandeur aux autorités concernées (La principale autorité mondiale Verisign).

Vous devrez fournir un document administratif certifiant l'existence de votre société ou organisation selon les cas suivants :

- Pour les sociétés (SARL, SA, etc.), entreprise individuelle, GIE, etc. Extrait K-BIS du greffe du Registre du Commerce de moins de 3 mois
- Profession libérale, Administration, Secteur public Extrait de "situation au répertoire SIRENE" de l'INSEE
- Association Récépissé de déclaration d'association (comportant le numéro d'enregistrement en préfecture)

- Individu (nom personnel) Extrait d'acte de naissance original + copie de facture France Telecom d'une ligne fixe ou facture EDF + chèque annulé à votre nom ou relevé d'identité bancaire.

VI – Post face

Un grand merci, à PHPFrance.com, et a J.F. Moreau pour leurs tutoriaux, qui m'ont grandement inspiré à faire celui-ci…

Par **:: XzoF::**