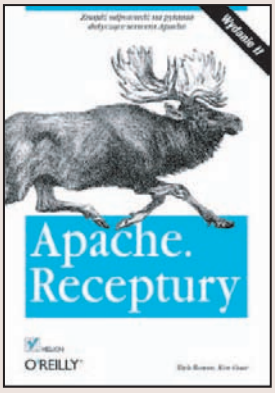
Apache. Receptury. Wydanie II



Autor: Rich Bowen, Ken Coar
ISBN: 978-83-246-1549-0
Tytuł oryginału: Apache Cookbook: Solutions and Examples for Apache Administration (Cookbook)
Format: 168x237, stron: 328

Czy wiesz, jaki serwer HTTP jest najpopularniejszy w sieci?

Właśnie tak, jest to Apache! W lipcu 2008 roku jego udział w rynku wynosił blisko 50% (według Netcraft). Historia tego serwera sięga roku 1995, kiedy ukazała się jego pierwsza oficjalna wersja, oznaczona numerem 0.6.2. Cechy, które zadecydowały o sukcesie tego rozwiązania, to bezpieczeństwo, skalowalność, wielowątkowość i obsługa różnorodnych języków skryptowych.

Dzięki książce „Apache. Receptury" zapoznasz się z gotowymi przepisami na rozwiązanie ciekawych, specyficznych oraz intrygujących problemów. Nauczysz się instalować serwer z różnych źródeł oraz na różnych platformach. Dowiesz się, w jaki sposób zwiększyć jego bezpieczeństwo, jak uruchomić serwery wirtualne oraz poprawić wydajność Apache. Autorzy książki pokażą Ci, jak uruchomić obsługę języków skryptowych, tak aby serwowane strony stały się dynamiczne. Cała wiedza zostanie przedstawiona w sprawdzony w tej serii sposób: problem – rozwiązanie – analiza.

- Sposoby instalacji serwera Apache
- Dodawanie funkcjonalności dzięki modułom
- Możliwości rejestracji zdarzeń
- Konfiguracja serwerów wirtualnych
- Wykorzystanie aliasów, przekierowań oraz przepisań (mod_rewrite)
- Zarządzanie dostępem do serwowanych zasobów
- Bezpieczeństwo serwera Apache
- Wykorzystanie szyfrowanej transmisji - protokół SSL
- Zapewnienie wydajności
- Wykorzystanie języków skryptowychOto książka z najlepszymi przepisami na Apache!

Wydawnictwo Helion
ul. Kościuszki 1c
44-100 Gliwice
tel. 032 230 98 63
e-mail: helion@helion.pl

Spis treści

ROZDZIAŁ 2.

Instalacja modułów

W pakiecie podstawowej dystrybucji serwera WWW Apache nie ma bardzo wielu popularnych modułów. Większość z nich nie trafiła do dystrybucji podstawowej z powodów licencyjnych lub z powodów związanych z obsługą techniczną, inne nie są dystrybuowane przez Apache Software Foundation, gdyż taką decyzję podjęli twórcy serwera, a jeszcze inne są integralną częścią innych projektów. Na przykład moduł *mod_ssl* dla serwera Apache 1.3 jest tworzony i rozwijany oddzielnie, nie tylko z powodu amerykańskich ograniczeń eksportowych (które były znacznie bardziej rygorystyczne w czasie, gdy pakiet powstawał), ale głównie dlatego, że wymaga on wprowadzenia zmian do jądra serwera, na które nie zdecydowali się jego twórcy.

W niniejszym rozdziale przedstawiono receptury omawiające instalację niektórych najbardziej popularnych modułów pochodzących spoza oficjalnej dystrybucji. Jeżeli zachodzi taka potrzeba, przedstawione są osobne receptury omawiające instalację modułów w systemach uniksowych i instalację w systemie Windows.

Najbardziej kompletna lista modułów innych producentów znajduje się na stronie Apache Module Registry pod adresem *http://modules.apache.org/*. Niektóre moduły są tak popularne lub są tak złożone, że poświęcono im całe strony internetowe. Tak jest na przykład w przypadku modułów omówionych w tym rozdziale.

Wielu twórców modułów zajmuje się tworzeniem tylko jednego modułu. Oznacza to, że potencjalnie może istnieć tyle sposobów instalacji modułów, ile jest samych modułów. Pierwsza receptura tego rozdziału opisuje proces instalacji, który powinien być odpowiedni dla wielu modułów wersji 1.3 serwera Apache. Jednak w przypadku każdego modułu, należy sprawdzić w jego dokumentacji, czy jego instalacja nie przebiega w inny sposób.

Wiele z tych modułów można otrzymać od firm, które tworzą pakiety i dystrybuują oprogramowanie Apache — na przykład w postaci modułów RPM firm Mandrake czy Red Hat. Jednak takie pakiety budowane są przy pewnych założeniach poczynionych przez tworzącą je firmę. Mówiąc inaczej — jeżeli serwer został zbudowany z plików źródłowych, a jego pliki znajdują się w miejscach innych niż standardowe, nie należy się dziwić, gdy instalacja jakiegoś modułu się nie powiedzie.

Wszystkie moduły opisane w tym rozdziale są obsługiwane przez wersję 1.3 serwera Apache działającą w systemach uniksowych. Informacje na temat modułów dla wersji 2.0 serwera oraz w przypadku systemu Windows zebrano w tabeli 2.1.

Tabela 2.1. Obsługa modułów w systemie Windows oraz przez wersję 2.0 serwera Apache

Nazwa modułu	Obsługa w systemie Windows	Obsługa przez serwer Apache 2.0
`mod_dav`	Tak	Moduł dołączony do serwera — nie ma potrzeby instalowania.
`mod_perl`	Tak	Tak
`mod_php`	Tak	Tak
`mod_ssl`	Nie	Moduł dołączony do serwera — nie ma potrzeby instalowania.

2.1. Instalacja typowego modułu

Problem

Należy zainstalować posiadany moduł, którego instalacja nie została omówiona osobno w tym rozdziale.

Rozwiązanie

W katalogu, w którym znajduje się plik źródłowy modułu, należy wydać polecenie:

```
% /ścieżka/do/serwera/apache/bin/apxs -cia moduł.c
```

Analiza

Gdy moduł składa się z pojedynczego pliku o rozszerzeniu *.c*, istnieje duże prawdopodobieństwo, że moduł uda się zbudować i zainstalować za pomocą powyższego rozwiązania. Modułom składającym się z kilku plików źródłowych powinny towarzyszyć instrukcje instalacji.

Opcje `-cia` powodują kolejno kompilację, instalację, a następnie aktywację modułu. Pierwsza czynność jest oczywista, instalacja polega na umieszczeniu pliku *.so* w miejscu, w którym będzie go poszukiwał serwer Apache, a aktywacja polega na umieszczeniu odpowiedniego wpisu w pliku *httpd.conf*.

Zobacz również

- Strony podręcznika man dotyczące programu *apxs*, przeważnie *ServerRoot/man/man8/apxs.8*.

2.2. Instalacja modułu mod_dav w systemie uniksowym

Problem

Na serwerze WWW należy uruchomić usługę WebDAV. Dzięki usłudze WebDAV zdalni użytkownicy mogą dodawać, usuwać i uaktualniać pliki znajdujące się na serwerze w sposób niezawodny i bezpieczny, bez potrzeby korzystania z usługi FTP.

Rozwiązanie

Moduł *mod_dav* jest dołączony do wersji 2.0 serwera Apache, wystarczy go tylko uaktywnić za pomocą opcji kompilacji `--enable-dav`.

W przypadku serwera Apache 1.3 pakiet źródłowy modułu *mod_dav* należy pobrać ze strony *http://webdav.org/mod_dav/*, rozpakować, a następnie wydać polecenia:

```
% cd mod_dav-1.0.3-1.3.6
% ./configure --with-apxs=/usr/local/apache/bin/apxs
% make
# make install
```

Następnie należy zatrzymać i uruchomić serwer ponownie oraz zapoznać się z recepturą 6.18.

Analiza

Moduł *mod_dav* zachowuje się poprawnie i łatwo daje się zbudować i włączyć do działającego serwera. Aby sprawdzić, czy moduł został zainstalowany poprawnie, trzeba na potrzeby usługi WebDAV przeznaczyć na serwerze jakiś katalog i sprawdzić dostęp do niego za pomocą narzędzia wspomagającego WebDAV. Polecamy użycie do tego celu programu *cadaver* — korzystającego z wiersza poleceń narzędzia typu *open source*. (Adres strony, z której można pobrać program *cadaver*, znajduje się na końcu receptury).

Aby uaktywnić na serwerze WWW usługę WebDAV, należy do pliku *httpd.conf* dodać co najmniej dwie dyrektywy. Pierwsza wskazuje położenie bazy blokad plików, wykorzystywanej przez *mod_dav* po to, by operacje wykonywane przez WebDAV wzajemnie ze sobą nie kolidowały. Baza musi znajdować się katalogu, do którego serwer ma prawo zapisu. Na przykład:

```
# cd /usr/local/apache
# mkdir var
# chgrp nobody var
# chmod g+w var
```

Następnie w pliku *httpd.conf*, poza wszystkimi kontenerami, należy umieścić wiersze:

```
<IfModule mod_dav.c>
    DAVLockDB /usr/local/apache/var/DAVlock
</IfModule>
```

Baza DAVLockDB *nie może* znajdować się w systemie plików typu NFS, gdyż NFS nie obsługuje wymaganego przez moduł *mod_dav* sposobu blokowania plików. Umieszczenie bazy blokad plików w systemie plików typu NFS może doprowadzić do nieprzewidywalnych skutków.

Następnie należy utworzyć tymczasowy katalog służący do przetestowania działania usługi WebDAV:

```
# cd /usr/local/apache
# mkdir htdocs/dav-test
# chgrp nobody htdocs/dav-test
# chmod g+w htdocs/dav-test
```

Teraz do pliku *httpd.conf* należy dodać sekcję przeznaczającą utworzony katalog na potrzeby usługi WebDAV:

```
<Directory "/usr/local/apache/htdocs/dav-test">
    DAV On
</Directory>
```

Następnie należy zatrzymać i ponownie uruchomić serwer, który po uruchomieniu powinien rozpocząć obsługę operacji WebDAV kierowanych pod lokalny identyfikator URI */dav-test*. Aby przetestować działanie WebDAV za pomocą programu *cadaver*, należy wydać przedstawione polecenia, w wyniku czego powinno się otrzymać wyniki podobne do poniższych:

```
% cd /tmp
% echo "Zwykły tekst" > dav-test.txt
% cadaver
dav:!> open http://localhost/dav-test
Looking up hostname... Connecting to server... connected.
dav:/dav-test/> put dav-test.txt
Uploading dav-test.txt to '/dav-test/dav-test.txt': (reconnecting...done)
Progress: [= == == == == == == == == == == == == ==>] 100.0% of 11 bytes succeeded.
dav:/dav-test/> propset dav-test.txt MyProp 1023
Setting property on 'dav-test.txt': (reconnecting...done) succeeded.
dav:/dav-test/> propget dav-test.txt MyProp
Fetching properties for 'dav-test.txt':
Value of MyProp is: 1023
dav:/dav-test/> propdel dav-test.txt MyProp
Deleting property on 'dav-test.txt': succeeded.
dav:/dav-test/> close
Connection to 'localhost' closed.
dav:!> exit
% rm dav-test.txt
```

W tym przypadku właściwości (ang. *properties*) są atrybutami zasobów WebDAV. Niektórymi z nich (na przykład rozmiarem zasobu) zarządza system, a inne mogą być dowolnie dodawane, zmieniane i usuwane przez użytkownika.

Po sprawdzeniu poprawności działania modułu *mod_dav* katalog *htdocs/dav-test* należy usunąć, podobnie jak związaną z nim sekcję <Directory> pliku *httpd.conf*, a następnie należy zapoznać się z recepturą 6.18.

Zobacz również

- Receptura 6.18.
- *http://webdav.org/mod_dav/*.
- *http://webdav.org/cadaver/*.

2.3. Instalacja modułu mod_dav w systemie Windows

Problem

Na serwerze WWW Apache 1.3 działającym w systemie Windows należy za pomocą modułu *mod_dav* uruchomić usługę WebDAV.

Rozwiązanie

Moduł *mod_dav* w wersji 2.0 serwera Apache znajduje się standardowo, nie ma więc w takim przypadku konieczności pobierania go i instalowania.

Ze strony *http://webdav.org/mod_dav/win32/* należy pobrać i rozpakować pakiet modułu *mod_dav* przeznaczony dla systemu Windows. Następnie należy sprawdzić, czy w katalogu `ServerRoot` (w katalogu głównym serwera Apache) znajdują się pliki *xmlparse.dll* oraz *xmltok.dll*. Jeżeli ich tam nie ma, należy odnaleźć je w innych katalogach serwera Apache, a następnie skopiować je do katalogu `ServerRoot`. Do pracy moduł *mod_dav* potrzebuje pakietu *Expat*, który jest dołączany do serwera Apache począwszy od wersji 1.3.9. Poszukiwane pliki pochodzą z pakietu *Expat*.

Plik DLL modułu *mod_dav* należy przekopiować do katalogu, w którym znajdują się moduły serwera Apache:

```
C:\>cd mod_dav-1.0.3-dev
C:\mod_dav-1.0.3-dev>copy mod_dav.dll C:\Apache\modules
C:\mod_dav-1.0.3-dev>cd \Apache
```

W pliku *httpd.conf* należy umieścić następujący wiersz:

```
LoadModule dav_module modules/mod_dav.dll
```

Jeżeli plik *httpd.conf* zawiera dyrektywę `ClearModuleList` i dodaje wszystkie moduły, należy również dodać wiersz `AddModule`. Można też wiersz `LoadModule` dotyczący modułu *mod_dav* umieścić po dyrektywie `ClearModuleList`.

Analiza

Moduł *mod_dav* zachowuje się poprawnie i łatwo daje się zbudować i włączyć do działającego serwera. Aby sprawdzić, czy moduł został zainstalowany poprawnie, trzeba na potrzeby usługi WebDAV przeznaczyć na serwerze jakiś katalog i sprawdzić dostęp do niego za pomocą narzędzia wspomagającego WebDAV lub otworzyć go programem *Eksplorator Windows* (począwszy od systemu Windows 2000). Można też uzyskać do niego dostęp z innego komputera za pomocą programu *cadaver* lub innego narzędzia wspomagającego WebDAV.

Aby uaktywnić na serwerze WWW usługę WebDAV, należy do pliku *ServerRoot/conf/httpd.conf* dodać co najmniej dwie dyrektywy. Pierwsza wskazuje położenie bazy blokad plików wykorzystywanej przez *mod_dav* po to, by operacje wykonywane przez WebDAV wzajemnie nie kolidowały ze sobą. Baza musi znajdować się w katalogu, do którego serwer ma prawo do zapisu. Na przykład:

```
C:\Apache-1.3>mkdir var
```

Żeby uruchomić usługę WebDAV, do pliku *httpd.conf* należy dodać następujące wiersze:

```
<IfModule mod_dav.c>
    DAVLockDB "C:/Apache-1.3/var/dav-lock"
</IfModule>
```

Żeby przetestować pracę modułu *mod_dav*, należy utworzyć tymczasowy katalog:

```
C:\Apache-1.3>mkdir htdocs\dav-test
```

Następnie, po to, by usługa WebDAV rozpoczęła udostępnianie katalogu testowego, należy zmienić zawartość kontenera `<IfModule>`:

```
<IfModule mod_dav.c>
    DAVLockDB "C:/Apache-1.3/var/dav-lock"
    <Directory "C:/Apache-1.3/htdocs/dav-test">
        DAV On
    </Directory>
</IfModule>
```

Teraz należy zatrzymać serwer i uruchomić go ponownie, a następnie spróbować otworzyć katalog */dav-test* za pomocą klienta WebDAV. W przypadku zastosowania jako klienta WebDAV działającego w innym komputerze programu *cadaver*, przykład jego użycia można znaleźć w recepturze 2.2. Poniżej przedstawiono sposób testowania pracy modułu *mod_dav* za pomocą programu *Eksplorator Windows*.

Testowania pracy modułu mod_dav za pomocą programu Eksplorator Windows

Po przeznaczeniu katalogu *htdocs\dav-test* na potrzeby usługi WebDAV i ponownym uruchomieniu serwera WWW Apache należy uruchomić program *Eksplorator Windows*. Aby uzyskać dostęp do katalogu usługi WebDAV, należy wykonać niżej opisane czynności. Można to zrobić na komputerze, na którym uruchomiono usługę WebDAV lub na innym komputerze z systemem Windows, który ma dostęp do tego komputera.

1. W programie *Eksplorator Windows* należy kliknąć pozycję *Moje miejsca sieciowe*[1].
2. Następnie w prawym oknie programu *Eksplorator Windows* należy kliknąć dwukrotnie ikonę *Dodaj miejsce sieciowe*.
3. Jako lokalizację nowego miejsca sieciowego należy wpisać:

   ```
   http://127.0.0.1/dav-test/
   ```

   Gdy czynności te wykonywane są w innym komputerze niż ten, w którym uruchomiono usługę WebDAV, zamiast adresu `127.0.0.1` należy wpisać nazwę serwera, w którym uruchomiono moduł *mod_dav*.
4. Po naciśnięciu przycisku *Dalej* należy nadać utworzonemu miejscu sieciowemu nową nazwę lub pozostać przy proponowanej.
5. Program *Eksplorator Windows* powinien teraz otworzyć okno o nazwie zdefiniowanej w poprzednim kroku. Zawartość okna powinna być pusta, gdyż otwarty został pusty katalog.
6. W głównym oknie programu *Eksplorator Windows* należy przejść do dowolnego katalogu zawierającego pliki.

[1] Poniższa procedura dotyczy systemów Windows 2000 lub Windows Me. W systemie Windows XP należy kliknąć dwukrotnie znajdującą się na pulpicie ikonę Moje miejsca sieciowe, a następnie ze znajdującego się po lewej stronie panela Zadania sieciowe wybrać Dodaj miejsce sieciowe — *przyp. tłum.*

7. Przytrzymując wciśnięty przycisk *Ctrl*, należy przeciągnąć do okna otwartego w punkcie 5. dowolny plik lub pliki.
8. System Windows powinien na krótko pokazać okno informujące o postępie procesu kopiowania, po czym kopiowany plik powinien znaleźć się w oknie docelowym.

Gratulacje! Plik został skopiowany do serwera za pomocą usługi WebDAV.

Po zakończeniu testowania należy usunąć katalog *htdocs\dav-test*, a także usunąć z pliku konfiguracyjnego wiersz `<Directory "C:/Apache-1.3/htdocs/dav-test">`. W przeciwnym razie swoje pliki będzie mógł umieszczać tam każdy.

Zobacz również

- Receptura 6.18.
- *http://webdav.org/mod_dav/*.

2.4. Instalacja modułu mod_perl w systemie uniksowym

Problem

Aby skrypty Perla były wykonywane szybciej i były lepiej obsługiwane przez serwer WWW, należy zainstalować moduł *mod_perl*.

Rozwiązanie

W przypadku serwera Apache 1.3 ze strony *http://perl.apache.org/* należy pobrać pakiet źródłowy modułu *mod_perl* 1.0, rozpakować go, a następnie wydać następujące polecenia:

```
% perl Makefile.PL \
>     USE_APXS=1 \
>     WITH_APXS=/usr/local/apache/bin/apxs \
>     EVERYTHING=1 \
>     PERL_USELARGEFILES=0
% make
% make install
```

Następnie należy uruchomić serwer ponownie.

Dla serwera Apache 2.0 i wersji późniejszych proces przebiega analogicznie. Należy pobrać i rozpakować pakiet źródłowy *mod_perl* 2.0, a następnie wykonać następujące polecenie:

```
% perl Makefile.PL MP_APXS=/usr/local/apache2/bin/apxs
```

Analiza

Moduł *mod_perl* jest modułem dość złożonym. Na serwerze Apache można zainstalować go na kilka sposobów. W recepturze przedstawiono sposób najszybszy i najprostszy. Jeżeli z jakichś powodów sposób ten jest nieodpowiedni, należy zapoznać się z różnymi plikami *README.**

znajdującymi się w katalogu pakietu. Ponieważ podstawowym językiem modułu jest Perl, a nie C, instrukcja instalacji różni się zdecydowanie od stosowanych w przypadku większości innych modułów.

Gdy po instalacji modułu serwer uruchomi się poprawnie, moduł *mod_perl* rozpoczyna działanie i jest skonfigurowany. Działanie modułu można przetestować, dokonując zmian w pliku *httpd.conf*, dodając kilka skryptów i obserwując, czy serwer obsługuje je poprawnie. Oto przykład testowania działania modułu *mod_perl*.

1. Na potrzeby skryptów modułu *mod_perl* należy utworzyć nowy folder:

```
# cd ServerRoot
# mkdir lib lib/perl lib/perl/Apache
```

2. W katalogu *conf/* serwera należy utworzyć plik *startup.pl* zawierający instrukcje uruchomieniowe dla modułu *mod_perl*:

```
#!/usr/bin/perl
BEGIN {
    use Apache ( );
    use lib Apache->server_root_relative('lib/perl');
}
use Apache::Registry ( );
use Apache::Constants ( );
use CGI qw(-compile :all);
use CGI::Carp ( );
1;
```

3. Następnie należy utworzyć wykorzystywany w teście plik *lib/perl/Apache/HelloWorld.pm*:

```
package Apache::HelloWorld;
use strict;
use Apache::Constants qw(:common);
sub handler {
    my $r = shift;
    $r->content_type('text/plain; charset=ISO-8859-2');
    $r->send_http_header;
    $r->print("Witaj świecie! Pozdrowienia od modułu mod_perl.\n");
    return OK;
}
1;
```

4. Teraz trzeba otworzyć do edycji plik konfiguracyjny serwera i umieścić w nim dyrektywy umożliwiające modułowi *mod_perl* odnalezienie potrzebnych mu składników oraz informujące go, kiedy ma uruchomić skrypt testowy. Do pliku *httpd.conf* należy dodać następujące wiersze:

```
<IfModule mod_perl.c>
    PerlRequire conf/startup.pl
    <Location /mod_perl/howdy>
        SetHandler perl-script
        PerlHandler Apache::HelloWorld
    </Location>
</IfModule>
```

5. Teraz należy uruchomić serwer ponownie, a następnie uruchomić skrypt, wpisując: *http://localhost/mod_perl/howdy*.

Jeżeli konfiguracja jest poprawna, pojawi się strona zawierające zdanie: „Witaj świecie! Pozdrowienia od modułu mod_perl.”.

Zobacz również

- *http://perl.apache.org/*.
- Książka *Writing Apache Modules with Perl and C*, autorzy Doug MacEachern oraz Lincoln Stein, wydawnictwo O'Reilly.
- Książka *mod_perl Developer's Cookbook*, autorzy Geoffrey Young, Paul Lindner oraz Randy Kobes wydawnictwo Sams[2].

2.5. Instalacja modułu mod_php w systemie uniksowym

Problem

Do działającego serwera WWW Apache należy dodać moduł obsługi skryptów *mod_php*.

Rozwiązanie

Pakiet źródłowy modułu *mod_php* należy pobrać ze strony *http://php.net/*, rozpakować go, a następnie wydać następujące polecenia:

```
% cd php-5.2.3
% ./configure \
>     --with-apxs=/usr/local/apache/bin/apxs
% make
# make install
```

Następnie należy uruchomić serwer ponownie.

Analiza

Aby przekonać się, że instalacja zakończyła się powodzeniem, w katalogu `DocumentRoot` serwera WWW należy utworzyć składający się z jednego wiersza plik *info.php*:

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

Do pliku konfiguracyjnego *httpd.conf* należy dodać poniższe wiersze:

```
<IfModule mod_php4.c>
    AddHandler application/x-httpd-php .php
</IfModule>
```

Po ponownym uruchomieniu serwera WWW należy za pomocą przeglądarki spróbować otworzyć dokument *info.php*. W wyniku tego powinien pojawić się szczegółowy opis aktywnych opcji PHP. Jeżeli opis rzeczywiście się pojawi, będzie to wskazywać, że instalacja zakończyła się pomyślnie i plik *info.php* można usunąć.

Podczas instalacji PHP można zastosować wiele opcji i rozszerzeń, w tej recepturze omówiono tylko najprostszy sposób instalacji modułu.

[2] Polskie wydanie: mod_perl. Podręcznik programisty, Helion 2003 — *przyp. red.*

Zobacz również

- Receptura 8.16.
- Receptura 8.17.
- *http://php.net/*.

2.6. Instalacja modułu mod_php w systemie Windows

Problem

Do działającego w systemie Windows serwera WWW Apache należy dodać moduł obsługi skryptów *mod_php*.

Rozwiązanie

W tej recepturze zamiast podawania szczegółowych poleceń czynności, które należy wykonać, przedstawione zostaną w sposób opisowy.

1. Ze strony *http://php.net/* należy pobrać plik binarny *.zip* (a nie plik *.exe*) PHP dla systemu Windows zawierającego rozszerzenia API.
2. Plik *.zip* należy rozpakować do katalogu, w którym zawartość pliku może pozostać na zawsze (na przykład do katalogu *C:\PHP4*). W przypadku użycia programu *WinZip* należy zaznaczyć pole wyboru *Use folder names*, aby utworzona została taka struktura plików jak umieszczona w pliku *.zip*.
3. Do katalogu *\modules* znajdującego się w katalogu `ServerRoot` serwera Apache należy przekopiować plik *PHP4\SAPI\php4apache.dll*.
4. W oknie wiersza polecenia należy przejść do katalogu *PHP4*, do którego został rozpakowany plik *.zip*, a następnie wydać polecenia:

   ```
   ...\PHP4>copy php.ini-dist %SYSTEMROOT%\php.ini
   ...\PHP4>copy php4ts.dll %SYSTEMROOT%
   ```

   (W przypadku systemów Windows 95 oraz Windows 98 zamiast `%SYSTEMROOT%` należy użyć `%WINDOWS%`).
5. Następnie należy otworzyć do edycji plik *%SYSTEMROOT%\php.ini*, odnaleźć w nim wiersz zaczynający się od `extension_dir` i zmienić jego wartość tak, by wskazywała na katalog *PHP4\extensions*. Gdy, na przykład, plik *.zip* został rozpakowany do katalogu *C:\PHP4*, wiersz ten powinien wyglądać następująco:

   ```
   extension_dir = C:\PHP4\extensions
   ```
6. Następnie trzeba otworzyć do edycji plik *conf\httpd.conf* znajdujący się w katalogu `ServerRoot` i w pobliżu innych wierszy `LoadModule` należy dodać wiersz:

   ```
   LoadModule php4_module modules/php4apache.dll
   ```

   W pobliżu należy umieścić również wiersze dotyczące plików *.php*:

   ```
   <IfModule mod_php4.c>
       AddType application/x-httpd-php .php
   </IfModule>
   ```

7. Na koniec należy zrestartować serwer Apache, w wyniku czego moduł PHP powinien się uaktywnić.

Analiza

Aby zainstalować moduł PHP w systemie Windows, należy wykonać wiele drobiazgowych czynności. Żeby przekonać się, że proces instalacji zakończył się powodzeniem, w katalogu `DocumentRoot` serwera należy utworzyć składający się z jednego wiersza plik *info.php*:

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

Po ponownym uruchomieniu się serwera WWW należy za pomocą przeglądarki spróbować otworzyć dokument *info.php*. W wyniku tego powinien pojawić się szczegółowy opis aktywnych opcji PHP.

W czasie instalacji PHP można zastosować wiele opcji i rozszerzeń, w tej recepturze omówiono tylko najprostszy sposób instalacji modułu. Więcej szczegółów na ten temat można znaleźć w pliku *install.txt* znajdującym się w katalogu *PHP4* oraz w dokumentacji znajdującej się na stronach WWW.

Zobacz również

- *http://php.net/*.

2.7. Instalacja modułu mod_ssl

Problem

Do działającego serwera WWW Apache należy dodać obsługę SSL, instalując w nim *mod_ssl* — moduł bezpiecznego HTTP.

Rozwiązanie

Windows
: Sposób instalacji protokołu SSL w systemie Windows jest tematem receptury 7.2. Mówiąc w największym skrócie, najlepiej jest pobrać XAMPP z witryny *ApacheFriends.org*, chyba że posiada się odpowiednie doświadczenie w budowaniu kodu źródłowego w systemie Microsoft Windows.

Apache 2.0
: Moduł *mod_ssl* został włączony do wersji 2.0 serwera Apache, ale w przypadku budowy serwera z kodu źródłowego, moduł nie jest automatycznie kompilowany ani instalowany. Aby w takim przypadku zainstalować moduł *mod_ssl*, należy w poleceniu `./configure` zastosować opcję `--enable-ssl` oraz uaktywnić moduł dyrektywami `LoadModule` oraz `AddModule`.

Apache 1.3
: Aby zainstalować moduł *mod_ssl* w systemie uniksowym, ze strony *http://www.modssl. org/* należy pobrać i rozpakować archiwum tar modułu, a następnie wydać polecenia:

```
% cd mod_ssl-2.8.14-1.3.27[3]
% ./configure \
>    --with-apache=../apache_1.3.27 \
>    --with-ssl=SYSTEM \
>    --prefix=/usr/local/apache
% cd ../apache_1.3.27
% make
% make certificate
```

Analiza

Żeby moduł *mod_ssl* mógł działać, kod źródłowy serwera Apache musi zostać zmodyfikowany. Dzięki temu można zainstalować tylko taką wersję pakietu *mod_ssl*, która odpowiada wersji posiadanej dystrybucji serwera Apache. Jeżeli instalacja serwera Apache nie zawiera plików źródłowych (co ma miejsce na przykład w przypadku instalacji serwera z pakietu RPM czy z innej tego typu dystrybucji) — instalacja modułu *mod_ssl* nie powiedzie się.

Poza kodami źródłowymi serwera Apache do zainstalowania modułu *mod_ssl* potrzebne są jeszcze Perl oraz biblioteki OpenSSL. Ich położenie określa się za pomocą opcji `--with-ssl`. Jeżeli biblioteki znajdują się w katalogu utworzonym przez ich dostawcę, słowo kluczowe `SYSTEM` poinformuje, że należy ich szukać właśnie tam, dzięki czemu nie trzeba będzie ich szukać samemu.

W przeciwieństwie do większości innych modułów serwera Apache, aby zainstalować moduł *mod_ssl*, należy uruchomić skrypt *./configure* znajdujący się w katalogu modułu *mod_ssl*, a nie w katalogu plików źródłowych serwera Apache. Skrypt modułu wprowadza odpowiednie zmiany do skryptu serwera, a następnie uruchamia go.

W tej recepturze omówiono jedynie podstawowy sposób instalacji modułu *mod_ssl*. W czasie konfiguracji modułu *mod_ssl* można zlecić modułowi wykorzystanie wielu dodatkowych składników oraz funkcji. Więcej informacji na ten temat można znaleźć w plikach *README* oraz *INSTALL* znajdujących w katalogu plików źródłowych modułu *mod_ssl* lub na stronie WWW *http://www.modssl.org/*.

Zobacz również

- Receptura 7.3.
- *http://www.modssl.org/*.

2.8. Wyszukiwanie modułów na witrynie modules.apache.org

Problem

Trzeba znaleźć moduły serwera Apache udostępniające określone funkcje lub noszące określone nazwy. Wiadomo jednocześnie, że istnieje rejestr modułów serwera Apache.

[3] Nazwa katalogu zależna jest od instalowanej wersji mod_ssl. — *przyp. red.*

Rozwiązanie

Na witrynie *http://modules.apache.org* należy podać słowa kluczowe odnoszące się do poszukiwanych funkcji lub fragment nazwy modułu i przeprowadzić wyszukiwanie.

Analiza

Rejestr modułów serwera Apache to nieoficjalna witryna internetowa, na której autorzy modułów z własnej inicjatywy mogą zarejestrować efekty swoich prac, aby udostępnić je innym użytkownikom.

Wspomniana witryna w żadnym wypadku nie zawiera wszystkich modułów serwera Apache; wiele z nich jest dostępnych na witrynie SourceForge albo na domowych witrynach ich autorów. Jeżeli poszukiwany moduł nie zostanie znaleziony na *http://modules.apache.org*, można spróbować go poszukać na witrynie SourceForge (pod adresem *http://sourceforge.net*), FreshMeat (*http://freshmeat.net*) albo po prostu przeszukać internet przy użyciu Google lub innej wyszukiwarki internetowej.

Zobacz również

- *http://sourceforge.net*.
- *http://freshmeat.net*.

2.9. Instalacja modułu mod_security

Problem

Trzeba zainstalować moduł *mod_security*, aby skorzystać z udostępnianych przez niego prostych, a jednocześnie rozbudowanych mechanizmów filtrujących.

Rozwiązanie

Należy wykonać następujące czynności:

1. Pobrać moduł *mod_security* oraz główne reguły modułu z witryny *http://modules.apache.org*. Aby znaleźć moduł do pobrania na podanej stronie WWW, należy skorzystać z pozycji menu *Browse* lub *Search*.

Po pobraniu modułu warto sprawdzić jego sygnaturę PGP, aby upewnić się, że plik nie uległ zmianie. Więcej informacji na ten temat znajduje się na witrynie internetowej modułu *mod_security*.

2. Rozpakować zestaw (bez reguł) do katalogu roboczego:

```
% cd /usr/local/build
% tar xzf /usr/local/kits/modsecurity-apache_2.1.1
```

3. Przejść do rozpakowanego katalogu i zbudować pakiet, wykorzystując dostarczony wraz z nim skrypt *Makefile*. W poleceniu make trzeba wskazać odpowiednią wartość *ServerRoot*:

```
% cd /usr/local/build/modsecurity-apache_2.1.1/apache2
% make top_dir=/usr/local/apache2
# make top_dir=/usr/local/apache2 install
```

W odróżnieniu od innych modułów dostawców zewnętrznych *mod_security* trzeba budować przy użyciu jego własnych mechanizmów, a nie przez zwykłe wywołanie narzędzia *apxs* serwera Apache.

4. Rozpakować główne reguły do podkatalogu katalogu wskazanego jako *ServerRoot*:

```
# cd /usr/local/apache2/conf
# mkdir mod_security
# cd mod_security
# tar xzf /tmp/modsecurity-core-rules_2.1-1.4.tar.gz
```

5. W odpowiednim miejscu w pliku *httpd.conf* dodać następujące wiersze:

```
LoadModule security_module modules/mod_security2.so
Include conf/mod_security/*.conf
```

6. Zrestartować serwer.

Analiza

Skrypt *Makefile* dołączony do pakietu *mod_security* zbuduje moduł i umieści go w odpowiedniej lokalizacji, lecz ponowne włączenie serwera jest już powinnością samego użytkownika. Najnowsze wersje pakietu zawierają zestaw głównych reguł obsługujących przypadki takie jak spamowanie blogów czy najczęściej spotykane ataki. Reguły są również dostępne w oddzielnym archiwum tar, które może być uaktualniane niezależnie od reguł dołączanych do samego modułu.

Aktualna wersja modułu *mod_security* obsługuje wyłącznie serwer Apache w wersji 2. Dostępna jest również starsza wersja obsługująca wersję 1.3 serwera, lecz jest mało prawdopodobne, by była utrzymywana przez dłuższy czas.

Zobacz również

- Witryna internetowa modułu *mod_security* pod adresem *http://modsecurity.org*.

2.10. Dlaczego ten moduł nie działa?

Problem

Pomimo próby zainstalowania modułu pochodzącego od dostawcy zewnętrznego serwer WWW Apache nie rozpoznaje go.

Rozwiązanie

Należy sprawdzić kod źródłowy modułu, jego dokumentację albo zwrócić się bezpośrednio do autora modułu, aby ustalić wersję serwera Apache obsługiwaną przez moduł.

Analiza

W miarę wprowadzania do serwera Apache kolejnych znaczących zmian może się zdarzyć, że zmiana w API serwera doprowadzi do powstania niezgodności jego modułów. Wprawdzie twórcy rozwijający serwer dążą do tego, by podobne niezgodności zdarzały się jak najrzadziej, lecz czasami jest to po prostu nieuniknione.

Aby zapobiec ładowaniu niezgodnego modułu i uniknąć w ten sposób załamania serwera WWW, zarówno w module, jak i w samym serwerze umieszczono wbudowany, „magiczny" numer zapisywany w momencie ich budowania, wskazujący wersję API. Gdy serwer spróbuje załadować moduł DSO, najpierw porówna numer wersji w module z własnym numerem wersji zapisanym na serwerze. Jeżeli okaże się, że numery wersji są niezgodne, serwer nie załaduje modułu.

Zespół rozwijający serwer dąży do utrzymania zgodności między numerami wersji głównych, lecz nie między różnymi numerami wersji głównych. Inaczej mówiąc, moduł zbudowany dla serwera Apache 1.3 powinien działać z praktycznie wszystkimi wersjami 1.3 serwera zbudowanymi po dacie budowy modułu, natomiast na pewno nie będzie obsługiwać serwera w wersji 2.0. I odwrotnie: moduł dla wersji 2.0 w żadnym wypadku nie będzie współpracował z serwerem w wersji 1.3.

Zobacz również

- Rejestr modułów serwera Apache pod adresem *http://modules.apache.org*.