helion.pl
KSIĘGARNIA INTERNETOWA · WYDAWNICTWO

IDŹ DO
PRZYKŁADOWY ROZDZIAŁ
SPIS TREŚCI

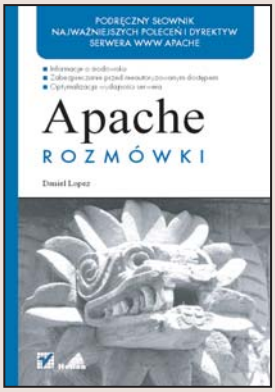
KATALOG KSIĄŻEK
KATALOG ONLINE
ZAMÓW DRUKOWANY KATALOG

TWÓJ KOSZYK
DODAJ DO KOSZYKA

CENNIK I INFORMACJE
ZAMÓW INFORMACJE O NOWOŚCIACH
ZAMÓW CENNIK

CZYTELNIA
FRAGMENTY KSIĄŻEK ONLINE

Apache. Rozmówki



Autorzy: Daniel Lopez, Jesus Blanco
Tłumaczenie: Grzegorz Werner
ISBN: 83-246-0669-6
Tytuł oryginału: Apache Phrasebook
Format: B6, stron: 272

Apache to jeden z najpopularniejszych serwerów WWW. Dostępny nieodpłatnie na licencji open source jest „motorem napędowym" ponad 70% witryn WWW w internecie. W miarę rozwoju serwera Apache jego możliwości i złożoność wzrosły do tego stopnia, że początkujący użytkownicy mogą czuć się przytłoczeni samą koniecznością jego skonfigurowania. Właściwe dostrojenie działania serwera wymaga modyfikowania plików konfiguracyjnych i odpowiedniego dobrania parametrów dyrektyw w nich zawartych. Na szczęście większość typowych funkcji realizuje się w podobny sposób na wszystkich platformach systemowych.

Książka „Apache. Rozmówki" to zestawienie najistotniejszych wiadomości dotyczących konfigurowania serwera Apache i administrowania nim. Można tam znaleźć także wiele przykładów i konkretnych wskazówek. Przeczytaj, a dowiesz się, jak zainstalować serwer Apache w różnych systemach operacyjnych oraz jak uruchomić go i zatrzymać. Nauczysz się diagnozować przyczyny niewłaściwego działania serwera i korzystać z dzienników. Poznasz sposoby łączenia katalogów zawierających pliki stron WWW z odpowiednimi adresami URL, zabezpieczania serwera i zestawiania połączeń szyfrowanych SSL/TLS. Proste stanie się dla Ciebie także maksymalizowanie wydajności serwera.

Książka zawiera omówienie takich oto zagadnień:

- instalowanie Apache'a w systemach Windows i Linux
- włączanie i wyłączanie serwera oraz jego pojedynczych modułów
- struktura plików konfiguracyjnych
- analiza dzienników serwera
- odwzorowywanie adresów URL
- metody uwierzytelniania użytkowników
- stosowanie certyfikatów SSL
- korzystanie z protokołu WebDAV
- poprawa wydajności Apache'a

Dzięki niej szybko skonfigurujesz serwer Apache, a kilkusetstronicowe instrukcje obsługi przestaną Ci być potrzebne.

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

Spis treści

O autorach **13**

Wstęp **15**

1 Apache — podstawy **17**

Podstawowe informacje o serwerze Apache 17
Sprawdzanie, czy Apache jest już zainstalowany 19
Instalowanie serwera Apache 1.3 w Linuksie i Uniksie 21
Instalowanie serwera Apache 2.0 w Linuksie i Uniksie 22
Instalowanie serwera Apache w Windows 23
Podstawowe informacje o pliku konfiguracyjnym 24
Używanie wielu plików konfiguracyjnych 26
Uruchamianie, zatrzymywanie i restartowanie serwera Apache ... 27
Zmiana adresu i portu używanego przez serwer Apache 29
Zmiana użytkownika, z którego przywilejami działa serwer Apache 30
Określanie nazwy serwera 31
Określanie ikony strony WWW 32
Ustalanie dostępnych modułów serwera 32
Włączanie i wyłączanie pojedynczych modułów 33
Dodawanie modułów bez rekompilowania serwera Apache ... 34
Publikowanie treści 36

2 Rozwiązywanie problemów 41
Pomocy! Mój serwer nie działa! 41
Dziennik błędów 42
Rejestrowanie zdarzeń przez syslog 43
Kontrolowanie ilości rejestrowanych informacji 43
Testowanie konfiguracji serwera Apache 45
Testowanie serwera Apache z wiersza poleceń 45
Sprawdzanie, czy serwer Apache działa 47
Inne sposoby zatrzymywania serwera Apache 48
Diagnozowanie serwera Apache za pomocą modułów 50
Błędy podczas uruchamiania serwera 51
Błędy odmowy dostępu 54
Wewnętrzne błędy serwera 55
Dodatkowe pliki dziennika błędów 57
Przekierowania nie działają 58
Lista diagnostyczna 58

3 Dzienniki i monitorowanie 65
Rejestrowanie żądań w serwerze Apache 65
Domyślne pliki dziennika serwera Apache 66
Tworzenie formatów dziennika 66
Tworzenie własnego pliku dziennika 68
Przekierowywanie dzienników do zewnętrznego programu 69
Warunkowe rejestrowanie żądań 70
Monitorowanie stron wskazujących witrynę 71
Monitorowanie serwera Apache za pomocą modułu mod_status ..71
Monitorowanie serwera Apache za pomocą SNMP 72
Analizowanie dzienników za pomocą narzędzi open source ... 74
Monitorowanie dzienników w czasie rzeczywistym 74
Rejestrowanie żądań w bazie danych 75
Rotacja i archiwizowanie dzienników 76

Odwzorowywanie adresów IP 77
Przetwarzanie zarejestrowanych adresów IP 78
Automatyczne restartowanie serwera Apache w razie awarii 79
Scalanie i dzielenie plików dziennika 80
Prowadzenie oddzielnego dziennika dla każdego hosta wirtualnego 81
Często spotykane wpisy dziennika 82

4 Odwzorowywanie adresów URL i treść dynamiczna 85
Odwzorowywanie adresów URL 85
Odwzorowywanie adresów URL na pliki za pomocą dyrektywy Alias 86
Odwzorowywanie wzorców URL na pliki za pomocą dyrektywy AliasMatch 87
Przekierowywanie żądań do innej lokacji 87
Przekierowywanie żądań do najnowszej wersji pliku 88
Przekierowywanie nieudanych lub nieautoryzowanych żądań ... 89
Definiowanie procedur obsługi treści 90
Typy MIME 91
Konfigurowanie typów MIME 92
Zasady wykonywania skryptów CGI 92
Oznaczanie zasobów jako wykonywalnych skryptów CGI 93
Wiązanie skryptów z metodami HTTP i typami MIME 94
Rozwiązywanie problemów z wykonywaniem skryptów CGI .. 95
Zwiększanie wydajności skryptów CGI 96
Server Side Includes 97
Konfigurowanie SSI 97
Ustawianie zmiennych środowiskowych 98
Dynamiczne ustawianie zmiennych środowiskowych 99
Specjalne zmienne środowiskowe 101
Negocjacja treści 101

Konfigurowanie negocjacji treści 103
Określanie domyślnych zestawów znaków i priorytetu języków . 104
Zaawansowane odwzorowywanie adresów URL za pomocą modułu mod_rewrite 105
Problem „końcowego ukośnika" 106
Poprawianie pomyłek w pisowni 107
Rozwiązywanie problemów z wielkością liter 108
Weryfikowanie stron za pomocą programu Tidy 109

5 Hosting wirtualny 111
Co to jest hosting wirtualny? 111
Co to jest hosting wirtualny oparty na adresach IP? 112
Konfigurowanie hostingu wirtualnego opartego na adresach IP ..113
Co to jest hosting wirtualny oparty na nazwach? 114
Konfigurowanie hostingu wirtualnego opartego na nazwach 115
Co się dzieje, kiedy żądanie nie pasuje do żadnego hosta wirtualnego? 116
Konfigurowanie domyślnego hosta wirtualnego opartego na nazwie 117
Konfigurowanie domyślnego hosta wirtualnego opartego na adresie IP 118
Łączenie hostów wirtualnych opartych na nazwach i na adresach IP 119
Diagnozowanie konfiguracji hostów wirtualnych 120
Używanie SSL w połączeniu z hostami wirtualnymi opartymi na nazwach 121
Inna metoda hostingu wirtualnego 121
Inne moduły do hostingu wirtualnego 123
Pliki konfigurujące poszczególne katalogi 124
Określanie zasięgu plików konfigurujących poszczególne katalogi 125
Wyłączanie plików konfigurujących poszczególne katalogi ... 126

6 Bezpieczeństwo i kontrola dostępu 127
Stosowanie kontroli dostępu 127
Różnice między wersjami serwera Apache 128
Uwierzytelnianie podstawowe i ze skrótem komunikatu 129
Kontrola dostępu do serwera Apache 131
Konfiguracja autoryzacji i uwierzytelniania w serwerze Apache 132
Tworzenie bazy użytkowników 133
Autoryzowanie użytkowników i grup za pomocą dyrektywy Require 134
Obsługa wielu kont użytkowników 135
Zezwalanie na dostęp tylko spod określonych adresów IP 136
Blokowanie dostępu spod określonych adresów IP 137
Łączenie metod kontroli dostępu 138
Dostosowywanie strony odmowy dostępu 138
Oddawanie kontroli w ręce użytkowników 140
Blokowanie dostępu do plików systemowych i poufnych 141
Ograniczanie wykonywania programów 142
Zapobieganie nadużyciom 143
Wyłączanie listingów katalogów 144
Zmiana nagłówka Server: 145
Zapobieganie tworzeniu zewnętrznych łączy do obrazów 145
Ograniczanie wybranych metod HTTP 146
Ograniczanie dostępu na podstawie typu przeglądarki 148
Korzystanie z sekcji Location i Directory 149
Dodatkowe moduły uwierzytelniające 149
Apache 2.2 151
Aktualizowanie zabezpieczeń serwera Apache 152
Procedura zabezpieczania serwera 153

7 SSL/TLS 159
Co to jest SSL? 159
Jak działa SSL? 160
Kompilowanie pakietu OpenSSL 162
Klucze szyfrowania 163
Tworzenie pary kluczy 163
Tworzenie pary kluczy chronionych hasłem 164
Usuwanie hasła z klucza 165
Certyfikaty 165
Tworzenie wniosku o podpisanie certyfikatu 166
Wyświetlanie zawartości wniosku o podpisanie certyfikatu .. 168
Tworzenie samodzielnie podpisanego certyfikatu 168
Kompilowanie serwera Apache 1.3 z obsługą SSL 169
Kompilowanie serwera Apache 2.x z obsługą SSL 171
Minimalna konfiguracja serwera Apache 171
Uruchamianie serwera Apache z obsługą SSL 172
SSLPassPhraseDialog 173
Zwiększanie wydajności SSL 174
Realizacja wszystkich żądań z wykorzystaniem SSL 175
SSL i hosty wirtualne oparte na nazwach 176
Używanie modułów uwierzytelniających w połączeniu z SSL 177
Komunikaty ostrzegawcze wyświetlane podczas dostępu do witryny obsługującej SSL 177
Tworzenie certyfikatów klienta 178
Uwierzytelnianie z wykorzystaniem certyfikatów klienta 179
Rozwiązania alternatywne 180
Testowanie witryn SSL z wiersza poleceń 181
Rozwiązywanie problemów z nieprawidłowymi implementacjami SSL 181

Złożona kontrola dostępu
z wykorzystaniem modułu mod_ssl 182
Powiązane rozdziały 183

8 Publikowanie treści za pomocą DAV 185
Publikowanie treści w serwerze Apache 185
Wprowadzenie do WebDAV 186
Korzyści związane z użyciem modułu mod_dav 187
WebDAV i protokół HTTP 188
Instalowanie modułu mod_dav w serwerze Apache 2.0 189
Instalowanie modułu mod_dav w serwerze Apache 1.3 190
Podstawowa konfiguracja WebDAV 191
Zabezpieczanie konfiguracji WebDAV 191
Dostęp do zasobów DAV z programów Microsoft Office 193
Dostęp do zasobów DAV z systemu Microsoft Windows 194
Dostęp do zasobów DAV z przeglądarki Firefox 196
Dostęp do zasobów DAV z wiersza poleceń 197
Obsługa błędnie działających klientów 198
Moduł mod_spelling i DAV 199
Treść dynamiczna i DAV 199
Włączanie stron użytkowników 200
Inny sposób obsługi stron użytkowników 202
Rozwiązywanie problemów z plikiem DAVLockDB 202

9 Wydajność i skalowalność 205
Dostrajanie serwera Apache 205
Wydajność i skalowalność 206
Dostrajanie sprzętu 206
Zwiększanie limitów systemu operacyjnego 207
Zwiększanie liczby procesów 208
Zwiększanie liczby deskryptorów plików 209

Kontrolowanie zewnętrznych procesów 210
Zwiększanie wydajności systemu plików 211
Dostrajanie ustawień sieci i śledzenia stanu serwera 214
Zapobieganie nadużyciom 217
Ograniczanie pasma i liczby połączeń 218
Blokowanie robotów 220
Pośredniki odwrotne i urządzenia do równoważenia obciążenia 221
Buforowanie i kompresja 222
Optymalizacje specyficznych modułów 223
Alternatywne serwery 223

10 Obsługa pośredniczenia i buforowania w serwerze Apache 225

Zastosowanie buforowania i pośredniczenia 225
Pośredniki zwykłe i odwrotne 226
Różnice między wersjami 1.3, 2.0 i 2.2 serwera Apache 226
Włączanie obsługi modułu mod_proxy 227
Włączanie obsługi pośredniczenia zwykłego 228
Unifikowanie przestrzeni URL za pomocą pośrednika odwrotnego 229
Ukrywanie serwerów zaplecza 230
Blokowanie pośredniczenia odwrotnego w przypadku wybranych adresów URL 231
Zwiększanie wydajności 232
Wspomaganie przetwarzania SSL 233
Przekazywanie informacji o pośredniczeniu w nagłówkach 234
Manipulowanie nagłówkami 235
Konfigurowanie pośrednika buforującego 236
Buforowanie w serwerze Apache 2 237
Równoważenie obciążenia 238

Łączenie się z serwerem Tomcat 239
Inne pośredniki 240
Przezroczyste pośredniki HTTP 241

11 Moduły wieloprocesowe i protokołowe 243
Ewolucja architektury serwera Apache 243
Wybór modułu MPM 244
Procesowe moduły MPM 245
Konfigurowanie modułu MPM prefork 246
Wątkowe i hybrydowe moduły MPM 247
Konfigurowanie modułu MPM worker 248
Inne moduły MPM 249
Filtry w serwerze Apache 2 250
Apache jako serwer FTP 251
Apache jako serwer POP3 252
Kompresowanie treści „w locie" 253

Skorowidz 255

2

Rozwiązywanie problemów

W tym rozdziale zostaną szczegółowo opisane najczęściej spotykane problemy — na przykład błędne ustawienia praw dostępu do plików i brak możliwości powiązania serwera z określonym portem — oraz ich rozwiązania. Omówionych zostanie też kilka narzędzi i zasobów pozwalających zidentyfikować przyczyny problemów.

Pomocy! Mój serwer nie działa!

Wiadomo, że nie ma nic bardziej frustrującego, niż przerwa w lekturze technicznej książki wymuszona zmaganiami z błędnie działającym oprogramowaniem. Nie chcemy, aby była to jedna z takich książek! Właśnie dlatego poruszamy te kwestie na samym początku. Z tej przyczyny niniejszy rozdział opisuje zarówno zagadnienia podstawowe, jak i zaawansowane; początkujący użytkownicy Apache'a mogą opuścić te, które ich nie dotyczą.

Dziennik błędów

```
ErrorLog logs/error_log
```

Plik dziennika błędów przechowuje informacje o ważnych zdarzeniach, w tym o uruchomieniach, restartach, błędach w działaniu serwera oraz zabronionych albo nieprawidłowych żądaniach. Jest to pierwsze miejsce, w które należy zajrzeć podczas rozwiązywania problemów z serwerem.

W systemach uniksowych plik `error_log` jest umieszczany domyślnie w podkatalogu `logs/` katalogu instalacyjnego Apache'a. Jeśli instalacja Apache'a stanowi część dystrybucji systemu, plik może znajdować się w innym miejscu, najczęściej `var/log/httpd`.

W Windows plik nosi nazwę `error.log` i również znajduje się w podkatalogu `logs`.

Za pomocą dyrektywy `ErrorLog` można określić ścieżkę do pliku dziennika. Można również podać ścieżkę do programu i poprzedzić ją symbolem potoku (|), aby błędy były wysyłane do innego programu. Tę często używaną technikę opisano w rozdziale 3.

Zwróćmy uwagę, że plik dziennika błędów zostanie utworzony dopiero po pierwszym uruchomieniu Apache'a!

Rejestrowanie zdarzeń przez syslog

```
ErrorLog syslog
ErrorLog syslog:local7
```

W systemach uniksowych można określić `syslog` jako argument dyrektywy `ErrorLog`, aby Apache rejestrował błędy za pomocą demona `syslog`. Pokazano to w powyższym przykładzie. Opcjonalnie można dołączyć kanał (domyślnie `local7`). Kanał `syslog` to pole informacyjne związane z komunikatem `syslog`, które wskazuje źródło komunikatu. Kanały od `local0` do `local10` są zarezerwowane na użytek administratora i aplikacji takich jak Apache.

Kontrolowanie ilości rejestrowanych informacji

```
LogLevel notice
```

Informacje o zdarzeniach generowane przez Apache'a mają różne stopnie ważności. Dyrektywa `LogLevel` w połączeniu z argumentami wymienionymi w tabeli 2.1 pozwala wybrać rodzaj rejestrowanych komunikatów. Rejestrowane będą tylko błędy o określonym i wyższym poziomie ważności.

Tabela 2.1. Opcje dyrektywy LogLevel według dokumentacji serwera Apache

Ustawienie	Opis	Przykład
`emerg`	Poważne błędy — system nie nadaje się do użytku	`Child cannot open lock file. Exiting.`
`alert`	Konieczna jest natychmiastowa reakcja	`getpwuid: couldn't determine user name from uid`
`crit`	Zdarzenia krytyczne	`socket: Failed to get socket, exiting child.`
`error`	Błędy	`Premature end of script headers.`
`warn`	Ostrzeżenia	`Child process 1234 did not exit, sending another SIGHUP.`
`notice`	Zwykłe, ale istotne informacje	`httpd: caught SIGBUS, attempting to dump core in...`
`info`	Informacje	`Server seems busy. (You may need to increase StartServers, or Min/MaxSpareServers) ...`
`debug`	Komunikaty diagnostyczne	`Opening config file...`

Domyślny poziom błędów `warn` jest odpowiedni w większości instalacji Apache'a. Jeśli jednak konieczne jest zdiagnozowanie specyficznej konfiguracji, można obniżyć poziom błędów aż do `debug`, aby uzyskać znacznie bardziej szczegółowe informacje.

Testowanie konfiguracji serwera Apache

```
# apachectl configtest
```

Polecenie to pozwala zbadać plik konfiguracyjny Apache'a pod kątem problemów przed zastosowaniem go w aktywnym serwerze. Apache używa tej samej procedury do przetestowania konfiguracji za każdym razem, kiedy użytkownik wydaje mu polecenie restartu za pośrednictwem skryptu `apachectl`. Gwarantuje to, że działający serwer uruchomi się prawidłowo z nowym plikiem konfiguracyjnym.

Testowanie serwera Apache z wiersza poleceń

```
$ telnet www.apache.org 80
Trying 192.87.106.226...
Connected to ajax-1.apache.org (192.87.106.226).
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Sun, 04 Sep 2005 20:42:02 GMT
Server: Apache/2.0.54 (Unix) mod_ssl/2.0.54
     OpenSSL/0.9.7a DAV/2 SVN/1.2.0-dev
Last-Modified: Sat, 03 Sep 2005 11:35:42 GMT
ETag: "203a8-2de2-3ffdc7a6d3f80"
Accept-Ranges: bytes
Content-Length: 11746
Cache-Control: max-age=86400
```

```
Expires: Mon, 05 Sep 2005 20:42:02 GMT
Connection: close
Content-Type: text/html; charset=ISO-8859-1
Connection closed by foreign host.
```

HTTP jest prostym protokołem tekstowym, więc można użyć klienta Telnetu — programu, który pozwala połączyć się bezpośrednio ze wskazanym komputerem i portem — do sprawdzenia obecności serwera w zdalnym hoście. Jeśli serwer nie odpowie, a wiadomo, że sieć jest prawidłowo skonfigurowana, oznacza to, że Apache nie nasłuchuje pod określonym adresem i w określonym porcie. Technika ta bywa przydatna w środowiskach, w których niedostępna jest przeglądarka WWW, na przykład podczas zdalnego dostępu do serwera przez SSH. Jeśli na przykład można uzyskać dostęp do Apache'a w zdalnym komputerze przez adres `localhost`, ale nie za pomocą zdalnej przeglądarki, może to oznaczać problemy z zaporą sieciową albo nieprawidłowe ustawienie dyrektywy `Listen`.

Należy połączyć się przez Telnet z adresem *www.apache.org* (albo inną witryną WWW) w porcie 80. i wpisać:

```
HEAD / HTTP/1.0
```

albo:

```
GET / HTTP/1.0
```

a następnie dwukrotnie nacisnąć klawisz *Enter*. Pojawi się odpowiedź podobna do pokazanej w powyższym przykładzie.

Jeśli w systemie zainstalowana jest tekstowa przeglądarka WWW `lynx`, można uzyskać podobny wynik za pomocą poniższego polecenia:

```
lynx -head -dump http://www.apache.org
```

W rozdziale 7. zostanie opisany moduł `mod_ssl`, a Czytelnicy poznają sposób łączenia się z serwerem obsługującym SSL za pomocą narzędzia `openssl`.

Sprawdzanie, czy serwer Apache działa

```
ps -aux | grep httpd
25297 ?        S      0:00 /usr/local/www/bin/httpd
-k start
15974 ?        S      0:06 /usr/local/www/bin/httpd
-k start
14441 ?        S      0:02 /usr/local/www/bin/httpd
-k start
...

/usr/sbin/lsof | grep httpd |grep IPv
httpd     14441  nobody    3u  IPv4     136524
                  TCP www.example.com:http (LISTEN)
httpd     25297    root    3u  IPv4     136524
                  TCP www.example.com:http (LISTEN)
httpd     30277  nobody    3u  IPv4     136524
                  TCP www.example.com:http (LISTEN)
...
netstat -ltnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address          Foreign
     Address         State       PID/Program name
tcp        0      0 192.168.1.151:80       0.0.0.0:
     *               LISTEN      25297/httpd
tcp        0      0 0.0.0.0:22             0.0.0.0:
     *               LISTEN      1038/sshd
```

Czasem nie można połączyć się z serwerem, ale nie wiadomo, czy to serwer nie działa, czy może wystąpił jakiś problem z siecią. W systemach uniksowych dostępnych jest kilka narzędzi, które pozwalają ustalić przyczynę problemu. Niektóre z nich pokazano w powyższym przykładzie.

Narzędzie `ps` pokazuje, czy w systemie działa proces `httpd`.

Narzędzia `netstat` oraz `lsof` pokazują adresy i porty, w których nasłuchuje serwer Apache.

W systemie Windows można użyć Menedżera zadań Windows (uruchamianego przez jednoczesne naciśnięcie klawiszy *Alt*+*Ctrl*+*Delete*), aby sprawdzić, czy działa proces `Apache.exe`. Stan serwera można również sprawdzić za pomocą programu monitorującego dołączanego do nowszych dystrybucji i działającego w zasobniku systemowym.

Inne sposoby zatrzymywania serwera Apache

```
# kill -HUP 25297
# kill -9 25297
```

Czasem wygodniej jest bezpośrednio wysłać sygnał do serwera za pomocą narzędzia `kill` zamiast używać skryptu `apachectl`. W tym celu należy najpierw ustalić identyfikator procesu działającego serwera z wykorzystaniem programów `ps` lub `lsof`. Następnie można zakończyć proces

za pomocą polecenia `kill`, które przyjmuje dwa argumenty: wysyłany sygnał oraz identyfikator procesu serwera Apache (w tym przykładzie 25297). Sygnał `KILL` (albo `TERM`) zatrzymuje serwer, a sygnał `HUP` uruchamia go ponownie. Można również zastąpić nazwę sygnału jej liczbowym odpowiednikiem, jak pokazano w powyższym przykładzie. Więcej informacji można znaleźć w podręczniku systemowym pod hasłem `kill`.

W Linuksie można również wysłać sygnał do wszystkich procesów o nazwie `httpd` za pomocą polecenia `killall`. Aby na przykład usunąć wszystkie procesy `httpd`, należy wydać polecenie:

```
# killall -KILL httpd
```

Należy jednak zachować ostrożność, ponieważ jeśli w systemie działa kilka egzemplarzy Apache'a, powyższe polecenie usunie je wszystkie!

Aby powyższe polecenia zadziałały, użytkownik musi mieć odpowiednie uprawnienia. W niemal wszystkich przypadkach trzeba być albo superużytkownikiem, albo właścicielem procesu Apache'a, żeby go zatrzymać lub uruchomić ponownie.

W systemach Windows można wymusić zamknięcie Apache'a za pomocą przycisku *Zakończ proces* w Menedżerze zadań Windows.

Diagnozowanie serwera Apache za pomocą modułów

Istnieje kilka modułów Apache'a, które mogą pomóc podczas rozwiązywania problemów z konfiguracją serwera albo aplikacją WWW.

Moduł `mod_loopback`, narzędzie do diagnozowania klientów WWW, po prostu odsyła do przeglądarki wszystkie odebrane informacje dotyczące żądania HTTP, w tym dane POST lub PUT. Moduł jest dostępny pod adresem:

http://www.snert.com/Software/mod_loopback/index.shtml

Moduły `mod_tee` oraz `mod_trace_output` zapisują treść wysyłaną przez serwer. Można je znaleźć pod następującymi adresami:

http://apache.webthing.com/mod_tee/
http://trace-output.sourceforge.net/

Moduł `mod_logio` wchodzący w skład Apache'a 2 zapisuje wszystkie dane odebrane lub zwrócone przez serwer w dzienniku błędów.

Wszystkie te moduły zmniejszają wydajność serwera, ale bywają bardzo przydatne, na przykład podczas analizowania problemów z nagłówkami albo plikami cookie.

Błędy podczas uruchamiania serwera

W tym podrozdziale zostaną opisane problemy, które mogą zapobiec uruchomieniu Apache'a, oraz związane z nimi komunikaty o błędach.

Błąd składni

```
Syntax error on line xxx of /etc/http/httpd.conf:
Invalid command 'PiidFile', perhaps misspelled or
defined by a module not included in the server
configuration
```

Błąd składni oznacza, że użytkownik pomylił się podczas wpisywania dyrektywy (w tym przykładzie `PidFile`) albo że zastosował dyrektywę obsługiwaną przez moduł, który nie został dodany do serwera. Należy sprawdzić składnię pliku konfiguracyjnego w miejscu wskazanym przez komunikat o błędzie. W rozdziale 1. wyjaśniono, jak użyć dyrektywy `<ifModule>` do warunkowego wyłączania dyrektyw, aby plik konfiguracyjny mógł zostać przetworzony nawet wtedy, gdy pewien moduł jest niedostępny.

Zajęty adres

```
Address already in use: make_sock: could not bind to
port
```

Powyższy błąd oznacza, że inny program używa już portu, z którym Apache próbuje się związać. Aby rozwiązać problem, należy zatrzymać ten program przed uruchomieniem Apache'a albo otworzyć plik konfiguracyjny `httpd.conf` i zmienić port, w którym Apache nasłuchuje przychodzących żądań (dyrektywy `Listen` i `Port`).

W większości przypadków błąd ten występuje dlatego, że w systemie działa już inny egzemplarz serwera Apache. W Windows zdarza się, że port Apache'a jest zajęty przez program Internet Information Server albo Microsoft Personal Web Server. Port 80. bywa również używany przez inne popularne programy, na przykład Skype.

Niewystarczające uprawnienia

```
[Mon Jan 9 20:09:50 2005] [crit] (13)Permission
denied: make_sock: could not bind to port 80
```

Powyższy komunikat wskazuje, że użytkownik nie ma wystarczających uprawnień, aby nakazać Apache'owi związanie się z portem określonym w pliku konfiguracyjnym. W Uniksie tylko użytkownicy uprzywilejowani mogą wiązać się z portami z zakresu 1 – 1024. Aby rozwiązać ten problem, należy zalogować się jako superużytkownik albo wydać polecenie `su` i ponownie spróbować uruchomić serwer. Ci, którzy nie mają dostępu do konta superużytkownika, mogą otworzyć plik `httpd.conf` i zmienić port używany przez Apache'a na wyższy niż 1024.

Niezgodny moduł

```
module xxx is not compatible with this version of
Apache
```

Powyższy błąd występuje wtedy, gdy Apache próbuje wczytać moduł skompilowany pod kątem nowszej (lub starszej) wersji serwera niż obecnie zainstalowana w systemie. Użytkownicy, którzy dysponują kodem źródłowym modułu, mogą zrekompilować go na użytek odpowiedniej wersji w sposób opisany w rozdziale 1. Ci, którzy nie mają kodu źródłowego albo nie mogą zrekompilować modułu, a potrzebują oferowanych przez niego funkcji, mogą zainstalować nowszą (albo starszą) wersję serwera zgodną z modułem.

Odwzorowywanie nazw

```
Cannot determine hostname
```

Kilka dyrektyw Apache'a, w tym `ServerName` i `Listen`, przyjmuje argumenty w postaci nazw hostów. Jeśli jednak podczas rozruchu Apache nie zdoła przetłumaczyć nazwy hosta na adres z wykorzystaniem *Domain Name System* (DNS) albo systemowej listy hostów, wystąpi powyższy błąd. Aby rozwiązać problem, należy zweryfikować ustawienia DNS i `/etc/hosts` oraz pisownię nazw hostów w pliku `httpd.conf`. Jeśli to możliwe, w dyrektywach takich jak `Listen` oraz `<VirtualHost>` należy używać adresów IP zamiast nazw hostów.

Błąd podczas otwierania dziennika albo pliku konfiguracyjnego

```
(13)Permission denied: httpd: could not open error
log file /usr/local/apache/logs/error_log.
```

Powyższy błąd wskazuje, że użytkownik nie ma wystarczających uprawnień, aby odczytać plik konfiguracyjny Apache'a albo zapisać dane w plikach dziennika.

Problem ten często pojawia się, kiedy serwer jest uruchamiany przez użytkownika innego niż ten, który go skompilował i zainstalował. Należy albo uruchomić Apache'a z konta superużytkownika, albo użyć polecenia `chmod`, aby zmienić właściciela pliku podanego w komunikacie o błędzie.

Błędy odmowy dostępu

```
Forbidden/You don't have permission to access /xxx
on this server
```

Jeśli przeglądarka zwraca błąd `403 Forbidden/Access Denied` podczas próby wczytania strony z serwera Apache, oznacza to, że dany adres URL podlega ograniczeniom dostępu, a żądanie nie spełnia określonych warunków. Aby rozwiązać problem, należy zmienić prawa dostępu do treści WWW albo plików i upewnić się, że właściciel procesu Apache'a ma uprawnienia do odczytu i wykonania we wszystkich katalogach prowadzących do dokumentu.

W systemach uniksowych uprawnienia te można ustawić za pomocą polecenia `chmod`.

Komunikat `Client denied by server configuration` w dzienniku błędów wskazuje, że dostępu odmówiono ze względu na dyrektywy kontrolne (takie jak `Allow` i `Deny`) w sekcjach `<Directory>` lub `<Location>` odnoszących się do danego adresu URL.

Komunikat `Directory index forbidden by rule` wskazuje, że użytkownik próbował uzyskać dostęp do katalogu, w którym nie ma pliku indeksu. Informacje o indeksowaniu katalogów i plikach indeksu można znaleźć w opisie opcji `Indexes` dyrektywy `Options` w rozdziale 6.

```
Options ExecCGI is off in this directory
```

Jeśli podczas próby wykonania skryptu CGI pojawi się komunikat `Options ExecCGI is off in this directory`, oznacza to, że skrypt nie został zaznaczony jako wykonywalny w pliku konfiguracyjnym Apache'a albo że w danym katalogu nie można wykonywać skryptów. Więcej informacji można znaleźć w opisie dyrektyw `ScriptAlias` lub `Options`.

Wewnętrzne błędy serwera

Wewnętrzne błędy serwera to błędy, które uniemożliwiają Apache'owi realizację żądania.

Błędy segmentacji

```
child pid exit signal Segmentation Fault (11)
```

Błąd segmentacji występuje, gdy Apache próbuje uzyskać dostęp do obszarów pamięci należących do innych procesów albo gdy w kodzie Apache'a pojawi się błędnie uformowana albo nielegalna instrukcja. Może to wynikać z usterek w źle napisanych albo eksperymentalnych bibliotekach lub modułach, a także z błędów sprzętowych, zwykle w systemowej pamięci, chipsecie, magistrali lub procesorze.

Przedwczesny koniec nagłówków skryptu

```
[error] [client 192.168.200.3] Premature end of
script headers:
/usr/local/aoache/cgi-bin/test-cgi
```

Błąd ten jest spowodowany niekompletnym wykonaniem skryptu CGI. Należy upewnić się, że skrypt ma ustawione prawo do wykonania i że jego pierwszy wiersz wskazuje prawidłowy program interpretera. Błąd ten może wystąpić na przykład wtedy, gdy skrypt zaczyna się od wiersza `#!/usr/local/bin/perl`, a w rzeczywistości interpreter Perla znajduje się w pliku `/usr/bin/perl`.

Błędy przedwczesnego końca nagłówków zazwyczaj pojawiają się wtedy, gdy działanie skryptu zostanie przerwane, zanim zwróci on jakiekolwiek dane. Może to wynikać z wielu przyczyn, na przykład z błędów w kodzie

albo braku bibliotek, z którymi skonsolidowany jest program. Jak wyjaśniono w rozdziale 9., w niektórych przypadkach proces może zostać przerwany przez system operacyjny albo przez Apache'a, jeśli zużycie zasobów (pamięci, czasu procesora) przekroczy pewien limit.

Źle uformowane nagłówki

```
[error] [client 192.168.200.3] malformed header from
script. Bad header=xxx:
/usr/local/apache/cgi-bin/example.cgi
```

Ten błąd występuje wtedy, gdy nagłówki generowane przez skrypt mają nieprawidłowy format (zwykle z powodu błędu programisty). Odpowiedź skryptu powinna zaczynać się od nagłówków (zera lub więcej), po których następuje pusty wiersz.

Dodatkowe pliki dziennika błędów

```
RewriteLog /usr/local/apache/logs/rewrite_log
RewriteLogLevel warn
SSLLog /usr/local/apache/logs/ssl_log
SSLLogLevel warn
ScriptLog logs/cgi_log
```

Kilka modułów — w tym moduł SSL Apache'a 1.3, `mod_rewrite` oraz `mod_cgi` — udostępnia własne dyrektywy do rejestrowania danych specyficznych dla modułu w oddzielnym pliku.

Przekierowania nie działają

```
UseCanonicalName off
```

Jeśli serwer Apache staje się niedostępny, kiedy przekierowuje żądanie do innej strony, może to oznaczać, że kanoniczna nazwa hosta jest nieprawidłowa albo niedostępna na zewnątrz sieci.

Jeśli na przykład dyrektywa `ServerName` jest ustawiona na `localhost`, 127.0.0.1 albo prywatny adres, serwer będzie niedostępny, gdy przekieruje użytkownika pod adres URL oparty na tych wartościach.

Żeby rozwiązać ten problem, należy podać prawidłową wartość `ServerName` albo ustawić dyrektywę `UseCanonicalName` na `off`, aby samoreferencyjne adresy URL były konstruowane na podstawie nazwy hosta podanej przez klienta. Problem ten występuje często w przypadku komputerów znajdujących się za odwrotnym pośrednikiem (co opisano w rozdziale 10.).

Lista diagnostyczna

W tym rozdziale podsumowano najczęściej występujące problemy z serwerem Apache.

Uruchamianie serwera

Jeśli serwer nie uruchamia się, należy sprawdzić w pliku dziennika, czym jest to spowodowane.

Jeśli pod określonym adresem działa inny serwer, należy wybrać inną kombinację adresu i portu dla nowego serwera.

Jeśli użytkownik nie ma wystarczających uprawnień, aby powiązać serwer z żądanym portem, powinien uruchomić Apache'a z konta superużytkownika (`root`), aby uzyskać dostęp do uprzywilejowanych portów.

Jeśli Apache nie może otworzyć pliku konfiguracyjnego albo dziennika, należy sprawdzić, czy pliki należą do użytkownika, który zainstalował Apache'a, i czy ma on prawo do zapisywania tych plików.

Łączenie się z serwerem

Jeśli próba dostępu do strony kończy się niepowodzeniem, najpierw trzeba ustalić, czy problem jest spowodowany przez serwer, sieć czy też przez przeglądarkę.

Najpierw należy sprawdzić, czy Apache działa. W tym celu należy wydać polecenie `ps` lub `netstat` albo skorzystać z Menedżera zadań (w Windows). Może się okazać, że serwer w ogóle nie został uruchomiony.

Następnie należy sprawdzić, czy można połączyć się z Apache'em z lokalnego komputera. W tym celu należy bezpośrednio połączyć się z serwerem za pomocą programu `telnet` i wpisać proste żądanie.

Lista diagnostyczna

Następnie należy sprawdzić, czy Apache używa prawidłowej kombinacji adresu i portu. Jeśli można uzyskać dostęp do serwera lokalnie, ale nie zdalnie, prawdopodobnie Apache nasłuchuje pod lokalnym adresem albo w porcie niedostępnym zdalnie. Za pomocą poleceń `netstat` lub `lsof` należy ustalić, pod jakimi adresami nasłuchuje Apache i czy są one poprawne.

Należy też sprawdzić, czy prawidłowo skonfigurowana jest zapora sieciowa lub router. Jeśli Apache nasłuchuje pod prawidłowym adresem, ale jest niedostępny na zewnątrz sieci, może to oznaczać, że ruch do serwera jest blokowany. Łączność między hostami można sprawdzić za pomocą polecenia `traceroute` (`tracert` w Windows). Wiele systemów operacyjnych domyślnie blokuje dostęp z zewnątrz z wyjątkiem kilku wybranych portów. Sposób rozwiązania tego problemu zależy od dystrybucji. Na przykład w systemach Fedora można użyć narzędzia `system-config-securitylevel`, a w Windows — programu Zapora systemu Windows znajdującego się w Panelu sterowania.

Wreszcie jeśli połączenie z serwerem jest nawiązywane za pośrednictwem *Secure Sockets Layer* (SSL — rozdział 7.), a użytkownik korzysta ze starszej przeglądarki albo nietypowej konfiguracji, należy poszukać w dzienniku błędów komunikatów świadczących o problemach z szyfrowaniem danych SSL.

Nie znaleziono dokumentu

Jeśli można uzyskać dostęp do serwera, ale pojawia się błąd `Document not found`, należy sprawdzić, czy dokument rzeczywiście istnieje w systemie plików.

Następnie należy sprawdzić, czy żądanie dotarło do serwera. W tym celu należy wyszukać w pliku `access_log` żądania pochodzące od określonego hosta. Jeśli w systemie jednocześnie działa kilka egzemplarzy Apache'a, klient mógł nawiązać połączenie z niewłaściwym serwerem.

Potem należy sprawdzić, czy dyrektywy `Alias` wskazują właściwą lokację — to znaczy katalog, w którym przechowywany jest docelowy dokument. Należy upewnić się, czy nazwa katalogu została wpisana prawidłowo i czy nie został on przypadkowo usunięty.

Wreszcie należy zbadać nieprawidłowe przekierowania, w tym „końcowe ukośniki" oraz problemy z dyrektywą `ServerName` opisane wcześniej w tym rozdziale.

Zabroniony dostęp

Jeśli dokument istnieje, ale dostęp do niego jest zabroniony, może to wynikać z kilku często popełnianych pomyłek.

Należy sprawdzić, czy Apache ma uprawnienia do odczytu pliku.

Należy sprawdzić, czy właściciel procesu Apache'a ma uprawnienia do odczytu i wyświetlania zawartości wszystkich katalogów na ścieżce wiodącej do pliku.

Lista diagnostyczna

Należy sprawdzić, czy użytkownik próbuje uzyskać dostęp do katalogu bez pliku indeksu, a tworzenie indeksów katalogów jest zabronione w pliku konfiguracyjnym Apache'a.

Należy upewnić się, że spełnione są wszystkie wymagania określone przez dyrektywy kontroli dostępu w pliku konfiguracyjnym Apache'a.

Jeśli użytkownik próbuje uzyskać dostęp do skryptu CGI, należy sprawdzić, czy skrypt ma ustawione prawa do odczytu i wykonania.

Wewnętrzne błędy serwera

Jeśli podczas próby wczytania strony w przeglądarce pojawia się komunikat `Internal server error`, należy poszukać przyczyny w pliku `error_log`.

Należy zadać sobie następujące pytania: Czy użytkownik próbuje uzyskać dostęp do skryptu CGI? Czy skrypt ma właściwe prawa do odczytu i wykonania? Czy ścieżka do interpretera w pierwszym wierszu skryptu jest prawidłowa? Czy jest on oznaczony jako skrypt CGI przez dyrektywę `ScriptAlias` lub podobną?

Jeśli wszystko inne zawiedzie...

W niniejszym rozdziale omówiono tylko najczęstsze problemy, z którymi mają do czynienia użytkownicy Apache'a. Jeśli ktoś napotka inny problem, pierwszym krokiem

powinno być poszukanie informacji w dziennikach błędów. Aby uzyskać więcej wskazówek, można zwiększyć poziom `LogLevel`. Następnie należy przeszukać dokumentację Apache'a, listy dyskusyjne oraz bazę usterek. Wreszcie można wysłać pytanie na listę dyskusyjną użytkowników Apache'a, przestrzegając dwóch reguł — najpierw należy samemu „odrobić zadanie domowe", a następnie podać wystarczająco dużo informacji, aby inni mogli pomóc. Na listę można zapisać się pod adresem:

http://httpd.apache.org/lists.html#http-users