

Das XAMPP-Handbuch

open source library

Open Source Software wird gegenüber kommerziellen Lösungen immer wichtiger. Addison-Wesley trägt dieser Entwicklung Rechnung mit den Büchern der **Open Source Library**. Administratoren, Entwickler und User erhalten hier professionelles Know-how, um freie Software effizient einzusetzen. Behandelt werden sowohl Themen wie Betriebssysteme, Netzwerke und Sicherheit als auch Programmierung.

Eine Auswahl aus unserem Programm:



phpMyAdmin ist das am weitesten verbreitete Tool zur Verwaltung von MySQL-Datenbanken. Dieses Buch gibt Ihnen einen profunden Einstieg in die effiziente Verwaltung von MySQL-Datenbanken mit phpMyAdmin (neue Version 2.6.0) bis hin zur professionellen Administration. Der Autor zeigt, wie Sie mit phpMyAdmin Datenbanken anlegen, editieren, abfragen und pflegen und wie Sie Benutzer anlegen und verwalten. Sie lernen, wie Sie Abfragen optimieren und automatisieren und was Sie beim Arbeiten mit verschiedenen Zeichensätzen und MIME-Typen beachten müssen u.v.a.m. Ein Troubleshooting-Kapitel hilft bei Problemen.

phpMyAdmin

Marc Delisle

288 S.

Euro 19,95 (D), 20,60 (A)

ISBN-13: 978-3-8273-2250-0

ISBN-10: 3-8273-2250-2



Ob Sie mit Joomla! Ihre kleine Webpräsenz attraktiver machen oder es als multifunktionales Framework für Ihre Teamarbeit oder die Firmenwebsite einsetzen wollen – dieses Buch führt Sie von der Installation bis zu Administration und Betrieb.

Diese deutlich erweiterte Neuauflage beschreibt neben den wichtigen Erweiterungen für Kalender, Shops, Newsletter und Foren noch umfassender, wie Sie bestehende Joomla!-Templates anpassen und eigene Templates schreiben. Neue Kapitel behandeln u.a. die Themen Barrierefreiheit und Workflow-Steuerung – beide Features sind mit Joomla! 1.5 standardmäßig verfügbar.

Joomla!

Hagen Graf

400 S., 1 CD

Euro 29,95 (D), 30,80 (A)

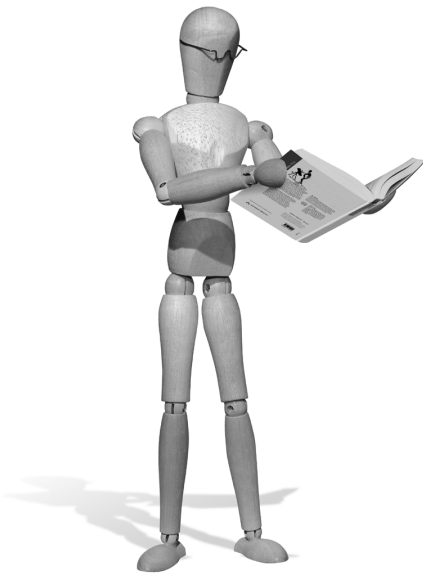
ISBN-13: 978-3-8273-2395-8

ISBN-10: 3-8273-2395-9

Kai Seidler, Kay Vogelgesang

Das **XAMPP-** Handbuch

Der offizielle Leitfaden zu
Einsatz und Programmierung



 **ADDISON-WESLEY**

An imprint of Pearson Education

München • Boston • San Francisco • Harlow, England
Don Mills, Ontario • Sydney • Mexico City
Madrid • Amsterdam

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

Die Informationen in diesem Produkt werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht.

Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt.

Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden.

Verlag, Herausgeber und Autoren können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Für Verbesserungsvorschläge und Hinweise auf Fehler sind Verlag und Herausgeber dankbar.

Alle Rechte vorbehalten, auch die der fotomechanischen Wiedergabe und der Speicherung in elektronischen Medien.

Die gewerbliche Nutzung der in diesem Produkt gezeigten Modelle und Arbeiten ist nicht zulässig.

Fast alle Hardware- und Softwarebezeichnungen und weitere Stichworte und sonstige Angaben, die in diesem Buch verwendet werden, sind als eingetragene Marken geschützt. Da es nicht möglich ist, in allen Fällen zeitnah zu ermitteln, ob ein Markenschutz besteht, wird das ® Symbol in diesem Buch nicht verwendet.

Umwelthinweis:

Dieses Produkt wurde auf chlorfrei gebleichtem Papier gedruckt.

10 9 8 7 6 5 4 3 2 1

07 06

ISBN-13: 978-3-8273-2281-4

ISBN-10: 3-8273-2281-2

© 2006 by Addison-Wesley Verlag,

ein Imprint der Pearson Education Deutschland GmbH

Martin-Kollar-Straße 10–12, D-81829 München/Germany

Alle Rechte vorbehalten

Einbandgestaltung: Marco Lindenbeck, webwo GmbH (mlindenbeck@webwo.de)

Lektorat: Boris Karnikowski, bkarnikowski@pearson.de

Korrektorat: Marita Böhm, München

Herstellung: Monika Weiher, mweiher@pearson.de

Satz: reemers publishing services gmbh, Krefeld (www.reemers.de)

Druck: Bercker Graph. Betrieb, Kevelaer

Printed in Germany

Inhaltsübersicht

Vorwort	17
Teil I XAMPP-Grundlagen	21
1 Was ist eigentlich XAMPP?	23
2 Installation	29
Teil II Fortgeschrittene Benutzung von XAMPP	49
3 Apache	51
4 PHP	73
5 ProFTPD	141
6 FileZilla Server	151
7 XAMPP und E-Mail	165
8 XAMPP und XAMPP im Duett	179
9 Performance Tuning	189
10 Sicherheit	217
11 Online mit DynDNS	249
12 XAMPP im Unterricht	261
Teil III Datenbanken	273
13 MySQL	275
14 SQLite	289
15 Microsoft SQL Server	299
16 Oracle	315
17 Firebird	333
18 PostgreSQL	347

Teil IV	XAMPP erweitern und ausbauen	361
19	Komponenten selbst kompilieren	363
20	PHP erweitern	387
21	Apache erweitern	397
22	Perl erweitern	403
23	XAMPP und Java	409
24	Applikationsbeispiele	437
25	Der BIND-Nameserver	483
26	Eigenes XAMPP für Windows	501
	Stichwortverzeichnis	517

Inhaltsverzeichnis

Vorwort	17
Teil I XAMPP-Grundlagen	21
1 Was ist eigentlich XAMPP?	23
1.1 Die klassische Client-Server-Architektur	23
1.2 Auf der Serverseite	24
1.3 Wer braucht XAMPP?	25
1.4 Was macht XAMPP besonders?	26
1.5 Systemvoraussetzungen	28
2 Installation	29
2.1 Windows	29
2.1.1 Eine kleine Geschichte der Windows-Version	30
2.1.2 Installation des XAMPP-Basispakets	32
2.1.3 Dynamische Installation auf Wechseldatenträgern	37
2.1.4 Installationen zusätzlicher XAMPP-Add-Ons	38
2.1.5 Die Installation von Upgrades	41
2.1.6 XAMPP Lite-Besonderheiten	41
2.2 Linux	42
2.2.1 Installation	42
2.2.2 Die Installation von Upgrades	46
2.3 Das Thema Sicherheit	47
Teil II Fortgeschrittene Benutzung von XAMPP	49
3 Apache	51
3.1 Virtual Hosts	52
3.1.1 Namensauflösung	52
3.1.2 httpd-vhosts.conf mit Virtual Hosts	53
3.2 Verzeichnisschutz mit .htaccess	56
3.3 Eine Frage der Sicherheit: SSL	58
3.3.1 Ein selbstzertifiziertes Serverzertifikat erstellen	58
3.3.2 Ein CA-zertifiziertes Serverzertifikat erstellen	61

3.4	WebDAV	64
3.4.1	Digest	64
3.4.2	Basic	64
3.4.3	XAMPP-Spezifika	65
3.4.4	Die Beispielkonfiguration im XAMPP für Linux	65
3.4.5	Die Beispielkonfiguration im XAMPP für Windows	66
3.4.6	AuthType Digest	68
3.4.7	Anwendungsmöglichkeiten mit WebDAV	69
4	PHP	73
4.1	Entwicklungsumgebungen	73
4.1.1	Zend Studio 5	73
4.1.2	Installation für XAMPP unter Windows adaptieren	75
4.1.3	Installation für XAMPP unter Linux adaptieren	81
4.1.4	Freie Alternativen	84
4.1.5	Weaverslave	85
4.2	PDF generieren	85
4.2.1	pdf class	86
4.2.2	FPDF	95
4.3	XML/RSS	101
4.3.1	DOM und SAX	101
4.3.2	SAX	102
4.3.3	DOM	107
4.3.4	XML und RSS	111
4.3.5	XML_sql2xml	115
4.3.6	SOAP	117
4.4	PHP und Flash	120
4.4.1	Flash Art	122
4.4.2	Balkendiagramm	129
4.5	Benutzerauthentifizierung	134
4.5.1	Benutzerdatenbank mit phpMyAdmin einrichten	134
4.5.2	Passwortabfrage programmieren	136
5	ProFTPD	141
5.1	XAMPP und ProFTPD	141
5.2	Virtual Hosts	143
5.3	Anonymous FTP	145
5.3.1	Anonymous FTP ohne Upload-Möglichkeit	146
5.3.2	Anonymous FTP mit Upload-Möglichkeit	147

6	FileZilla Server	151
6.1	FileZilla Server-Interface	153
6.2	Virtual Hosts	154
6.3	Anonymous FTP	156
6.4	FTP-Beispiele mit PHP & Perl	158
6.5	Voraussetzungen für den externen FTP-Zugriff	159
6.6	Secure-FTP mit FileZilla und SSL/TLS	161
7	XAMPP und E-Mail	165
7.1	Windows: Mercury Mail Transport System	165
7.1.1	Eine Frage der Domain	167
7.1.2	Postfächer und Aliase	169
7.1.3	SMTP: Beschränkungen oder Authentifizierung	171
7.1.4	Automatische Antworten	173
7.2	Fake sendmail	174
8	XAMPP und XAMPP im Duett	179
8.1	Das Front-/Backend-Szenario	179
8.2	PHP-Zugriff auf COM-Komponenten über XAMPP für Windows	180
8.2.1	MS ACCESS	181
8.2.2	MS EXCEL	183
8.3	Umleiten mit RewriteRule und ReverseProxy	186
9	Performance Tuning	189
9.1	Gefühlter oder realer Performance-Gewinn?	189
9.2	ApacheBench	189
9.3	Testlabor	192
9.4	Apache	194
9.4.1	CustomLog	194
9.4.2	HostnameLookups	195
9.4.3	EnableMMap und EnableSendfile	196
9.4.4	DirectoryIndex	197
9.4.5	ServerTokens	198
9.4.6	AllowOverride	199
9.4.7	LoadModule	200
9.4.8	Optionen FollowSymLinks und SymLinksIfOwnerMatch	202
9.4.9	Und sonst?	203

9.5	PHP	204
9.5.1	Komprimieren mit ob_gzhandler	204
9.5.2	eAccelerator	206
9.6	MySQL	207
9.6.1	Die richtige Konfigurationsdatei	207
9.6.2	table_cache	208
9.6.3	max_connections	210
9.6.4	table_locks_immediate vs. table_locks_waited	211
9.6.5	key_buffer_size	212
9.6.6	log_slow_queries	213
9.7	Perl	214
9.7.1	mod_perl	215
10	Sicherheit	217
10.1	Begrifflichkeiten: Server, Ports und Protokolle	217
10.2	Linux	219
10.2.1	Eine generelle Absicherung	219
10.2.2	Zugriffsrechte	223
10.2.3	Firewalls	226
10.2.4	Backups	229
10.3	Windows	231
10.3.1	Eine generelle Absicherung	231
10.3.2	Benutzerkonten für die XAMPP-Server	233
10.3.3	Firewalls	235
10.3.4	Backups	239
10.4	»Moderne« Gefahren	244
10.4.1	SQL-Injections	244
10.4.2	Cross-Site Scripting (XSS)	246
11	Online mit DynDNS	249
11.1	Überlegungen	250
11.1.1	Der Test der DynDNS-Fähigkeit in drei Schritten	251
11.1.2	Sicherheit	253
11.2	DynDNS einrichten	254
11.2.1	Ein DynDNS-Konto anlegen	254
11.2.2	DynDNS mit direkter PPP-Verbindung	255
11.2.3	DynDNS mit einem Hardware-Router als Gateway	257
11.3	Optimierungen	259
11.3.1	Forever online!	259

12	XAMPP im Unterricht	261
12.1	Zentrale Netzwerklösungen	261
12.1.1	Benutzerzugriff unter Windows	262
Teil III	Datenbanken	273
13	MySQL	275
13.1	Starten und Stoppen	275
13.1.1	Linux	275
13.1.2	Windows	276
13.2	Administrative Werkzeuge auf Kommandoebene	277
13.2.1	mysql	277
13.2.2	mysqladmin	278
13.2.3	mysqlbinlog	278
13.2.4	mysqlcheck	278
13.2.5	mysqldump	278
13.2.6	mysqlimport	278
13.2.7	MySqlManager	278
13.2.8	mysqlshow	279
13.2.9	mysqltest	279
13.3	MySQL GUIs	279
13.4	phpMyAdmin	280
13.5	Beispielbenutzer und -datenbank einrichten	281
13.6	Der Datenbankzugriff mit PHP	282
13.7	Der Datenbankzugriff mit Perl	285
14	SQLite	289
14.1	Installation	289
14.2	Beispieldatenbank einrichten	289
14.3	Datenbankzugriff mit PHP	292
14.4	Datenbankzugriff mit Perl	295
14.5	Tools	296
14.5.1	SQLiteCC	296
14.5.2	phpSQLiteAdmin	296
15	Microsoft SQL Server	299
15.1	Die Installation	299
15.1.1	MS SQL Server 2005 Express	300
15.1.2	MS SQL Server 2000 Desktop Engine	304

15.2	Eine Beispieldatenbank einrichten	305
15.3	Datenbankzugriff mit PHP unter Linux	307
15.4	Datenbankzugriff mit PHP unter Windows	309
15.4.1	Datenbankzugriff mit Perl unter Windows	310
16	Oracle	315
16.1	Installation	315
16.1.1	Windows	316
16.1.2	Linux	317
16.2	Beispielbenutzer und -datenbank einrichten	320
16.2.1	Beispielbenutzer aktivieren	321
16.2.2	Beispieltabelle einrichten	322
16.3	Aktivierung der PHP-Oracle-Unterstützung unter Linux	323
16.4	Aktivierung der PHP-Oracle-Unterstützung unter Windows	325
16.5	Datenbankzugriff mit PHP	325
16.6	Datenbankzugriff mit Perl	329
16.7	Typische Fehlermeldungen und ihre Ursachen	331
17	Firebird	333
17.1	Installation	333
17.1.1	Windows	333
17.1.2	Linux	334
17.2	Beispielbenutzer und -datenbank einrichten	335
17.3	Datenbankzugriff mit PHP	338
17.3.1	Interbase-Erweiterung aktivieren	338
17.3.2	Das Beispiel	338
17.4	Datenbankzugriff mit Perl	340
17.4.1	Das Beispiel unter Windows	340
17.4.2	Das Beispiel unter Linux	342
17.5	Tools	343
17.5.1	ibWebAdmin	344
18	PostgreSQL	347
18.1	Installation	348
18.1.1	Windows	348
18.1.2	Linux	350
18.2	Beispielbenutzer und -datenbank einrichten	351

18.3	Datenbankzugriff mit PHP	353
18.3.1	PostgreSQL-Erweiterung unter Linux aktivieren	353
18.3.2	PostgreSQL-Erweiterung unter Windows aktivieren	353
18.3.3	Das Beispiel	354
18.4	Datenbankzugriff mit Perl	356
18.4.1	Das Beispiel	356
18.5	Ein praktisches Werkzeug: phpPgAdmin	358
Teil IV	XAMPP erweitern und ausbauen	361
19	Komponenten selbst kompilieren	363
19.1	Windows	364
19.1.1	OpenSSL kompilieren	364
19.1.2	Zlib übersetzen	365
19.1.3	Apache-Quellen übersetzen	366
19.1.4	MySQL kompilieren	368
19.1.5	PHP übersetzen	370
19.2	Linux	378
19.2.1	Vorbereitungen	378
19.2.2	Generelle Vorgehensweise	379
19.2.3	Beispiel: Apache neu übersetzen	382
19.2.4	Besonderheiten bei PHP	384
19.2.5	Perl neu übersetzen	385
20	PHP erweitern	387
20.1	PEAR	388
20.1.1	list	388
20.1.2	list-all	388
20.1.3	list-upgrades	389
20.1.4	install	389
20.1.5	upgrade	390
20.1.6	upgrade-all	390
20.1.7	uninstall	391
20.2	PECL	391
20.2.1	Unter Linux	391
20.2.2	Unter Windows	393

21	Apache erweitern	397
21.1	Das Beispiel ModSecurity	397
21.1.1	Die Kompilierung unter Linux	397
21.1.2	Die Kompilierung unter Windows	399
21.1.3	Das Konfigurationsbeispiel	400
22	Perl erweitern	403
22.1	Windows	403
22.1.1	CPAN	403
22.1.2	PPM	405
22.2	Linux	406
23	XAMPP und Java	409
23.1	Apache Tomcat	409
23.1.1	Installation unter Linux	410
23.1.2	Installation unter Windows	415
23.1.3	Tomcat und Connector individualisieren	419
23.1.4	Die Administrationsbereiche des Tomcat sichern	421
23.2	XML	423
23.2.1	XML Parser	423
23.2.2	Cocoon	424
23.3	Java Server Pages und Datenbanken	425
23.3.1	MySQL	425
23.3.2	PostgreSQL	429
23.3.3	Oracle	430
23.3.4	Microsoft SQL Server	433
24	Applikationsbeispiele	437
24.1	Joomla!	437
24.1.1	Installation	437
24.1.2	Ein allererster Schritt	441
24.2	TYPO3	443
24.2.1	Installation ImageMagick	443
24.2.2	Installation TYPO3	444
24.2.3	Ein allererster Schritt	452
24.3	Gallery	457
24.3.1	Voraussetzungen	457
24.3.2	Vorbereitungen	458
24.3.3	Setup	459
24.3.4	Konfiguration	462

24.4	SquirrelMail	464
24.4.1	Entpacken unter Linux	465
24.4.2	Entpacken unter Windows	466
24.4.3	Setup	466
24.4.4	Verwendung mit einem externen IMAP-Server	469
24.5	phpBB	470
24.5.1	Installation	471
24.5.2	Konfiguration	474
25	Der BIND-Nameserver	483
25.1	Warum ein eigener Nameserver?	483
25.2	DNS für Sparsame: die Hosts-Datei	484
25.3	Eine kleine Einführung in das DNS	485
25.3.1	Die Resource Records	486
25.3.2	Ein komplettes Zone File	490
25.4	Installation	490
25.4.1	Linux	491
25.4.2	Windows	492
25.5	Konfigurationsbeispiele	494
25.5.1	Caching Name Server	494
25.5.2	Caching und Primary Name Server	495
25.5.3	Primary Name Server	496
25.5.4	Zone File	496
25.5.5	BIND starten	497
25.5.6	Eigenen Nameserver verwenden	498
25.5.7	Testen	499
26	Eigenes XAMPP für Windows	501
26.1	Der XAMPP NSIS Installer mit eigenem Programm	501
26.1.1	Vorbereitungen	502
26.1.2	Der Nullsoft Installer	505
26.1.3	Die Installer-CD-ROM	510
26.2	Das eigene XAMPP-Add-On	511
26.2.1	DokuWiki anpassen	511
26.2.2	Das Add-On als ZIP und 7-zip-exe	512
26.2.3	Das Add-On als Installer	513
	Stichwortverzeichnis	517



Vorwort

Als wir im Frühjahr 2002 die erste XAMPP-Version veröffentlichten, hätten wir es nie für möglich gehalten, dass das Feedback so positiv sein würde und XAMPP eine so große Verbreitung finden könnte. Das Ganze begann, nachdem wir unser Buch *Apache für Dummies* beendet hatten. Wir überlegten uns weitere Möglichkeiten, wie man den Einstieg in die Apache-Welt erleichtern könnte. Dabei kam uns recht schnell die Idee zu XAMPP. Zunächst gründeten wir *Apache Friends*, um dem Ganzen einen Rahmen zu geben. Ein paar Monate später erblickte dann die erste XAMPP-Version das Licht der Öffentlichkeit. Damals zwar noch unter den Namen LAMPP und WAMPP und im blauen Look-and-Feel, aber abgesehen von den Namen und der Oberfläche war es unser erstes XAMPP.

Mit XAMPP haben Sie die Möglichkeit, auf ganz einfache Weise Techniken wie Apache, MySQL, PHP und Perl auf Ihren Rechner nach Hause oder ins Büro zu holen, und das ohne ein spezielles Vorwissen und ohne großen Aufwand. Darin besteht der ganze Sinn und Zweck von XAMPP.

An wen richtet sich dieses Buch?

Mit diesem Buch können wir endlich noch einen Schritt weitergehen. Wir stellen Ihnen das Werkzeug XAMPP in all seinen Facetten vor und zeigen Ihnen, was Sie alles mit XAMPP machen und wie Sie es effektiv einsetzen können.

Kurz und knapp: Dieses Buch richtet sich an alle, die XAMPP professionell nutzen wollen. Darüber hinaus spricht es auch die Anwender an, die XAMPP erweitern und an ihre eigenen Anforderungen anpassen möchten.

Inhaltlich haben wir dieses Buch an den Fragen und Problemen orientiert, die in den letzten Jahren immer wieder bei der Benutzung von XAMPP aufgetreten sind. Zum einen handelt es sich dabei um grundlegende Fragen, wie zum Beispiel die Verwendung von Virtual Hosts, zum anderen um spezielle Fragen, wie zum Beispiel die Anbindung von XAMPP an eine Oracle-Datenbank. Alle diese oft sehr unterschiedlichen Probleme und Fragen greifen wir in diesem Buch auf.

Dieses Buch erhebt keinen Anspruch auf eine umfassende Referenz für alle hier behandelten Themen. Vielmehr ist es ein praxisorientierter Leitfaden zu XAMPP und keine Apache-Bibel, keine MySQL-Bibel, keine PHP-Bibel, keine Oracle-Bibel, keine

Was-weiß-ich-Bibel. Dazu gehört auch, dass wir bestimmte Themen nicht vertiefen können, in diesem Fall müssen wir stoppen und frei nach Dr. McCoy sagen: »Dies ist ein XAMPP-Buch und kein ...«

Worum geht es in diesem Buch?

Dieses Buch ist in vier Teile gegliedert. Und mit jedem Teil werden höhere Anforderungen an den Leser gestellt. So setzt der erste Teil kaum Vorkenntnisse voraus. Im dritten Teil jedoch, in dem es um Datenbanken geht, gehen wir beispielsweise davon aus, dass SQL nicht mehr erklärt werden muss. Im vierten Teil, der sich unter anderem mit dem Selbstkompilieren vom Apache beschäftigt, sollte der Leser zum Beispiel wissen, was ein C-Compiler ist.

Aber eins nach dem anderen:

Teil I

Im ersten und zugleich kleinsten Teil geht es um die Basics. Das erste Kapitel beschäftigt sich mit den Grundlagen von XAMPP: Was ist XAMPP eigentlich und was ist es nicht? Im zweiten Kapitel wird die Installation von XAMPP unter Linux und Windows gezeigt.

Teil II

Der zweite Teil konzentriert sich auf die einzelnen Komponenten von XAMPP und deren fortgeschrittene Nutzung. Das dritte Kapitel behandelt grundlegende Fragen wie etwa die Einrichtung von SSL und die Benutzung von Virtual Hosts im Apache. Im vierten Kapitel werden verschiedene Aspekte von PHP beleuchtet, unter anderem das dynamische Generieren von PDFs und die Schnittstellen, die PHP bietet, um mit XML-Daten zu arbeiten.

In Kapitel 5 und Kapitel 6 werden die FTP-Server von XAMPP vorgestellt. Unter Windows ist es der FileZilla Server und bei Linux der ProFTPD. Sie erfahren, wie man auf beiden Systemen Virtual Hosts und Anonymous FTP einrichtet.

Im siebten Kapitel geht es rund um das Thema E-Mail. Der Schwerpunkt liegt allerdings auf der Windows-Version von XAMPP, denn nur sie bringt einen eigenen Mailserver mit. Unter Linux setzen wir ein bereits funktionierendes Mailsystem voraus.

Das achte Kapitel widmet sich dem Zusammenspiel der Linux- und der Windows-Version von XAMPP. Es gibt Dinge, die nur unter Windows möglich sind – wie zum Beispiel der Zugriff auf MS Access oder Excel –, zugleich möchte man vielleicht aber nicht, dass der Windows-Server, auf dem diese Anwendungen laufen, vom Internet aus erreichbar ist.

Das neunte Kapitel befasst sich mit dem Tuning eines bestehenden Systems. Wie kann man einen XAMPP-Server beschleunigen, ohne zu sehr in die Architektur des Systems eingreifen zu müssen? Es werden Möglichkeiten gezeigt, wie man Apache, PHP, MySQL und Perl schneller machen kann.

Im zehnten Kapitel geht es um das Thema Sicherheit. Wir gehen nicht nur auf die Frage ein, wie man sein System vor fremden und unerwünschten Zugriffen schützen kann, sondern auch auf oft vergessene Sicherheitsaspekte: Wie mache ich ein Backup von meinem System?

Nach der Absicherung von XAMPP wird im elften Kapitel erläutert, wie man einen XAMPP-Server übers Internet erreichbar macht, welche Probleme dabei auftreten können und wie man diese einfach lösen kann.

Der zweite Teil endet mit Kapitel 12, das Aspekte von XAMPP im Unterricht beleuchtet. Dieses Kapitel richtet sich an Lehrer und Dozenten, die XAMPP einsetzen wollen, um in Apache, PHP oder MySQL zu unterrichten.

Teil III

Der gesamte dritte Teil widmet sich dem Thema Datenbanken. Zunächst geht es in den Kapiteln 13 und 14 um die beiden Datenbanken, die bereits im XAMPP enthalten sind: MySQL und SQLite.

Die weiteren Kapitel beschäftigen sich mit den Datenbanken MS SQL, Oracle, Interbase/Firebird und PostgreSQL, und zwar stets nach dem gleichen Schema: Installation der Datenbank, Werkzeuge zu der Datenbank, Einrichten einer Beispieldatenbank, Zugriff auf die Datenbank von XAMPP aus. Das Ganze wird jeweils anhand eines Beispiels in PHP und in Perl veranschaulicht.

Teil IV

Themen des vierten Teils sind die Erweiterung und die Anpassung von XAMPP an die eigenen Bedürfnisse. Sie erfahren hier, wie Sie die Komponenten von XAMPP selbst übersetzen und ggf. um Bestandteile, die bisher nicht im XAMPP enthalten sind, erweitern können. Im 19. Kapitel wird gezeigt, wie man grundsätzlich Komponenten wie den Apache selbst übersetzt. In den anschließenden drei Kapiteln wird dargestellt, wie man PHP, Apache und Perl um zusätzliche Fähigkeiten oder Module erweitern kann.

Das 23. Kapitel behandelt das Thema XAMPP und Java. Die Installation von Tomcat und die Anbindung von XAMPP und Tomcat aneinander werden beschrieben.

Kapitel 24 zeigt beispielhaft an phpBB, Gallery, TYPO3, SquirrelMail und Joomla, wie man unter XAMPP andere Webapplikationen installiert.

In Kapitel 25 geht es um Nameserver und DNS. Mit Hilfe der Nameserver-Software BIND wird gezeigt, wie man einen eigenen Nameserver aufsetzt und ihn mit Domaindaten füttert.

Nachdem Sie in diesen Kapiteln erfahren haben, wie man XAMPP erweitert, lernen Sie schließlich im 26. Kapitel, wie Sie ein eigenes XAMPP-Paket schnüren und eigene Zusatzpakete für XAMPP zusammenstellen können.

Software auf der Buch-CD

Bis auf wenige Ausnahmen bei den kommerziellen Datenbanken sind alle in diesem Buch neben XAMPP beschriebenen Programme und Software-Pakete auch auf der beiliegenden CD-ROM enthalten. Auf der CD-ROM befinden sich auch genau die Versionen, die wir beim Schreiben dieses Buches verwendet haben. Zusätzlich haben wir aber auch immer auf die Originalquellen hingewiesen, von denen Sie sich die Software selbst herunterladen können.

Teil I

XAMPP-Grundlagen





1 Was ist eigentlich XAMPP?

XAMPP ist eine Serverumgebung zur Entwicklung und zum Betrieb von Websites auf dem eigenen Rechner. Der eigene PC wird so zum Server und damit unabhängig von Netzwerkverbindungen und anderen Rechnern in einem Netzwerkverbund.

Die Komponenten von XAMPP sind:

- der Webserver Apache
- das Datenbankmanagementsystem MySQL
- die Programmiersprachen Perl und PHP
- der FTP-Server ProFTPD (Linux, Mac OS X, Solaris) oder FileZilla Server (Windows)
- diverse Erweiterungen und Tools

Der Name XAMPP leitet sich aus den Komponenten Apache, MySQL, Perl und PHP ab. Das X am Anfang steht für ein beliebiges Betriebssystem. Zurzeit gibt es XAMPP für Windows, Linux, Mac OS X und Solaris.

1.1 Die klassische Client-Server-Architektur

Die Funktionsweise des WWW folgt der klassischen Client-Server-Architektur. Auf der einen Seite steht der Server, der eine bestimmte Aufgabe (Dienst) erfüllt, und auf der anderen Seite steht der Browser, das Programm, das der Benutzer benutzt, um auf den Dienst des Servers zuzugreifen.

In der Regel werden Client und Server auf unterschiedlichen Rechnern laufen. Im Falle von Webserver/Browser wird der Webserver – also der Apache – auf einem Server laufen, der irgendwo im Internet, in einem gekühlten Serverraum, steht. Und der Browser läuft zu Hause oder im Büro beim Endbenutzer.

Aber Client und Server müssen nicht auf unterschiedlichen Rechnern laufen und genau hier setzt XAMPP an: XAMPP bringt die Serverkomponenten des WWW auf den Heimrechner.

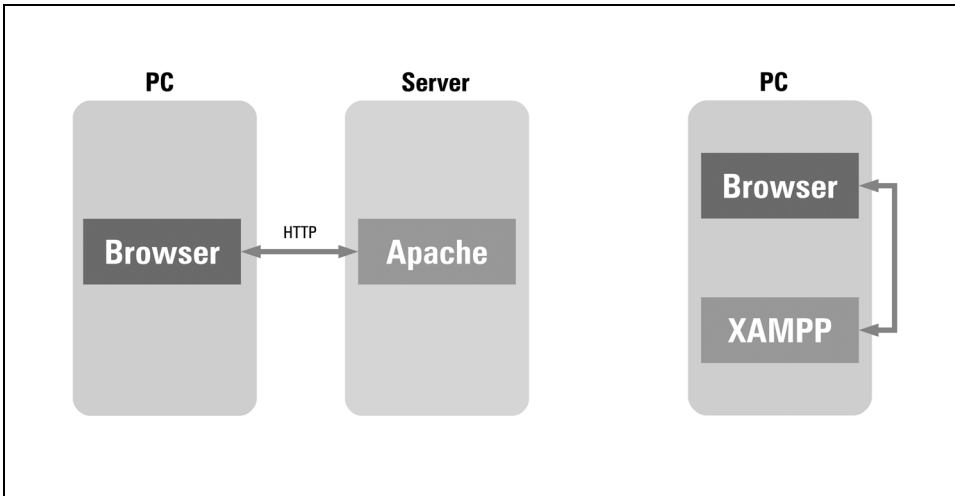


Abbildung 1.1: Auf der rechten Seite sieht man die klassische Client-Server-Architektur auf zwei unterschiedlichen Rechnern. Der Client läuft auf dem PC des Benutzers und der Server auf dem Serverrechner. XAMPP löst diese Trennung auf und bringt den Server mit auf den Rechner des Benutzers.

1.2 Auf der Serverseite

Nun ist XAMPP nicht nur ein einfacher Apache, sondern bringt auch noch andere Server und Serverkomponenten mit sich, die für den Betrieb eines Webservers benötigt werden. Abbildung 1.2 gibt einen kleinen Überblick über das Zusammenspiel der wichtigsten Komponenten von XAMPP.

Auf der linken Seite greift der Surfer, der Besucher der Website, auf die Daten der Website zu. Bei ihm läuft der Client, der Webbrowser, und auf der Serverseite kommuniziert er mit dem Apache, und zwar ausschließlich mit dem Apache. Von den restlichen Komponenten bekommt der Surfer nichts mit. Sie liegen für ihn verborgen »hinter« dem Apache.

Gleich nach dem Apache kommen die Programmiersprachen PHP und Perl. Mit ihnen ist die Webapplikation bzw. Website programmiert. Sicherlich gibt es noch andere Programmiersprachen wie Python, Java und Ruby, mit denen Webapplikationen programmiert werden, in diesem Fall beschränkt sich die Abbildung auf die Komponenten, die standardmäßig in XAMPP enthalten sind.

Die PHP- und Perl-Programme selbst liegen auf dem Dateisystem, also der Festplatte des Servers. Gleichzeitig greifen sie auch auf das Datenbankmanagementsystem (im Falle von XAMPP ist es MySQL) zu.

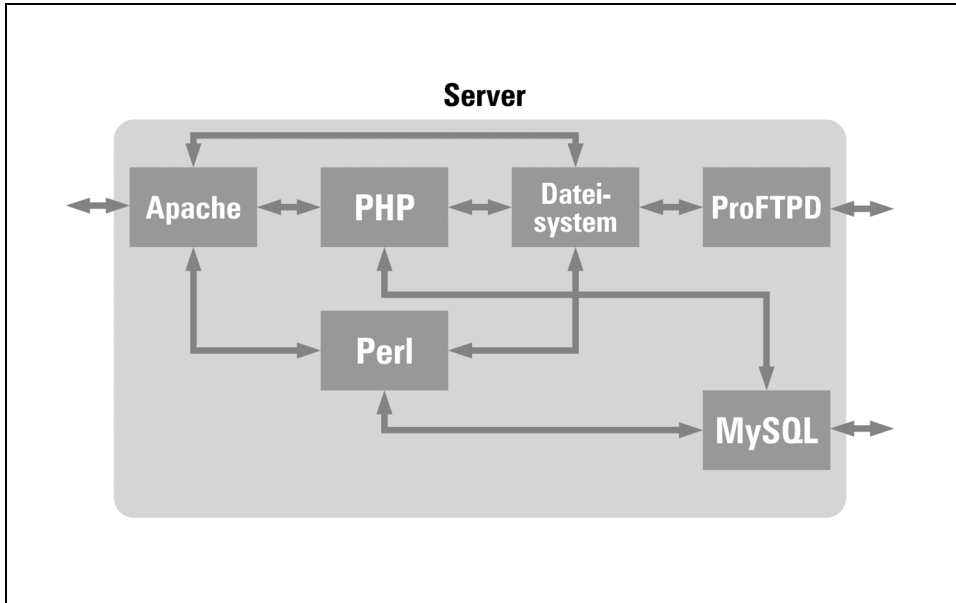


Abbildung 1.2: Auf den ersten Blick vielleicht ein wenig kompliziert: die Architektur von XAMPP auf der Serverseite, hier im Falle von XAMPP für Linux. Für XAMPP für Windows ersetzen Sie einfach ProFTPD durch FileZilla Server.

Auf der rechten Seite steht der Entwickler, Redakteur oder Grafiker der Website. Er greift zum Beispiel über FTP und den FTP-Server auf das Dateisystem zu und füttert so den Webserver mit HTML, Bildern, PHP- oder Perl-Programmen oder sonstigen Daten, die über den Webserver dem Surfer angeboten werden sollen.

Neben den Daten in Dateiform wird die Website evtl. noch mit Daten aus einer Datenbank gebildet. Die Datenbank wird von rechts mit Inhalten gefüllt, die dann später über PHP oder Perl an den Surfer weitergegeben werden.

1.3 Wer braucht XAMPP?

Die Zielgruppe von XAMPP ist überraschenderweise recht heterogen. Ursprünglich waren Entwickler, die sich nicht mit der Administration von Serverkomponenten beschäftigen möchten oder wollen, die Hauptzielgruppe von XAMPP. Im Laufe der Zeit haben aber auch andere Gruppen Nutzen an XAMPP gefunden:

Der Einsteiger

Der Einsteiger an sich: Er will sich entweder mit Webdesign, Webprogrammierung oder Administration beschäftigen. Für ihn bildet die Installation der einzelnen Komponenten von XAMPP eine unüberwindbare Schwelle und er benutzt XAMPP als Trittbrett, um einen Blick hinter die Kulissen des WWW zu werfen.

Der Webdesigner

HTML-Dateien und auch andere Inhalte von Websites verhalten sich auf einem Webserver anders, als wenn sie einfach nur vom Dateisystem aufgerufen werden. Das fängt bei absoluten/relativen Pfaden an, geht über Groß- und Kleinschreibung weiter und hört bei serverseitigen Programmiersprachen wie PHP auf. Auch wenn der Webdesigner vielleicht nicht gleich programmieren möchte, hat er einen Nutzen von einem lokalen Webserverssystem, wie XAMPP es bietet.

Der Webprogrammierer

Der Webprogrammierer programmiert Webapplikationen. Er braucht unbedingt einen Server, da seine Software nur innerhalb eines Webserver funktioniert. Allerdings mag er sich nicht mit der Administration der einzelnen Komponenten beschäftigen, sei es aus Zeitmangel oder aus Unwissenheit.

Der Administrator

Zwar ist XAMPP nur für die interne, nicht öffentliche Verwendung im Internet bestimmt. XAMPP ist einfach zu unsicher konfiguriert, als dass man es so, wie es ist, im Internet betreiben dürfte. Wer sich allerdings der sicherheitsrelevanten Einstellungen bewusst ist und auch weiß, wie man sie vornimmt, der kann XAMPP auch gerne im Internet betreiben. Der Vorteil von XAMPP, sehr einfach installierbar zu sein, ist auch für professionelle Administratoren verlockend. Warum auf komplizierte Weise selbst die Komponenten einzeln installieren und konfigurieren, wenn es doch so viel einfacher geht?

Der Dozent und Lehrer

Das Internet und, damit eng verbunden, das WWW sind die wichtigsten neuen Medien der letzten Jahrzehnte geworden. HTML oder Programmiersprachen wie PHP finden sich auf immer mehr Lehrplänen und werden in Schulen, Volkshochschulen, privaten Fortbildungsinstituten und Universitäten angeboten. Alle diese Einrichtungen sind in der Regel nicht mit Serverparks ausgestattet, um Schulungen auf diesem Gebiet durchzuführen, und verfügen in der Regel nur über normale (Windows-) PCs als Schulungsrechner. Daher ist XAMPP in diesen Fällen ein bequemes Werkzeug geworden, um aus einfachen Windows-Schulungsrechnern ein Webserverssystem zu machen.

1.4 Was macht XAMPP besonders?

Die Idee hinter XAMPP ist nicht neu. Es gibt oder gab andere Projekte, die eine ähnliche Lösung anbieten: FoxServ, NuSphere, PHPTriad, Sokkit, EasyPHP, phpdev und Uniform Server, um die bekanntesten zu nennen. Was macht aber XAMPP so besonders?

- Sehr einfach zu installieren und zu aktualisieren

Ein Ziel, das uns fast das wichtigste ist: XAMPP muss sich ganz einfach installieren lassen. Jede Frage, die der Benutzer bei der Installation stellt, ist uns ein Dorn im Auge. Ebenso haben wir versucht, das Aktualisieren von XAMPP so einfach wie möglich zu gestalten.

- Kostenlos erhältlich

XAMPP ist kostenlos und enthält auch keinerlei Werbung.

- Regelmäßig aktualisiert

Vielleicht der größte Vorteil, den XAMPP gegenüber den anderen Lösungen hat, dürfte die Aktualität des Pakets sein. Sobald es zwei neue Versionen der Hauptkomponenten von XAMPP (Apache, MySQL, Perl und PHP) gibt, erscheint auch eine neue Version von XAMPP.

- Große Community und Verbreitung

Hinter XAMPP stecken Millionen Anwender und eine große Community. In unserem Forum sind zurzeit (Juni 2006) ca. 18 000 Benutzer registriert. Im Schnitt sind davon 80 Benutzer gleichzeitig im Forum online. Seit September 2002 wurde XAMPP insgesamt 4 Millionen Mal heruntergeladen, allein im kurzen Februar 2006 250 972 Mal. Dies entspricht einem monatlichen Trafficvolumen von 11,8 TB (Quelle: XAMPP-Download-Statistiken bei Source Forge).

- Plattformübergreifend

Während es die anderen Apache-Bundles in der Regel nur für Windows gibt, unterstützt XAMPP zurzeit vier Betriebssysteme: Linux, Microsoft Windows (98SE, XP, 2000, 2003 Server), Mac OS X und Sun Solaris.

- Kapselung der webbasierten Anwendung

XAMPP kapselt die webbasierte Anwendung und deren Laufumgebung gegenüber dem Betriebssystem, d. h. unabhängig vom Betriebssystem bietet XAMPP:

- gleiche Pfade
- gleiche Versionen der Komponenten
- gleiche Konfiguration

Allein wenn man sich nur die verschiedenen Linux-Distributionen anschaut, dann hat jede Distribution ihre eigene Vorstellung davon, wo Konfigurationsdateien und Webserverinhalte zu liegen haben.

- XAMPP bringt neue Technik in alte Systeme

Gerade unter Linux gibt es ja eigentlich schon alle die Komponenten, die XAMPP mit sich bringt. Aber diese Komponenten sind oft veraltet, besonders dann, wenn es sich um Versionen von Distributionen handelt, die nicht mehr vom Hersteller unterstützt werden.

In solchen Fällen kann XAMPP diese Systeme mit aktuellen Komponenten versehen.

XAMPP läuft auf Distributionen ab Debian 2.2 (August 2000), SUSE 7.0 (Oktober 2000), RedHat 7 (September 2000) sowie allen neueren Distributionen.

■ Vereinfachtes Deployment

Durch die oben erwähnte Kapselung der Webanwendung gegenüber dem Betriebssystem ergibt sich auch ein vereinfachtes Deployment im Entwicklungsprozess. Eine Website, die unter einem neuen openSUSE 10.1 mit XAMPP entwickelt wurde, ist ohne irgendwelche Änderungen auf einem 6 Jahre alten Debian 2.2 mit XAMPP lauffähig.

1.5 Systemvoraussetzungen

Eigentlich ist XAMPP recht ressourcenfreundlich. Meist wird das Betriebssystem viel größere Systemvoraussetzungen haben als XAMPP selbst. Je mehr man mit XAMPP machen will, desto größer werden dann natürlich auch die Hardwareanforderungen an das System.

Unter normalen Bedingungen sollte aber folgende Hardwareausstattung mehr als ausreichend sein:

Prozessor	Pentium II, PowerPC G4, Sun SPARC
Speicher	128 Mbyte RAM, 150 Mbyte Plattenplatz
Betriebssystem	GNU/Linux, Mac OS X, Sun Solaris, Microsoft Windows (98SE, XP, NT, 2000, 2003 Server)

Tabelle 1.1: Systemvoraussetzungen von XAMPP



2 Installation

Die Installation eines *AMPP*-Systems kann manchmal recht aufwendig sein. Nicht so mit XAMPP! Wir beschreiben nachfolgend die Installation von XAMPP für die Plattformen Microsoft Windows und Linux. Los geht's!

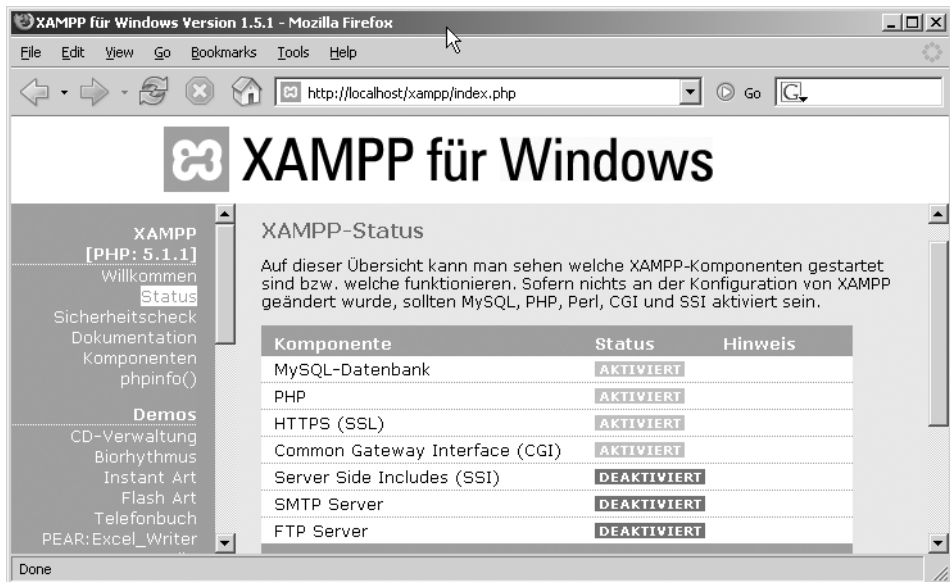


Abbildung 2.1: Der XAMPP-Status auf dem lokalen Rechner im Firefox-Browser

2.1 Windows

Der XAMPP für Windows gliedert sich grundlegend in ein Basispaket (den eigentlichen XAMPP) und den zusätzlichen *Add-Ons* auf. Der Begriff *Add-On* steht hier synonym für eine Erweiterung des Basispakets mit einem Extraprogramm. Diese Erweiterung kann auch ein weiterer Server sein wie beim *Tomcat-Add-On* oder einfach ein zusätzliches Programm wie beim *Perl-Add-On* mit dem kompletten Perl-Bundle. Die offiziellen *Add-Ons* von Apache Friends enthalten in der Regel nur Programme, die über ein *Third-Party-Modul* (d.h. Extramodul) des Apache steuerbar sind. Bei dem

Perl-Add-On ist es *mod_perl*, das mit dem enthaltenen Perl und dem aktuellen Apache im XAMPP übersetzt wurde. Ohne eine Übersetzung wäre die Gefahr recht groß, dass Sie das Modul nicht in den Apache-Prozess einpassen können. Darüber hinaus erscheint regelmäßig eine *Lite-Version* (»schlanke Version«) des XAMPP mit dem Namen *XAMPP Lite*. Diese kleine und damit insbesondere für USB-Sticks geeignete Version ist nicht (!) kompatibel mit den Add-Ons. Sie brauchen immer das normale Basispaket von XAMPP, um die Add-Ons einzuspielen.

Name	Windows 95	Windows 98	NT/2000/XP
XAMPP Basispaket	Unbekannt (benötigt winsocks2)	Ja	Ja
XAMPP Lite	Nein	Ja	Ja
XAMPP Add-Ons	Nein	Nein	Ja

Tabelle 2.1: Unter welchen Systemen funktioniert XAMPP?

Detaillierte Informationen zum Inhalt der aktuellen Releases oder den Add-Ons erhalten Sie immer in den entsprechenden Readme-Dateien.

2.1.1 Eine kleine Geschichte der Windows-Version

Wer die strukturellen Eigenheiten von XAMPP unter Windows verstehen möchte, sollte einen kurzen Ausflug in die jüngste Geschichte unternehmen. So manche Leser werden sich vielleicht noch erinnern. Der XAMPP hieß einmal WAMPP unter Windows bzw. LAMPP unter Linux. Als Windows-Variante gab es nur ein ZIP-Paket, das auf oberster Verzeichnishierarchie entpackt werden musste. Danach konnten die Server über Batch-Dateien sofort gestartet werden. Zugegeben, früher gab es im Paket weniger Server, PHP lief ausschließlich im CGI-Modus und an PEAR war noch gar nicht zu denken. Der Hauptmakel lag aber im Mangel der Flexibilität dieses sog. Bundles. Viele Anwender wollten den XAMPP ja nicht unbedingt auf der obersten Ebene ihrer Partition einsetzen, sondern klassischerweise z.B. unter *C:\Programme*.

Ein Problem und damit Grund genug für eine »Häuptlingssitzung« in einer der vielen Berliner Lokalitäten. Die beiden Autoren (Kai »Oswald« Seidler und Kay Vogelgesang) waren sich einig: Hier musste etwas geschehen. Ein Installer-System? Nein, das kam zu diesem Zeitpunkt nicht in Frage. Denn viele Anwender lobten ja gerade die Möglichkeit, den WAMPP nicht installieren zu müssen. Also schlug Oswald eine Umsetzung mit *awk* vor. Und Kay ergänzte den Vorschlag durch eine Einbettung in ein PHP CLI-Programm (CLI = *Command Line Interface*). Die *install.php* mit einer *awk*-Unterstützung war geboren. Damit war es dem Anwender nun möglich, über die *setup_xampp.bat* das Paket an einer beliebigen Stelle im Verzeichnissystem zu etablieren. Die *install.php* sorgte lediglich für die korrekten absoluten Pfade in den einzelnen Konfigurationsdateien.

Irgendwann hieß der WAMPP dann XAMPP. Und irgendwann kam auch die Installer-Frage erneut auf die Tagesordnung. Mit der zunehmenden Beliebtheit von XAMPP wuchs nämlich der Kreis der Anwender, die sich für eine zusätzliche Installer-Variante aussprachen. Das bedeutete eine erneute Sitzung, auf der wir uns auf den freien *Nullsoft Installer* als Basis festlegten. Als bald erschien dann zusätzlich die erste Installer-Version.

Danach wurde im Vorfeld jedes Erscheinungstermins kräftig gebastelt. Neue Add-Ons mussten erarbeitet werden, allein schon deshalb, weil das komplette Perl mit seinen über 100 Mbyte vielen einfach zu groß erschien. Die `install.php` wurde also mit neuem Code für eine Add-On-Unterstützung erweitert. Verschieben sollte man den XAMPP auch können. Gesagt, getan: Also auch dieser Wunsch wurde in der `install.php` berücksichtigt. Mittlerweile hat das Skript einen recht ansehnlichen Funktionsumfang und wurde damit zu einem kleinen Monstrum. Aus einer simplen PHP-awk-Lösung ist eine administrative Konsole geworden. Selbst wir tun uns manchmal schwer, im Quellcode durchzublicken. Dies ist ein typisches Beispiel für ein Programm, das als kleine Lösung gedacht war, dann aber durch immer weitere Erweiterungen zu einem kompletten System aufstieg. Trotzdem: Die Lösung über ein freies (d.h. nicht kompiliertes) Programm (Skript) ist aufgrund der Transparenz immer einer möglichen binären Lösung vorzuziehen. Und vielleicht hat ja auch der eine oder andere Anwender Ideen für seine eigene Anwendung in der `install.php` gefunden. Gerade das Beispielhafte von XAMPP finden wir Entwickler enorm wichtig.

Zugegeben, binäre Anteile gibt es auch im XAMPP. Da wäre wohl als Erstes das *XAMPP Control Panel* zu nennen, das netterweise als C-Programm für XAMPP über www.nat32.com entwickelt wurde. Als Zweites aufzuführen sind die `xampp_start.exe` und die `xampp_stop.exe`. Beide Dateien starten bzw. stoppen Apache und MySQL. Die Anforderung hierfür kam von Windows 98-Anwendern, die XAMPP mit einem Fenster über den *Autostart* automatisch starten und stoppen wollten. Eine mögliche Lösung über die Batch-Dateien würde dagegen immer zwei Fenster öffnen, und das war nicht erwünscht.

Zum Ende dieses Abschnitts möchten wir auf ein Phänomen hinweisen: das *vermeintliche XAMPP-Virus*. Enthält XAMPP also ein Virus oder anderen »Schädling«? Nein! Warum aber schlagen manche Virens Scanner (u.a. eine ältere Variante von McAfee) bei XAMPP Alarm? Der Grund liegt an dem Vorhandensein des Programms *PrcView* (`pv.exe`) von Igor Nys (früher war es übrigens die `kill.exe` von der *PC Welt*, auch hierfür übrigens Danke!) Die `pv.exe` ist ein »Schweizer Messer« für die Prozesssteuerung über die Konsole. Sie können Prozesse nach Attributen auflisten, identifizieren und wenn nötig auch eliminieren (»kill«). Linux-Anwender haben jetzt vielleicht einen Aha-Effekt. Die `pv.exe` ist in ihrer Funktionalität eine Mischung aus den Programmen `ps` und `kill` unter Linux. Aber ist sie auch gefährlich? Nein.

```

C:\WINNT\System32\cmd.exe
D:\Programme\Xampp\apache\bin\pv --help

PrcView v 3.7.2.5 command line utility by Igor Nys
Usage: pv -[<MODE>] -[<OPTIONS>] <ARGUMENTS>...-[<OPTIONS>]

Modes:
-h,-?    --help        display this help information
-k        --kill        kill PROCESS
-a        --activate    activate PROCESS
-c        --close       close (send WM_CLOSE) to the PROCESS
-m        --module      show modules used by specified PROCESS
-g        --getenv       get startup environment for the PROCESS
-p[nihr] --priority     set priority to "Normal", "Idle", "High", "Real Time"
[lbal]   "Below Normal" and "Above Normal" only on W2K or higher
-t        --tree        display process tree, -te for computer-readable format
-u        --usage       show processes that uses specified MODULE
-s        --summary     show MODULE usage summary

Options:
-f        --force       never prompt
-e        --extend      show additional information if available
-i        --id          use process ID instead of the PROCESS name
-q        --quiet       suppress headers and produce a tab-separated list
-d[time] --delay        delay time in milliseconds before executing command
-l[mask]  --long        include process command line
-w[mask]  --window      show only processes with visible windows, -e show hidden
-r        --repeat      repeat command in a cycle

Arguments can contain '*' and '?' wildcards.
Process return code (%ERRORLEVEL%) can be used in batch files
0 - process found, 1 - empty result set, 2 - programm error

Examples:
pv myprocess.exe          get process ID for myprocess.exe.

```

Abbildung 2.2: Das Tool PrcView von Igor Nys: Sieht so ein Virus aus?

Wozu braucht XAMPP einen Prozess-Kill? Weil der *Apache 2* bedauerlicherweise keine reguläre Stopp-Funktionalität bei einem Handstart (also ohne Dienstinstitution) bereithält. Zwar beendet sich der Apache 2 nach dem Schließen des Apache-Fensters in der Regel automatisch. Aber es kamen bei unseren Tests immer wieder Fälle vor, in denen sich der Apache-Prozess nicht selbst beenden wollte. In manchen Szenarien blockierte der nicht beendete Prozess eine weitere Programmausführung. Ein Problem, das wir schließlich mit einem zusätzlichen Tool für eine gezielte Prozesssteuerung lösen konnten. Übrigens ist das Programm PrcView ein starkes Werkzeug für den erfahrenen Windows-Administrator. Probieren Sie es doch mal aus!

2.1.2 Installation des XAMPP-Basispakets

XAMPP für Windows liegt in drei Versionen vor: als *ZIP*, als selbstextrahierende *7-zip-Exe* sowie als *Installer*. Der Gebrauch der *Installer*-Version ist der einfachste Weg zum Ziel und damit Einsteigern zu empfehlen.

XAMPP herunterladen

Laden Sie sich als Erstes eines der *XAMPP für Windows*-Pakete von der Apache Friends-Website herunter:

<http://www.apachefriends.org/de/xampp-windows.html>

Speichern Sie das XAMPP-Paket in einem beliebigen Verzeichnis auf Ihrem System. Optional überprüfen Sie nun den MD5-Hash, um sicherzustellen, dass die Datei unbeschädigt und vollständig heruntergeladen wurde. Hierzu können Sie das Programm *md5summer* verwenden (<http://www.md5summer.org>).

Installation des ZIP- oder 7-zip-Pakets

Extrahieren Sie das Paket in ein Verzeichnis Ihrer Wahl und rufen Sie in dem nun angelegten XAMPP-Verzeichnis die *setup_xampp.bat* auf.

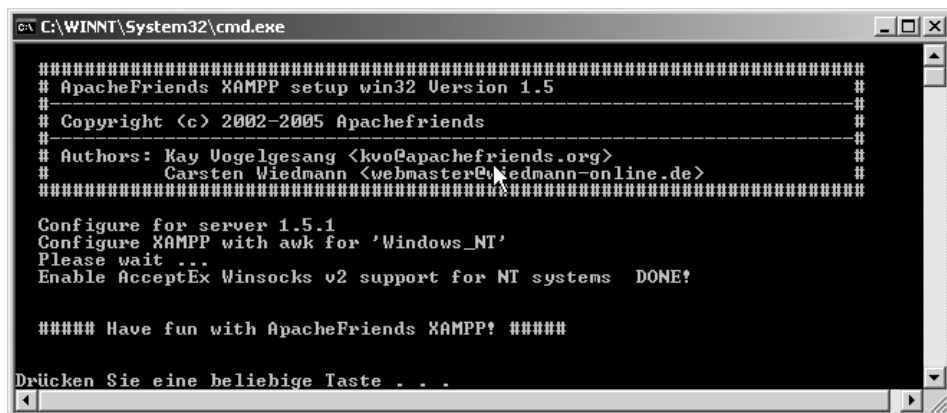


Abbildung 2.3: Die *setup_xampp.bat* nach der erfolgreichen Ausführung

Installation des Installer-Pakets

Starten Sie den Installer und wählen Sie den Zielpfad für die Installation aus.

Zum Abschluss der Installation beantworten Sie die Frage nach der Dienstinstallation. Wenn Sie sich hier nicht sicher sind, verneinen Sie diese Option. Es ist wirklich sehr einfach, die Server nachträglich als Dienstanwendung auf Ihrem System zu installieren!

XAMPP starten

Über das *XAMPP Control Panel* (*xampp-control.exe*) können Sie alle Server starten oder beenden und die Server auch als Dienst (hier als *Svc* für Service bezeichnet) einrichten.



Abbildung 2.4: Die Menügruppe von XAMPP unter PROGRAMME > APACHEFRIENDS

Wenn Sie die Installer-Version verwendet haben, können Sie zur Ausführung des XAMPP Control Panels auch die XAMPP-Desktop-Verknüpfung benutzen. Weitere Optionen stehen Ihnen dann auch unter **START > PROGRAMME > APACHEFRIENDS > XAMPP** zur Verfügung.

Als Alternative zu der GUI können Sie mit den einzelnen Batch-Dateien für Start und Stopp arbeiten. Sie finden die entsprechenden Dateien im XAMPP-Hauptordner. Dies sind sog. Handstarts im Unterschied zu Starts und Stopps von Diensten.

Die Batch-Dateien für die Installation bzw. Deinstallation von Diensten sind in den Hauptverzeichnissen der einzelnen Server enthalten. Sie können hierfür natürlich auch das XAMPP Control Panel (*xampp-control.exe*) benutzen.

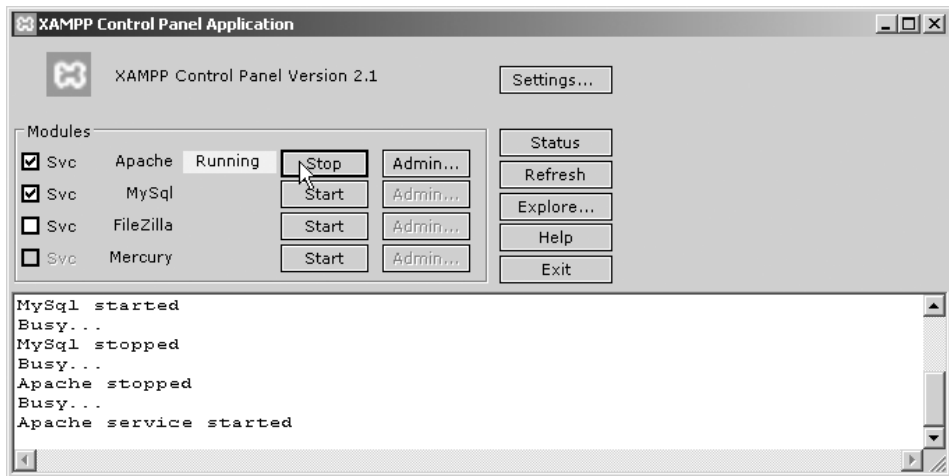


Abbildung 2.5: Bequem Dienste installieren und deinstallieren über die Option Svc

Achtung



Dienste können generell nur auf NT-Systemen installiert werden, das sind neben Windows NT die Systeme Windows 2000 sowie XP Professional.

Im Folgenden finden Sie eine Liste von Batch-Dateien für die Installation bzw. Deinstallation der verschiedenen Dienste:

- Apache-Dienst installieren: `apache\apache_installservice.bat`
- Apache-Dienst deinstallieren: `apache\apache_uninstallservice.bat`
- MySQL-Dienst installieren: `mysql\mysql_installservice.bat`
- MySQL-Dienst deinstallieren: `mysql\mysql_uninstallservice.bat`
- FileZilla-Dienst installieren: `filezilla_setup.bat`
- FileZilla-Dienst deinstallieren: `filezilla_setup.bat`

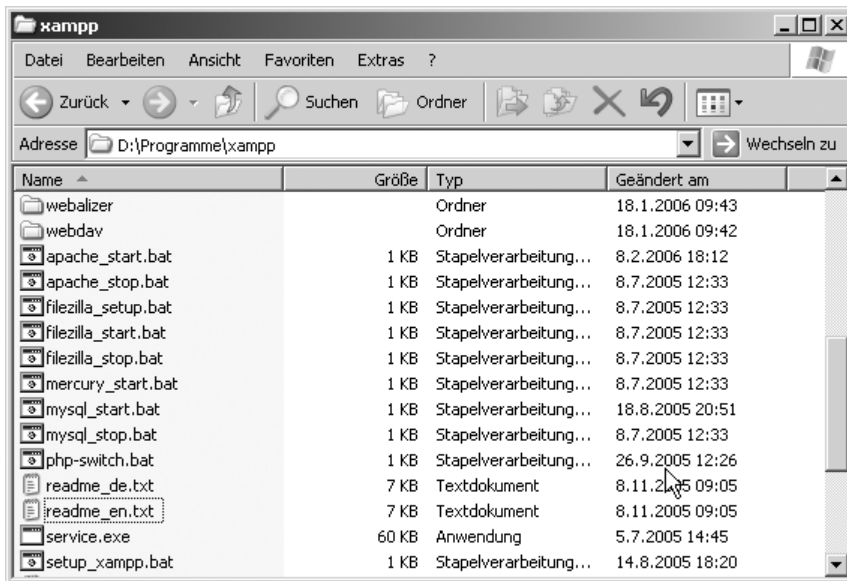


Abbildung 2.6: Die Start- und Stopp-Batch-Dateien im XAMPP-Hauptordner

XAMPP testen

O. K., das hört sich ja schon mal ganz gut an, aber woher wissen Sie nun, dass das alles wirklich funktioniert? Gut, dann greifen Sie einfach mal auf den Apache zu: Geben Sie dazu in die Adresszeile Ihres Webbrowsers die eigene URL ein:

`http://localhost`

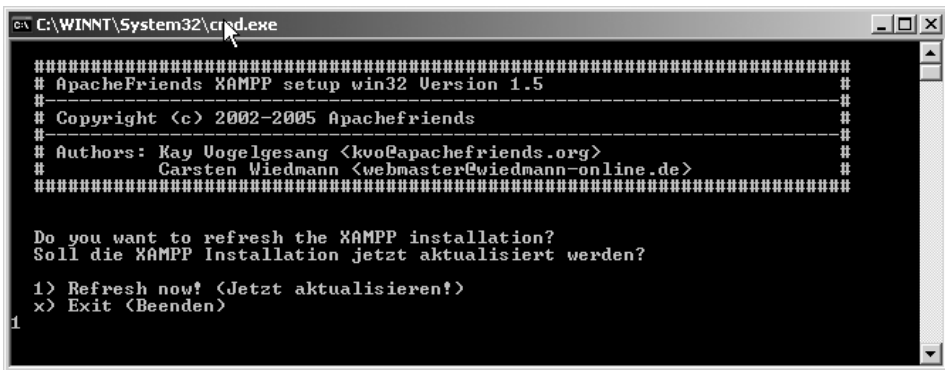
Hier meldet sich nun die Startseite von XAMPP. Hat man später diese Startseite durch seine eigene Seite ersetzt, kann man übrigens die XAMPP-Startseite immer noch unter `http://localhost/xampp/` erreichen.

Nach der Sprachauswahl finden Sie nun auf den XAMPP-Demoseiten einen Link zum Testen der Installation und ein paar kleine Programmbeispiele.

XAMPP verschieben

Den XAMPP zu verschieben ist zwar möglich, aber nicht ungefährlich! Gehen Sie dabei bitte mit der folgenden Methode auf Nummer sicher.

1. Deinstallieren Sie alle vorhandenen XAMPP-Dienste (nur die Dienste, damit ist nicht das Deinstallieren von XAMPP gemeint).
2. Kopieren Sie den gesamten XAMPP-Ordner in das Zielverzeichnis (nicht verschieben!).
3. Starten Sie in dem kopierten XAMPP-Ordner die `setup_xampp.bat` und wählen Sie Option 1 – *Refresh now! (Jetzt aktualisieren!)*. Alle Pfade werden damit angepasst.



```

C:\WINNT\System32\cmd.exe
#####
# ApacheFriends XAMPP setup win32 Version 1.5
#
# Copyright (c) 2002-2005 ApacheFriends
#
# Authors: Kay Vogelgesang <kvo@apachefriends.org>
#          Carsten Wiedmann <webmaster@wiedmann-online.de>
#####

Do you want to refresh the XAMPP installation?
Soll die XAMPP Installation jetzt aktualisiert werden?

1) Refresh now! (Jetzt aktualisieren!)
x) Exit (Beenden)
1

```

Abbildung 2.7: Mit der `setup_xampp.bat` alle Pfade in den Konfigurationen neu anpassen

4. Starten Sie die Server in der Kopie und testen Sie ausführlich!

Zum Schluss, wenn Sie sich sicher sind, dass die Kopie reibungslos funktioniert, können Sie den ursprünglichen XAMPP-Ordner löschen. Die Benutzer des XAMPP-Installers dürfen hingegen nicht das Verzeichnis löschen, sondern benutzen hierfür die Deinstallationsroutine.

XAMPP stoppen

Für die Beendigung der Serverprogramme benutzen Sie wahlweise das *XAMPP Control Panel* oder die verschiedenen *Stopp-Batch-Dateien* im XAMPP-Hauptordner. Für das Starten und Stoppen der Server bei einer Dienstinstallation können Sie darüber hinaus die *net-Befehle* auf der *Kommandozeile* einsetzen. Beispiele:

- Starten des Apache 2-Dienstes: `net start Apache2`
- Stoppen des Apache 2-Dienstes: `net stop Apache2`

XAMPP löschen

- 1. Deinstallieren Sie alle XAMPP-Dienste.
- 2. Entfernen Sie die XAMPP-Installer-Version über den Uninstaller.



Abbildung 2.8: Den XAMPP-Installer immer über den Uninstaller entfernen!

```
<xampp>\uninstall.exe
```

- 3. Löschen Sie den jetzt leeren XAMPP-Ordner über den Explorer.
Alle Benutzer der ZIP- bzw. 7-ZIP-Pakete löschen hingegen den XAMPP-Ordner über den Explorer.

2.1.3 Dynamische Installation auf Wechseldatenträgern

Im Zeitalter tragbarer USB-Datenträger bietet es sich an, den XAMPP transportabel zu machen. Der Trick ist hierbei, dass die Server Apache, MySQL und Mercury mit *relativen Pfadangaben* umgehen können.

Server	Absolute Pfade in Konfiguration	Relative Pfade in Konfiguration
Apache	Ja	Ja
MySQL	Ja	Ja
Mercury	Ja	Ja
FileZilla	Ja	Nein (!)

Tabelle 2.2: Welche Komponenten von XAMPP funktionieren auch mit relativen Pfaden?

Sie benötigen die ZIP- oder 7-ZIP-Version des XAMPP. Mit der Installer-Version funktioniert das dynamische Verfahren nicht! Der Wechseldatenträger sollte mindestens 200 Mbyte freien Speicherplatz für das XAMPP-Basispaket vorhalten. Der XAMPP Lite kommt hingegen schon mit 90 Mbyte aus.

1. Extrahieren Sie das XAMPP-Paket (Basis oder Lite) auf der obersten Hierarchie des Datenträgers bzw. Wechseldatenträgers. Wenn beispielsweise der USB-Stick den Laufwerksbuchstaben E: erhalten hat, sieht die Verzeichnisstruktur so aus:

```
E:\xampp  
E:\xampp\anonymous  
E:\xampp\apache  
[Und so weiter]
```

2. Achtung: Bitte führen Sie **nicht** `setup_xampp.bat` aus. Und die einzelnen Server auf dem Stick dürfen Sie ebenfalls *nicht als Dienst installieren*. Starten Sie die einzelnen Server also entweder über das Control Panel oder über die Batch-Dateien im XAMPP-Ordner. Stoppen Sie alle Server und beenden Sie alle weiteren XAMPP-Anwendungen, wenn Sie den USB-Stick entfernen möchten. Um den »XAMPP portable« zu testen, schließen Sie Ihren USB-Stick an einen weiteren Rechner an und starten die Server wie eben geschildert dynamisch über den erkannten Wechseldatenträger.

Tipp



Peppen Sie Ihr USB-XAMPP mit weiteren nützlichen Programmen auf!

Tipp 1: Der »Allround«-Editor *Weaverslave* (<http://www.weaverslave.ws>) braucht keine Installation und kann quasi »von überall aus« gestartet werden.

Tipp 2: Der *Firefox Portable* (<http://portablefirefox.mozdev.org>) bedarf ebenfalls keiner Installation. Der Vorteil eines eigenen Browsers auf dem Stick liegt vor allem in der konsistenten Präsentation Ihrer Projekte auf verschiedenen Systemen.

2.1.4 Installationen zusätzlicher XAMPP-Add-Ons

Die Möglichkeit einer Aufrüstung über weitere Add-Ons hängt stark davon ab, welches XAMPP-Release Sie benutzen. Auf der Apache Friends-Homepage finden Sie zu dem aktuellen Release des Basispakets in der Regel die dazu passenden Add-Ons. Anders ausgedrückt: Finden Sie hier keine, gibt es zu der aktuellen Version des XAMPP momentan keine Add-Ons. Diese erscheinen zeitlich verzögert zum XAMPP. Wir müssen diese Pakete dann erst einmal schnüren. In manchen Fällen ist aber auch die neue Apache- oder PHP-Version nicht kompatibel zum Quellcode des zu erstellenden Moduls. Erst wenn die Entwickler des entsprechenden Moduls eine angepasste Version erstellt haben, kann ein überarbeitetes Add-On erscheinen.

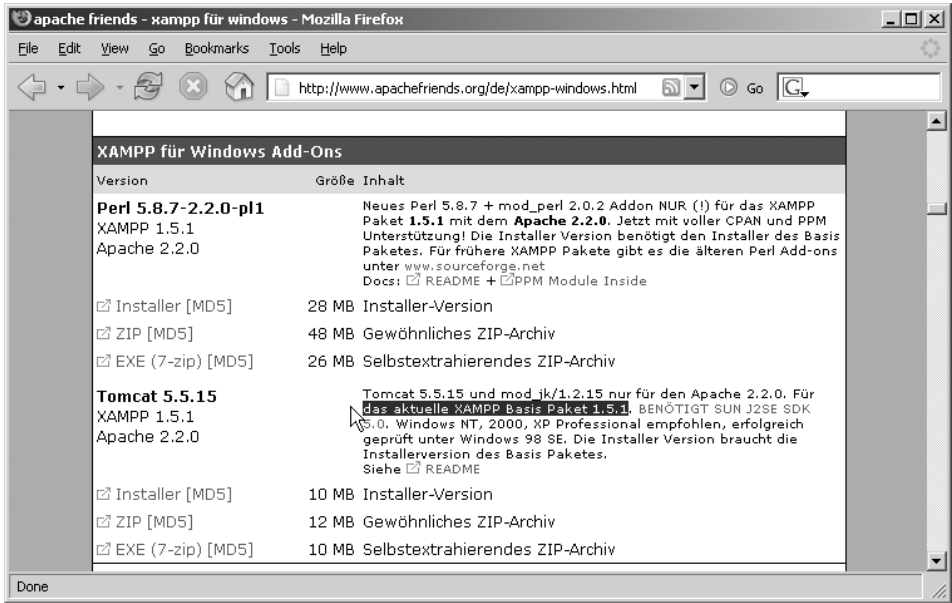


Abbildung 2.9: Zu jedem Add-On steht immer die passende Version des Basispakets.

Die offiziellen Add-Ons auf der Homepage korrespondieren mit den wichtigsten Third-Party-Anwendungen des Apache. Dazu gehören vor allem *Perl* mit *mod_perl* und der *Tomcat-Server* mit seinem *Connector mod_jk*.

Add-On	Inhalt	Bemerkung	Erscheint
Perl	Perl und mod_perl	–	Regelmäßig
Tomcat	Tomcat und mod_jk	Benötigt JDK von Sun	Regelmäßig
Python	Python und mod_python	–	Unregelmäßig
Cocoon	Cocoon	Abhängig von JDK	Unregelmäßig

Tabelle 2.3: Die offiziellen Add-Ons für die Windows-Version von XAMPP

Add-On-Installation mit dem Installer-Basispaket

Führen Sie das Setup des Add-Ons als *Installer-Version* aus. Es wird im Vorfeld geprüft, ob XAMPP in der korrekten Version für das Add-On installiert ist. Wenn keine kompatible XAMPP-Version gefunden wurde, wird die Installation abgebrochen. Starten Sie den Apache-Server bitte nach der Installation neu.

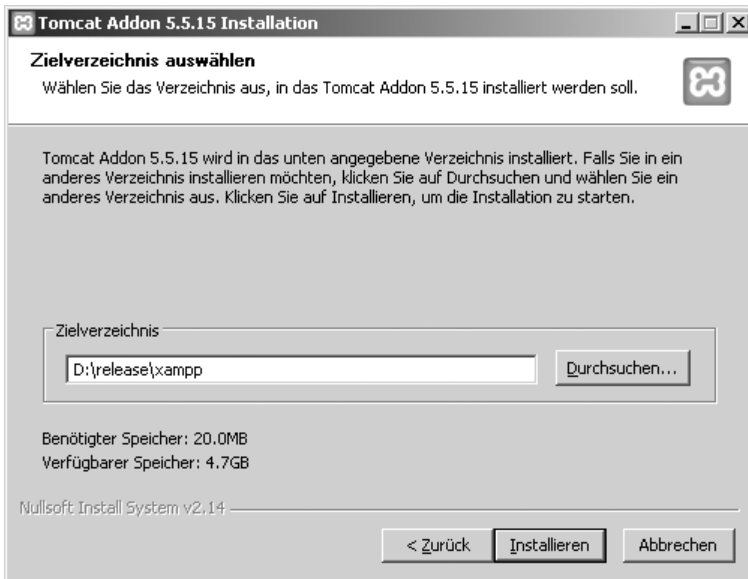


Abbildung 2.10: Wo auch immer der Installer XAMPP liegt, das Add-On findet ihn.

Add-On-Installation mit dem ZIP-/7-zip-Basispaket

1. Laden Sie das Add-On als ZIP- oder 7-zip-Paket herunter. Bitte versichern Sie sich zuvor über unsere Homepage (www.apachefriends.de), ob das von Ihnen gewählte Add-On mit Ihrer XAMPP-Version vereinbar ist.
2. Extrahieren Sie das Paket direkt in das XAMPP-Hauptverzeichnis. Überschreiben Sie ggf. dabei ältere XAMPP-Dateien.
3. Starten Sie im Hauptverzeichnis die `setup-xampp.bat`. Das Add-On wird damit initialisiert und installiert.
4. Starten Sie den Apache-Server neu.

Weitere Informationen entnehmen Sie selbstverständlich wieder den beiliegenden Readme-Dateien.

Hinweis



Die Erstellung eines XAMPP-Add-Ons ist kein Privileg der XAMPP-Entwickler. In Kapitel 26 erfahren Sie, wie Sie eigene XAMPP-Add-Ons erstellen können. Weitere interessante Third-Party-Add-Ons finden Sie beispielsweise unter <http://sourceforge.net/projects/xampp-Add-On>.

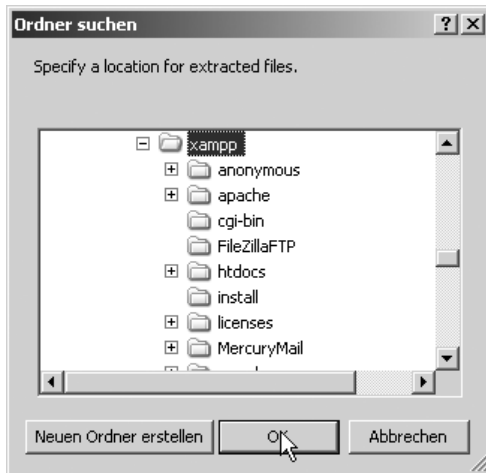


Abbildung 2.11: Add-Ons oder Upgrades werden immer direkt in den XAMPP-Ordner extrahiert.

2.1.5 Die Installation von Upgrades

Mit *XAMPP Upgrades* können Sie Ihre Version auf das aktuelle Release aufrüsten. Dabei werden nur Programmdateien ersetzt, keine Konfigurationsdateien. Das bedeutet in letzter Konsequenz auch, dass wir nicht immer ein Upgrade anbieten können. So geschehen mit dem XAMPP in der Version 1.5.1. Hier gab es strukturelle Änderungen im neuen Apache 2.2.0. Alte Konfigurationen der Apache 2.0.x-Reihe würde der Apache 2.2.0 zu einem gewissen Teil nicht mehr interpretieren können. In einem solchen Fall können wir Ihnen nur raten, das neue Basispaket zu installieren und Ihre Projekte dort einzupassen. Oder Sie bleiben einfach bei Ihrer alten Version. Nicht alles Neue muss auch automatisch besser sein!

Die Installation erfolgt analog zu der Installation der Add-Ons. Dabei sollten Benutzer der XAMPP-Installer-Version die Installer-Version des Upgrades benutzen.

2.1.6 XAMPP Lite-Besonderheiten

Der *XAMPP Lite* besteht aus folgenden Komponenten:

- Apache
- MySQL
- OpenSSL
- PHP 5 (ohne PEAR)
- phpMyAdmin

Die Lite-Version beansprucht 90 Mbyte auf der Festplatte. Apache- und MySQL-Server können über den sog. Handstart in der Regel mit einfachen Benutzerrechten ausgeführt werden. Für diese Version gibt es allerdings keine offiziellen Add-Ons auf der Apache Friends-Homepage.

2.2 Linux

Wer sich die Installation von XAMPP unter Windows auf den vorherigen Seiten angeschaut hat, der könnte schon Angst bekommen, wenn er sich vorstellt, wie das nur unter Linux aussehen wird. Oft ist das Installieren von Software (die nicht Bestandteil der eigenen Distribution ist) unter Linux komplizierter als unter Windows. Aber in diesem Fall ist die Furcht unbegründet: Unter Linux ist die Installation sogar noch einfacher als unter Windows. Das liegt vor allem daran, dass Sie unter Linux nicht die Wahl zwischen verschiedenen Installationsvarianten haben: Es gibt nur eine.

2.2.1 Installation

Bevor Sie mit der eigentlichen Installation beginnen können, müssen Sie das XAMPP-Paket von der Buch-CD kopieren oder aus dem Internet herunterladen.

XAMPP herunterladen

Laden Sie sich als Erstes das XAMPP für Linux-Paket von der Apache Friends-Website herunter:

<http://www.apachefriends.org/de/xampp-linux.html>

Speichern Sie das XAMPP-Paket in einem beliebigen Verzeichnis auf Ihrem System.



Achtung

Sollte der Browser Ihnen irgendwelche Archive Manager zum Öffnen des Downloads anbieten, dann lehnen Sie dies ab. XAMPP kann nur auf dem Weg installiert werden, wie es hier beschrieben wird. Variationen dieses Weges werden in der Regel zum Scheitern führen.

Sollten Sie XAMPP für Linux über einen Windows-Rechner herunterladen, dann ist dies auch kein Problem. Verwenden Sie aber niemals irgendwelche Windows-Entpacker, um XAMPP auszupacken. Kopieren Sie das XAMPP-Paket so, wie es ist, auf Ihren Linux-Rechner und fahren Sie dann dort mit der Installation, wie sie hier beschrieben ist, fort.

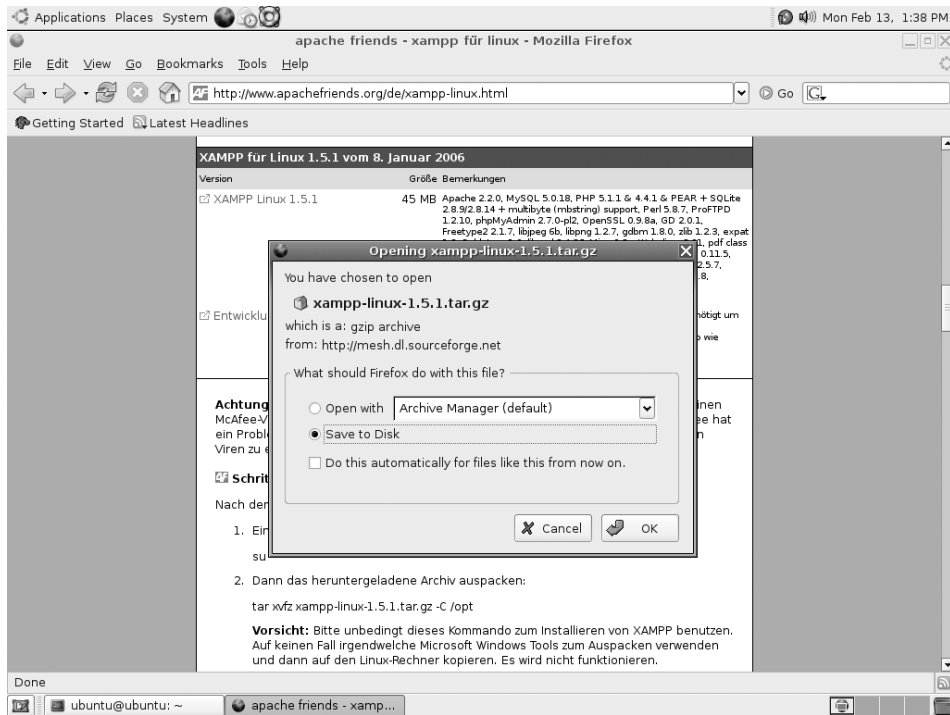


Abbildung 2.12: Auch wenn der Browser es vorschlägt: Verwenden Sie niemals irgendwelche anderen Entpackwerkzeuge außer tar. XAMPP wird mit größter Wahrscheinlichkeit nicht funktionieren.

Installation

1. Um XAMPP zu installieren, benötigen Sie Administratorrechte. Rufen Sie eine Terminal-Shell auf und werden Sie mit dem `su`-Befehl zum Administrator:

```
su
```

2. Danach wechseln Sie mit `cd` in das Verzeichnis, in das Sie das XAMPP-Paket heruntergeladen haben.
3. Rufen Sie nun den folgenden Befehl auf, um XAMPP zu installieren. Bitte achten Sie bei diesem Befehl unbedingt auf Groß- und Kleinschreibung.

```
tar xvfz xampp-linux-1.5.1.tar.gz -C /opt
```

Nun werden seitenweise Ausgaben über Ihren Bildschirm rollen. Das ist richtig so. Dieser Vorgang wird – je nach Ihrer Hardwareausstattung – ein paar Sekunden dauern.

XAMPP starten

Zum Starten von XAMPP tippen Sie einfach folgenden Befehl ein:

```
/opt/lampp/lampp start
```

Auf dem Bildschirm sollten nun folgende Mitteilungen zu sehen sein:

```
Starte XAMPP für Linux 1.5.1...
XAMPP: Starte Apache mit SSL...
XAMPP: Starte MySQL...
XAMPP: Starte ProFTPD...
XAMPP gestartet.
```

Fertig! XAMPP bzw. Apache, MySQL und ProFTPD laufen jetzt auf dem System.



Abbildung 2.13: Download, Installation und Start von XAMPP noch einmal in einem kleinen Screenshot zusammengefasst. Einfacher und schneller kann es nicht gehen.

XAMPP testen

O. K., das hört sich ja schon mal ganz gut an, aber woher wissen Sie nun, dass das alles wirklich funktioniert? Gut, dann greifen Sie einfach mal auf den Apache zu: Geben Sie dazu in die Adresszeile Ihres Webbrowsers die eigene URL ein:

`http://localhost`

Hier meldet sich nun die Startseite von XAMPP. Hat man später diese Startseite durch seine eigene Seite ersetzt, kann man übrigens die XAMPP-Startseite immer noch unter `http://localhost/xampp/` erreichen.

Nach der Sprachauswahl finden Sie nun auf den XAMPP-Demoseiten einen Link zum Testen der Installation und ein paar kleine Programmbeispiele .

XAMPP stoppen

Zum Stoppen von XAMPP geben Sie den folgenden Befehl ein:

```
/opt/lampp/lampp stop
```




Abbildung 2.14: XAMPP wurde erfolgreich gestartet und begrüßt Sie freundlich.

Auf dem Bildschirm sollten Sie nun folgende Meldungen sehen:

```
Stoppe XAMPP für Linux 1.5.1...
XAMPP: Stoppe Apache...
XAMPP: Stoppe MySQL...
XAMPP: Stoppe ProFTPD...
XAMPP beendet.
```

Und damit wurde XAMPP beendet.

XAMPP löschen

Sollten Sie eines Tages XAMPP löschen bzw. deinstallieren wollen, dann stoppen Sie zunächst XAMPP und löschen danach einfach das `/opt/lampp`-Verzeichnis, zum Beispiel mit dem folgenden Befehl:

```
rm -rf /opt/lampp
```

Damit ist XAMPP wieder komplett von Ihrer Festplatte verschwunden.

2.2.2 Die Installation von Upgrades

Wenn eine neue XAMPP-Version erscheint, stellen wir in den meisten Fällen auch ein sog. Upgrade-Paket zur Verfügung. Nur in ganz seltenen Fällen, in denen ein automatisiertes Upgraden aus technischen Gründen nicht möglich ist, gibt es kein solches Upgrade-Paket.

Sofern ein Upgrade angeboten wird, gibt es zwei gute Gründe, das Upgrade-Paket zu verwenden:

1. Zum einen ist die XAMPP-Distribution recht umfangreich und darum empfiehlt sich bei einer schon bestehenden XAMPP-Installation das viel kleinere Upgrade-Paket.
2. Der zweite Grund ist eigentlich sogar noch wichtiger: Bei einem Upgrade werden bestehende Konfigurations- und HTML-Dateien nicht überschrieben, was sonst bei einer normalen Installation der Fall wäre.

Um eine bestehende *XAMPP für Linux*-Installation zu aktualisieren, gehen Sie folgendermaßen vor:

1. Laden Sie zunächst das entsprechende Upgrade-Paket von <http://www.apache-friends.org/de/xampp-linux.html> herunter. Achten Sie darauf, das zur installierten XAMPP-Version passende Paket herunterzuladen.

Wohin Sie das Paket herunterladen, spielt keine Rolle. Nur muss es für den nächsten Schritt irgendwo auf dem Linux-System liegen, auf dem Sie XAMPP installiert haben.

2. Dann entpacken Sie das Upgrade-Paket:

```
tar xvfz xampp-linux-upgrade-x.x.x-y.y.y.tar.gz
```

Durch diesen Schritt wird ein Verzeichnis mit dem Namen `xampp-upgrade` erzeugt.

3. Nach dem Entpacken kann nun der eigentliche Upgrade-Vorgang gestartet werden. Dazu müssen Sie allerdings `root`-Rechte haben. Wenn Sie noch nicht als `root` angemeldet sind, dann holen Sie dies nun mit dem `su`-Befehl nach:

```
su
```

4. Starten Sie danach den Upgrade-Vorgang mit folgendem Befehl:

```
xampp-upgrade/start
```

Nun wird etwa folgender Text auf Ihrem Bildschirm erscheinen:

```
Stoppe XAMPP 1.5.2 fuer Linux...
XAMPP: Stoppe Apache mit SSL...
XAMPP: Stoppe MySQL...
XAMPP: Stoppe ProFTPD...
XAMPP beendet.
```

```
Aktualisiere LAMPP 1.5.2 auf 1.5.3...  
Aktualisiere bestehende Dateien...  
patching file lampp/etc/httpd.conf
```

Aktualisierung erfolgreich abgeschlossen.

```
Starte XAMPP 1.5.3 fuer Linux...  
XAMPP: Starte Apache mit SSL (und PHP5)...  
XAMPP: Starte MySQL...  
XAMPP: Starte ProFTPD...  
XAMPP fuer Linux gestartet..
```

Das war es auch schon. XAMPP wurde auf die neue Version aktualisiert.

Durch das Upgraden wird ein evtl. laufender XAMPP gestoppt und nach dem Upgrade wieder gestartet. XAMPP wird aber auch dann gestartet, wenn es vorher nicht lief.

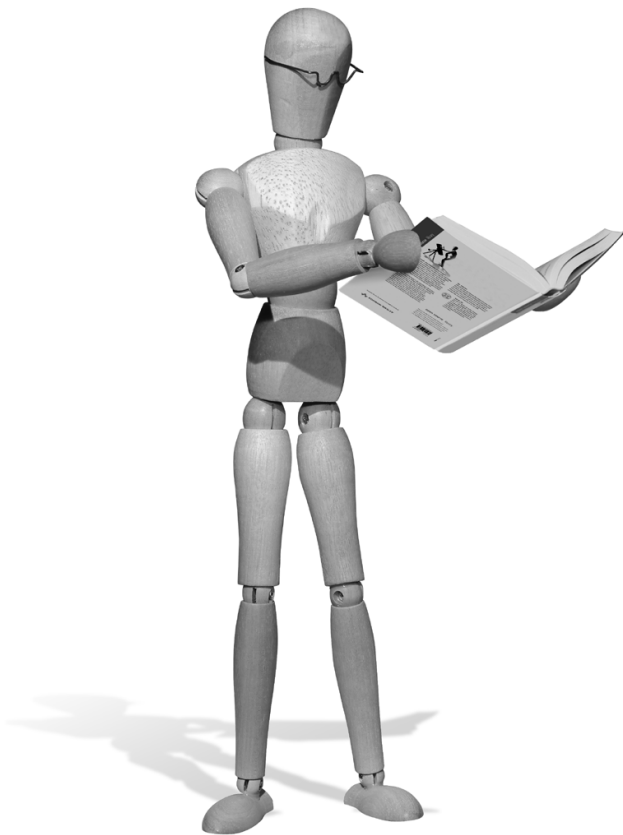
2.3 Das Thema Sicherheit

Wie schon eingangs erwähnt, ist XAMPP nicht für den Produktionseinsatz (also den Einsatz im Internet) gedacht, sondern nur für Entwickler in Entwicklungsumgebungen. XAMPP ist daher so konfiguriert, dass es besonders einfach für den Entwickler zu benutzen ist, es ist aber überhaupt nicht sicher konfiguriert.

Wenn Sie XAMPP in irgendeinem – auch noch so kleinen – Netzwerk verwenden, dann sollten Sie Ihre XAMPP-Installation unbedingt absichern. Lesen Sie hierzu auch das Kapitel 10, »Sicherheit«.

Teil II

Fortgeschrittene Benutzung von XAMPP





3 Apache

Seit der Version *XAMPP 1.5.1* mit *Apache HTTPD 2.2.0* ist die Konfiguration der `httpd.conf` in weitere Include-Dateien thematisch aufgeteilt. Zuvor war der Großteil der Konfiguration in der `httpd.conf` zu finden. Das hat sich seit der 2.2er-Version des Apache geändert.

Datei	Konfigurationsbereich	Verzeichnis
<code>httpd.conf</code>	Grundlegende Servereinstellung	<code>/opt/lampp/etc</code>
<code>httpd-autoindex.conf</code>	Index und Icons	<code>/opt/lampp/etc/extra</code>
<code>httpd-dav.conf</code>	WebDAV	<code>/opt/lampp/etc/extra</code>
<code>httpd-default.conf</code>	Standardeinstellungen	<code>/opt/lampp/etc/extra</code>
<code>httpd-info.conf</code>	Serverstatus und Info	<code>/opt/lampp/etc/extra</code>
<code>httpd-languages.conf</code>	Sprach-, Ländereinstellung	<code>/opt/lampp/etc/extra</code>
<code>httpd-manual.conf</code>	Apache-Manual-Konfiguration	<code>/opt/lampp/etc/extra</code>
<code>httpd-mpm.conf</code>	Server-Pool Management (MPM)	<code>/opt/lampp/etc/extra</code>
<code>httpd-multilang-errordoc.conf</code>	Mehrsprachige HTTP-Error-Dokumente definieren	<code>/opt/lampp/etc/extra</code>
<code>httpd-ssl.conf</code>	SSL/HTTPS-Konfiguration	<code>/opt/lampp/etc/extra</code>
<code>httpd-userdir.conf</code>	Standardisierte Benutzerpfade	<code>/opt/lampp/etc/extra</code>
<code>httpd-vhosts.conf</code>	Virtual Hosts einrichten	<code>/opt/lampp/etc/extra</code>
<code>httpd-xampp.conf</code>	XAMPP-spezifische Einstellungen	<code>/opt/lampp/etc/extra</code>

Tabelle 3.1: Apache-Konfigurationsdateien im XAMPP für Linux ab Version 1.5.1

Sie können diese Tabelle für *XAMPP für Windows* einfach gedanklich übertragen. Dabei ist der Linux-Pfad

`/opt/lampp/etc/extra`

gleich dem Windows-Pfad

`<xampp>\apache\conf\extra`

Hinweis



Die Inhalte vieler der hier vorgestellten »inkludierten« Konfigurationsdateien handeln von sehr spezifischen Einstellungen. Aus diesem Grund ist das Einlesen dieser Dateien in der `httpd.conf` oftmals auskommentiert.

3.1 Virtual Hosts

Ein physikalischer Rechner kann mehrere Virtual Hosts mit eigener Domain beherbergen. Sie kennen das sicherlich. Denn die meisten einschlägigen und preiswerten Hosting-Pakete bekannter Provider beruhen auf diesem Prinzip. Ein Server (d.h. Rechner) verwaltet viele Kunden, alle mit mindestens einer, wenn nicht sogar mehreren Domains. Alle Kunden besitzen für ihre Dokumente auf dem (einen) Server einen separaten Speicherplatz mit eigenem Login (meistens per FTP). Dort legt der Kunde die Webdokumente für den Auftritt seiner Domain ab. Doch wieso kann der Apache bei einer Anfrage eine Domain ihrem Stammverzeichnis richtig zuordnen, wenn er mehrere Domains verwaltet? Natürlich durch die Virtual Hosts. Bei den hier vorgestellten Virtual Hosts handelt es sich einzig um *HTTP*-Hosts für den Apache. Ein FTP-Server wie ProFTPD kann für sein Protokoll eigene Virtual Hosts bereitstellen, die nur beim FTP-Transfer wirken, nicht aber bei einer HTTP-Anfrage.

Prinzipiell unterscheidet der Apache-Webserver IP-basierte Virtual Hosts (*IP-based virtual hosts*) von namensbasierten Virtual Hosts (*name-based virtual hosts*). Beim IP-basierten Virtual Host besitzt jede Website ihre eigene IP-Adresse. Ein namensbasierter Virtual Host besitzt dagegen keine dedizierte, eigene IP-Adresse, sondern teilt sich eine IP-Adresse mit anderen Websites. D. h., mehrere Websites benutzen dieselbe IP-Adresse. Die Aufschlüsselung zu den verschiedenen Websites erledigt der Apache dann eigenständig. Der Trick: Ab HTTP/1.1 sendet der Client (meist der Browser) den Namen der gesuchten Website in dem HTTP-Host-Header mit.

Als Beispiel stellt dieses Kapitel die Einrichtung eines IP-basierten Virtual Hosts und die zweier namensbasierter Virtual Hosts für jeweils eine Website vor.

3.1.1 Namensauflösung

Die Auflösung des Domainnamens zu der IP-Adresse (*resolving*) wird häufig mit dem *Domain Name System* (DNS) in Verbindung gebracht. Tatsächlich stellen die öffentlichen DNS-Server das Rückrat der Auflösungsarbeit im Internet dar.

Die Übersetzung des Namens in die IP-Adresse kann für den Hausgebrauch, z. B. im privaten Intranet, auch über die `hosts`-Datei realisiert werden. Die Hosts finden Sie unter:

- Linux: `/etc/hosts`
- Windows NT/XP/2000: `C:\WINDOWS\system32\drivers\etc\hosts`
- Windows 95/98: `C:\WINDOWS\hosts`

Um dieses Beispiel nicht unnötig kompliziert zu machen, arbeiten wir hier mit der `hosts`-Datei. Für die Umsetzung der Auflösung über einen DNS-Server lesen Sie bitte Kapitel 25, »Der BIND-Nameserver«.

Ändern Sie Ihre `hosts`-Datei, indem Sie die folgenden Zeilen am Ende hinzufügen:

```
127.0.0.1      localhost1
127.0.0.2      localhost2 localhost3
```

Das war es. Und keine Angst, Ihr System funktioniert immer noch tadellos. Wir haben ja nur der *loopback*-Adresse drei Namen zugefügt: *localhost1*, *localhost2* und *localhost3*. Es sind zwar keine schönen Namen, aber sehr funktionale. Apropos: Das Funktionieren der neuen Namen können Sie mit *ping* gleich auf Ihrer Shell bzw. Eingabeaufforderung testen:

```
ping localhost1
ping localhost2
ping localhost3
```

3.1.2 httpd-vhosts.conf mit Virtual Hosts

Wie eingangs bereits erwähnt, wurden Teile der zuvor zentralen `httpd.conf` seit dem *Apache 2.2* in einzelne Konfigurationsdateien im Unterordner `extras` ausgelagert. Die für die Einstellung von Virtual Hosts vorgesehene Konfigurationsdatei heißt `httpd-vhosts.conf`.

- Linux: `/opt/lampp/etc/extra/httpd-vhosts.conf`
- Windows: `<xampp>\apache\conf\extra\httpd-vhosts.conf`

Prüfen Sie unbedingt, ob in der `httpd.conf` die Datei `httpd-vhosts.conf` über die `Include`-Anweisung geladen wird, also ob dort die folgende Zeile steht und nicht auskommentiert ist:

```
Include conf/extra/httpd-vhosts.conf
```

Zu Beginn arbeiten Sie mit der Direktive `NameVirtualHost`. Mit dieser Anweisung definieren Sie, dass es sich bei der Adresse `127.0.0.2` um eine »Sammel-IP-Adresse« für namensbasierte Virtual Hosts handelt, also um Virtual Hosts, die sich anhand ihres Namens und nicht anhand ihrer IP-Adresse unterscheiden.

Danach folgen die `VirtualHost`-Container für *localhost1*, *localhost2* und *localhost3*. Die Anweisung `ServerName` im zweiten und dritten Container kennzeichnet dann den entsprechenden Domainnamen. Das Dokumentenverzeichnis wird schließlich über `DocumentRoot` definiert.

Für dieses Beispiel haben wir ein neues Verzeichnis namens `/opt/lampp/vhosts` bzw. `<xampp>\vhosts` mit den folgenden Unterverzeichnissen erstellt.

Unter Linux:

```
/opt/lampp/vhosts/localhost1/htdocs
/opt/lampp/vhosts/localhost1/logs
/opt/lampp/vhosts/localhost2/htdocs
/opt/lampp/vhosts/localhost2/logs
/opt/lampp/vhosts/localhost3/htdocs
/opt/lampp/vhosts/localhost3/logs
```

Unter Windows:

```
C:\Programme\XAMPP\vhosts\localhost1\htdocs
C:\Programme\XAMPP\vhosts\localhost1\logs
C:\Programme\XAMPP\vhosts\localhost2\htdocs
C:\Programme\XAMPP\vhosts\localhost2\logs
C:\Programme\XAMPP\vhosts\localhost3\htdocs
C:\Programme\XAMPP\vhosts\localhost3\logs
```

Wobei in den Unterverzeichnissen `htdocs` jeweils eine unterscheidbare Startseite (z. B. `index.html`) für die drei Virtual Hosts liegt.

	localhost1	localhost2	localhost3
IP-Adresse	127.0.0.1	127.0.0.2	127.0.0.2
DocumentRoot	localhost1/htdocs/	localhost2/htdocs/	localhost3/htdocs/
Log-Dateien	localhost1/logs/	localhost2/logs/	localhost3/logs/

Tabelle 3.2: Ordnerstruktur unter `/opt/lampp/vhosts/` bzw. `<xampp>\vhosts\`

Hinweis



Liebe Linux-Benutzer: Übersetzen Sie bitte das folgende Beispiel für Windows in das Linux-Verzeichnissystem. Liegt der XAMPP unter `C:\Programme\xampp`, heißt das für Sie also `/opt/lampp`.

Und liebe Windows-Benutzer: Für den Apache existiert unter Windows immer nur das Slashzeichen (`/`) zum Auszeichnen der Pfadstrukturen, also `C:/Programme/xampp` usw.

Tragen Sie diese Zeilen in die `httpd-vhosts.conf` ein:

Listing 3.1: Ein IP-basierter Virtual Host für `localhost1` und zwei namensbasierte Virtual Hosts für `localhost2` und `localhost3`

```
NameVirtualHost 127.0.0.2

<VirtualHost 127.0.0.1:80>
    ServerAdmin webmaster@localhost1
```

```

DocumentRoot C:/Programme/xampp/vhosts/localhost1/htdocs
ErrorLog C:/Programme/xampp/vhosts/localhost1/logs/error_log
CustomLog C:/Programme/xampp/vhosts/localhost1/logs/access_log common
</VirtualHost>

<VirtualHost 127.0.0.2:80>
    ServerAdmin webmaster@localhost2
    DocumentRoot C:/Programme/xampp/vhosts/localhost2/htdocs
    ServerName localhost2
    ErrorLog C:/Programme/xampp/vhosts/localhost2/logs/error_log
    CustomLog C:/Programme/xampp/vhosts/localhost2/logs/access_log common
</VirtualHost>

<VirtualHost 127.0.0.2:80>
    ServerAdmin webmaster@localhost3
    DocumentRoot C:/Programme/xampp/vhosts/localhost3/htdocs
    ServerName localhost3
    ErrorLog C:/Programme/xampp/vhosts/localhost3/logs/error_log
    CustomLog C:/Programme/xampp/vhosts/localhost3/logs/access_log common
</VirtualHost>

```



Achtung

Weitere Verzeichnis-Zugriffsrechte für XAMPP für Windows

Unter XAMPP für Windows müssen Sie dem Apache noch erlauben, neben dem normalen `htdocs`-Verzeichnis von XAMPP auch auf andere Verzeichnisse zugreifen zu dürfen. Ändern Sie dazu die Datei

`<xampp>\apache\conf\httpd.conf` folgendermaßen:

Suchen Sie in der Datei die folgenden Zeilen:

```

<Directory />
    Options FollowSymLinks
    AllowOverride None
    Order deny,allow
    Deny from all
</Directory>

```

Setzen Sie vor die letzten beiden Zeilen in diesem Directory-Container jeweils ein Doppelkreuz:

```

<Directory />
    Options FollowSymLinks
    AllowOverride None
    #Order deny,allow
    #Deny from all
</Directory>

```

Das war's. Damit haben Sie ermöglicht, dass die Virtual Hosts auch außerhalb des normalen `htdocs`-Verzeichnis von XAMPP liegen dürfen.

Starten Sie den Apache neu. Danach kommt der obligatorische Test: Rufen Sie in Ihrem Browser `http://localhost1`, `http://localhost2` und `http://localhost3` nacheinander auf. Sie sollten nun jeweils die Startseite des jeweiligen Virtual Host sehen.

Aufgrund der Loopback-Adressen 127.0.0.1 und 127.0.0.2 funktioniert dieser Test selbstverständlich nur von dem Rechner aus, auf dem Ihr XAMPP installiert ist. Das Beispiel sollte nur zeigen, wie das Einrichten von Virtual Hosts funktioniert. Die Einrichtung von öffentlichen Virtual Hosts im Apache verlaufen aber alle nach diesem Prinzip.

Wenn Sie das Beispiel nicht mehr benötigen, löschen Sie es in der `httpd-vhosts.conf` oder kommentieren Sie das `Include` zur `httpd-vhosts.conf` in der `httpd.conf` einfach aus:

```
# Include conf/extra/httpd-vhosts.conf
```

3.2 Verzeichnisschutz mit .htaccess

Eine sekundäre Konfiguration über die Datei `.htaccess` erlaubt Ihnen, Einstellungen während des laufenden Betriebs des Apache und damit ohne den obligatorischen Neustart zu verändern. Über die `.htaccess` ist es somit möglich, weiteren nicht administrativen Benutzern die Möglichkeit einer beschränkten Konfiguration für ihr Dokumentenverzeichnis zu erlauben.

Eine typische Verwendung der `.htaccess`-Datei ist der Passwortschutz bestimmter Dateien oder Verzeichnisse. Hierbei dürfen nur über eine Passwort-Datei authentifizierte Benutzer in das geschützte Verzeichnis gelangen. Um neue Benutzer mit ihren Passwörtern anzulegen, benutzen Sie das Programm `htpasswd`:

- Linux: `/opt/lampp/bin/htpasswd`
- Windows: `<xampp>\apache\bin\htpasswd.exe`

1. Legen Sie ein Verzeichnis namens `test` in dem `htdocs`-Verzeichnis Ihrer XAMPP-Installation an.
2. Legen Sie in dieses Verzeichnis beispielhaft eine `index.html` mit einem beliebigen Inhalt. Greifen Sie danach testweise mit Ihrem Browser auf diese Datei über die URL `http://localhost/test/index.html` zu.
3. Nun erstellen Sie eine neue Passwort-Datei mit einem neuen Benutzer:

Unter Linux:

```
/opt/lampp/bin/htpasswd -c -b /opt/lampp/.htpasswd oswald geheim
```

Unter Windows:

```
C:\Programme\xampp\apache\bin\htpasswd -c -b  
C:\Programme\xampp\htpasswd oswald geheim
```

4. Erstellen Sie eine Datei mit dem Namen `.htaccess` im Verzeichnis `test` und füllen Sie diese mit folgendem Inhalt:

Linux:

```
AuthUserFile /opt/lampp/.htpasswd
AuthName "Protected Area"
AuthType Basic
<Limit GET POST>
require valid-user
</Limit>
```

Windows:

```
AuthUserFile D:/Programme/xampp/.htpasswd
AuthName "Protected Area"
AuthType Basic
<Limit GET POST>
require valid-user
</Limit>
```

Sobald Sie die `.htaccess` geschrieben haben, ist der Passwortschutz für das Verzeichnis, in dem sich die Datei befindet, aktiv.

5. Geben Sie nun im Browser die URL des `test`-Verzeichnisses ein: `http://localhost/test/index.html`. Sollten Sie die URL noch von Schritt 2 geöffnet haben, dann drücken Sie auf die Aktualisieren-Schaltfläche Ihres Browsers.

Ihnen wird nun vom Browser ein Login-Dialog angezeigt. Erst wenn Sie hier den Benutzernamen und das Passwort aus Schritt 4 angeben, können Sie auf das geschützte Verzeichnis zugreifen.

Wenn Sie später einen weiteren Benutzer hinzufügen möchten, dann tun Sie das mit dem folgenden Befehl:

Linux:

```
/opt/lampp/bin/htpasswd -b /opt/lampp/.htpasswd kay geheim2
```

Windows:

```
C:\Programme\xampp\apache\bin\htpasswd -b C:\Programme\xampp\htpasswd kay geheim2
```

Um einen Benutzer zu löschen, verwenden Sie den folgenden Befehl:

Linux:

```
/opt/lampp/bin/htpasswd -D /opt/lampp/.htpasswd kay
```

Windows:

```
C:\Programme\xampp\apache\bin\htpasswd -D C:\Programme\xampp\htpasswd kay
```

3.3 Eine Frage der Sicherheit: SSL

Verschlüsselte Datenübertragung über den *Secure Socket Layer (SSL)* kennen Sie bestimmt bereits als *HyperText Transfer Protocol Secure (HTTPS)*. Zuständig dafür ist im Apache das Modul *mod_ssl*.

Bei XAMPP ist die SSL-Unterstützung standardmäßig aktiviert und mit einem einfachen selbstzertifizierten Serverzertifikat vorkonfiguriert. Rufen Sie einfach mal *https://localhost* in Ihrem Browser auf und greifen Sie so automatisch verschlüsselt auf ihren Webserver zu.

In den folgenden Tabellen erhalten Sie eine erste Übersicht der für SSL unter XAMPP wichtigsten Dateien und Verzeichnisse:

Name	Bedeutung	Ort
openssl	Kommandozeilenprogramm	/opt/lampp/bin/
mod_ssl.so	Apache-SSL-Modul	/opt/lampp/modules/
openssl.cnf	OpenSSL-Konfigurationsdatei (hier als symbolischer Link)	/opt/lampp/etc/
httpd-ssl.conf	Konfigurationsdatei für mod_ssl im Apache	/opt/lampp/etc/extra/
ssl.crt	Verzeichnis für alle Zertifikate (*.crt)	/opt/lampp/etc/
ssl.key	Verzeichnis für die Serverschlüssel (*.key)	/opt/lampp/etc/

Tabelle 3.3: Die wichtigsten Dateien und Verzeichnisse unter Linux

Name	Bedeutung	Ort
openssl	Kommandozeilenprogramm	<xampp>\apache\bin\
mod_ssl.so	Apache-OpenSSL-Modul	<xampp>\apache\modules\
openssl.cnf	OpenSSL-Konfigurationsdatei (hier als symbolischer Link)	<xampp>\apache\bin\
httpd-ssl.conf	Konfigurationsdatei für mod_ssl im Apache	<xampp>\apache\conf\extra
ssl.crt	Verzeichnis für Zertifikate (*.crt)	<xampp>\apache\conf\
ssl.key	Verzeichnis für die Serverschlüssel (*.key)	<xampp>\apache\conf\

Tabelle 3.4: Die wichtigsten Dateien und Verzeichnisse unter Windows

3.3.1 Ein selbstzertifiziertes Serverzertifikat erstellen

Ein selbstzertifiziertes Serverzertifikat dürfte die einfachste Form einer SSL-Konfiguration sein. Mit ihr erreichen Sie, dass die Daten, die zwischen dem Browser und Ihrer Website ausgetauscht werden, verschlüsselt übertragen werden.

Allerdings bekommen Sie bei jedem ersten Zugriff eine Warnung angezeigt, da die Echtheit des Zertifikats nicht automatisch überprüft werden kann. Letzteres würde bei einem CA-zertifizierten Serverzertifikat entfallen. Dazu mehr im nächsten Abschnitt.

1. Rufen Sie eine Root-Shell unter Linux oder die Eingabeaufforderung unter Windows auf und geben Sie die folgenden Befehle ein.

Unter Linux:

```
cd /opt/lampp/etc
```

Unter Windows:

```
cd C:\Programme\XAMPP\apache\conf
set PATH=%PATH%;C:\Programme\XAMPP\apache\bin
```

Mit dem `set PATH`-Befehl ersparen Sie sich das Tippen des langen Pfads vor dem `openssl`-Befehl.

2. Erstellen Sie nun einen Serverschlüssel (*server key*). Dieser Schlüssel besteht intern aus einem privaten und einem öffentlichen Teil und bildet später die Basis für die Verschlüsselung der Daten.

Unter Linux:

```
/opt/lampp/bin/openssl genrsa 1024 > ssl.key/server.key
```

Unter Windows:

```
openssl genrsa 1024 > ssl.key\server.key
```

3. Im nächsten Schritt zertifizieren Sie Ihren Serverschlüssel:

Unter Linux:

```
/opt/lampp/bin/openssl req -new -x509 -key ssl.key/server.key -out  
ssl.crt/server.crt
```

Unter Windows:

```
openssl req -config c:\Programme\xampp\apache\bin\openssl.cnf -new -x509 -key  
ssl.key\server.key -out ssl.crt\server.crt
```

4. Nach Aufruf dieses Kommandos werden Ihnen einige Fragen gestellt (die Eingaben wurden fett hervorgehoben):

```
You are about to be asked to enter information that will be incorporated into  
your certificate request.
```

```
What you are about to enter is what is called a Distinguished Name or a DN.
```

```
There are quite a few fields but you can leave some blank
```

```
For some fields there will be a default value,
```

```
If you enter '.', the field will be left blank.
```

```
-----
Country Name (2 letter code) [AU]:DE
State or Province Name (full name) [Some-State]:.
Locality Name (eg, city) []:Berlin
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Apache Friends
Organizational Unit Name (eg, section) []:.
Common Name (eg, YOUR name) []:www.apachefriends.org
Email Address []:oswald@apachefriends.org
```

Wenn Sie ein Feld freilassen wollen, dann geben Sie einfach nur einen Punkt ein. Wichtig ist der leider sehr missverständliche Begriff »Common Name«: Hier geben Sie die Adresse Ihrer Website ein.

```
C:\WINDOWS\system32\cmd.exe

C:\>cd C:\Programme\XAMPP\apache\conf

C:\Programme\xampp\apache\conf>set PATH=%PATH%;C:\Programme\XAMPP\apache\bin

C:\Programme\xampp\apache\conf>openssl genrsa 1024 > ssl.key\server.key
Loading 'screen' into random state - done
Generating RSA private key, 1024 bit long modulus
.....+++++
..+++++
e is 65537 (0x10001)

C:\Programme\xampp\apache\conf>openssl req -config c:\Programme\xampp\apache\bin\
openssl.cnf -new -x509 -key ssl.key\server.key -out ssl.crt\server.crt
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value.
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:DE
State or Province Name (full name) [Some-State]:.
Locality Name (eg, city) []:Berlin
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Apache Friends
Organizational Unit Name (eg, section) []:.
Common Name (eg, YOUR name) []:www.apachefriends.org
Email Address []:oswald@apachefriends.org

C:\Programme\xampp\apache\conf>
```

Abbildung 3.1: Die Erstellung eines selbstzertifizierten Zertifikats samt Erstellung eines neuen Serverschlüssels passt locker auf eine Bildschirmseite (Screenshot von Windows XP).

5. Starten Sie nun den Apache neu, damit er seinen neuen Schlüssel und sein neues Zertifikat lädt.
6. Zum Testen Ihres neuen Zertifikats greifen Sie mit <https://localhost> auf Ihren Webserver zu.



Abbildung 3.2: Der Nachteil eines selbstzertifizierten Zertifikats: Die Echtheit kann vom Browser nicht automatisch festgestellt werden.

3.3.2 Ein CA-zertifiziertes Serverzertifikat erstellen

In der Regel kommt für den Produktiveinsatz ein kostenpflichtiges Zertifikat einer offiziellen Zertifizierungsstelle (*Certification Authority*, CA) zum Einsatz. Für dieses Beispiel beziehen wir ein Zertifikat der Klasse 0 (kostenfreies Testzertifikat) von dem Anbieter *TrustCenter* (<http://www.trustcenter.de>).

1. Rufen Sie eine Root-Shell unter Linux oder die Eingabeaufforderung unter Windows auf und geben Sie die folgenden Befehle ein.

Unter Linux:

```
cd /opt/lampp/etc
```

Unter Windows:

```
cd C:\Programme\XAMPP\apache\conf
set PATH=%PATH%;C:\Programme\XAMPP\apache\bin
```

Mit dem `set PATH`-Befehl ersparen Sie sich das Tippen des langen Pfads vor dem `openssl`-Befehl.

2. Erstellen Sie nun einen Serverschlüssel (*server key*). Dieser Schlüssel besteht intern aus einem privaten und einem öffentlichen Teil und bildet später die Basis für die Verschlüsselung der Daten.

Unter Linux:

```
/opt/lampp/bin/openssl genrsa 1024 > ssl.key/server.key
```

Unter Windows:

```
openssl genrsa 1024 > ssl.key\server.key
```

3. Im nächsten Schritt erstellen Sie Ihren *Certificate Signing Request (CSR)*:

Unter Linux:

```
/opt/lampp/bin/openssl req -new -key ssl.key/server.key -out server.csr
```

Unter Windows:

```
openssl req -config c:\Programme\xampp\apache\bin\openssl.cnf -new -key ↵
ssl.key\server.key -out server.csr
```

4. Nach Aufruf dieses Kommandos werden Ihnen einige Fragen gestellt (die Eingaben wurden fett hervorgehoben). Wenn Sie ein Feld freilassen wollen, dann geben Sie einfach nur einen Punkt ein. Wichtig ist der leider sehr missverständliche Begriff »Common Name«: Hier geben Sie die Adresse Ihrer Website ein.

You are about to be asked to enter information that will be incorporated into ↵
your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value,

If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:**DE**

State or Province Name (full name) [Some-State]:.

Locality Name (eg, city) []:**Berlin**

Organization Name (eg, company) [Internet Widgits Pty Ltd]:**Apache Friends**

Organizational Unit Name (eg, section) []:.

Common Name (eg, YOUR name) []:**www.apachefriends.org**

Email Address []:**oswald@apachefriends.org**

5. Nun werden Sie noch nach einem *challenge password* und einem *optional company name* gefragt. Geben Sie in beiden Fällen nichts ein und drücken Sie die -Taste:

Please enter the following 'extra' attributes
to be sent with your certificate request

A challenge password []:

An optional company name []:

Durch den letzten Befehl wurde die Datei `server.csr` im aktuellen Verzeichnis erstellt.

6. Öffnen Sie die Datei mit einem Texteditor Ihrer Wahl. Sie sollten etwa folgenden Inhalt sehen:

```
-----BEGIN CERTIFICATE REQUEST-----
MIIBwTCCASoCAQAwYwYxZAJBgNVBAYTAkRFRMQ8wDQYDVQQHEwZCZXJsaW4xZzAV
BgNVBAoTdKfWYWN0ZSBGcm1lbnRzMR4wHAYDVQQDEwV3d3cuYXBhY2h1ZnJpZW5k
cy5vcmxJzA1BgkqhkiG9w0BCQEWGG9zd2FsZEBhcGFjaGVmcm1lbnRzLm9yZzCB
nzANBgkqhkiG9w0BAQEFAAOBjQAwYkCgYEAxTYX60MgDjBt4szR+2poolTKAzBL
/BWAUKS7UvHRPNTGp9nJJwnK1WmkmewhmArz1n032UKwc4ZrWwsiVPK4rZOCV1
GvyNcQp8UzrnEOHLSAQ0j52b1zvQM/m0IJaCK/TsCy6hvurr2ki9cIQ0hpVW1xoc
cuP7qNY3b9ubGGkCAwEAaAAMAOGCSqGSIb3DQEBBQUAA4GBAIGZmPRxfMnVSC5a
J/H1QnxY2ZAJcRV9bJvqIwBVR10u9gz1IC1w1e9I0nToCekcPRhbbg+1vybcZeXI
5qv/PVnMZFLnLa81SOLYv7xNiAnObr7q+LG0zzpareg04dGEg3wBLVBZJD9DeeFs
ytcEVnWIsU2BBwV713tACHa/6eoT
-----END CERTIFICATE REQUEST-----
```

7. Übermitteln Sie diesen Inhalt an Ihre Zertifizierungsstelle. In der Regel hat diese ein Webformular, über das Sie die CSR mit Copy&Paste eingeben können. Die Zertifizierungsstelle wird Ihre Daten überprüfen und bei positiver Prüfung erhalten Sie per E-Mail oder per Web Ihr tatsächliches Serverzertifikat.

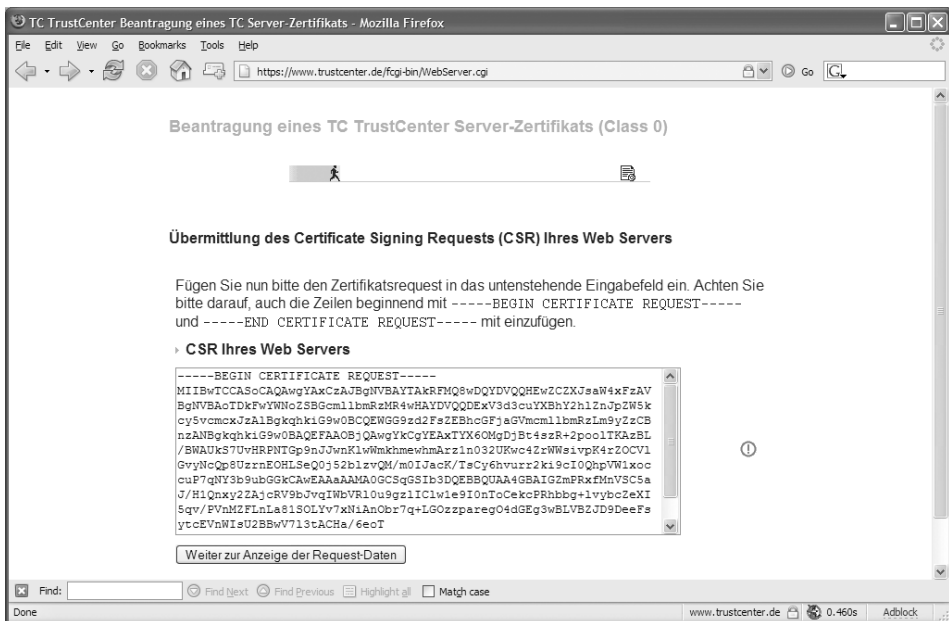


Abbildung 3.3: Übermitteln Sie Ihr CSR an Ihre Zertifizierungsstelle. Von ihr erhalten Sie dann das Serverzertifikat.

8. Wenn Sie das Serverzertifikat der CA erhalten haben, tragen Sie es in die Datei `server.crt` ein. Starten Sie Ihren Apache neu und greifen Sie danach über HTTPS auf Ihren Webserver zu.

Wenn alles richtig gelaufen ist, dann kann der Browser nun automatisch die Echtheit des Zertifikats überprüfen und zeigt Ihnen keine Warnung wie bei dem selbst-zertifizierten Zertifikat an.



Achtung

Wenn Sie nur ein Testzertifikat einer CA verwenden, dann wird auch in diesem Fall eine Warnung angezeigt. Die Warnung verschwindet wirklich erst dann, wenn Sie ein richtiges Zertifikat erworben haben.

3.4 WebDAV

Das *WebDAV*-Protokoll (*DAV* = *Distributed Authoring and Versioning*) ermöglicht Benutzern, über passwortgeschützte Verzeichnisse Dokumente direkt am Zielserver zu verändern. In der Praxis ist WebDAV mit den Programmen *Dreamweaver* und *Adobe GoLive* zu einer gewissen Bekanntheit gekommen. Aber auch *Netzwerklaufwerke* können über WebDAV freigegeben werden.

Zwei Authentifikationsmethoden sind im Apache HTTPD erlaubt:

3.4.1 Digest

Hierbei werden die Passwörter über einen speziellen Algorithmus mit dem Programm *htdigest* generiert. Beispiele dazu:

Apache-Direktive: `AuthType Digest`

Nachteil: `AuthType Digest` wird von *Dreamweaver MX* als Client **nicht unterstützt!**

3.4.2 Basic

Ein bekannter Algorithmus, der auch schon bei dem Beispiel des Verzeichnisschutzes mit einer `.htaccess` Verwendung fand.

Apache-Direktive: `AuthType Basic`

3.4.3 XAMPP-Spezifika

In XAMPP ist die WebDAV-Unterstützung bereits standardmäßig aktiviert. Sie müssen selbst nur noch eine Passwort-Datei über das Apache-Programm *htpasswd* erstellen. In diesen XAMPP-Versionen ist `AuthType Basic` die Standardmethode der Authentifizierung. Die WebDAV-Verzeichnisse im XAMPP lauten unter

Linux: `/opt/lampp/webdav`

Windows: `<xampp>\webdav`

Um in den älteren XAMPP-Versionen die WebDAV-Unterstützung des Apache in der `httpd.conf` zu aktivieren, kommentieren Sie die folgenden Zeilen aus:

```
LoadModule dav_module modules/mod_dav.so
LoadModule dav_fs_module modules/mod_dav_fs.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule auth_digest_module modules/mod_auth_digest.so
LoadModule setenvif_module modules/mod_setenvif.so
```

Überprüfen Sie ebenfalls in der `httpd.conf`, ob die `httpd-dav.conf` hier bereits geladen ist.

```
Include conf/extra/httpd-dav.conf
```

Alles Weitere passiert nun in der `httpd-dav.conf`:

Linux: `/opt/lampp/etc/extra/httpd-dav.conf`

Windows: `<xampp>\apache\conf\extra\httpd-dav.conf`

3.4.4 Die Beispielkonfiguration im XAMPP für Linux

Bei älteren XAMPP-Versionen müssen Sie vielleicht das Beispiel selbst erstellen. Hierfür müssen Sie zu Beginn als `su` noch zwei wichtige Schritte vollziehen:

1. Erstellen Sie ein WebDAV-Verzeichnis und setzen Sie Benutzerrechte.

```
mkdir /opt/lampp/webdav
chown -R nobody.root /opt/lampp/webdav
```

2. Erstellen Sie eine Passwort-Datei mit `htpasswd`.

```
/opt/lampp/bin/htpasswd -c -b /opt/lampp/htpasswd.webdav username secret
```

3. Passen Sie die `httpd-dav.conf` an.

Ihre Konfiguration der `httpd-dav.conf` könnte wie folgt aussehen:

```
<IfModule dav_module>
<IfModule dav_fs_module>
<IfModule setenvif_module>
<IfModule authn_file_module>
```

```
DavLockDB "/tmp/DavLock"
Alias /webdav "/opt/lampp/webdav"

<Directory "/opt/lampp/webdav">
    Dav On
    Order allow,deny
    Allow from all
    AuthName "XAMPP with WebDAV"
    AuthType Basic
    AuthUserFile "/opt/lampp/htpasswd.webdav"

    <LimitExcept GET HEAD OPTIONS>
        require valid-user
    </LimitExcept>
</Directory>
BrowserMatch "Microsoft Data Access Internet Publishing Provider" ↵
    redirect-carefully
BrowserMatch "MS FrontPage" redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
BrowserMatch "WebDAVFS/1.[0123]" redirect-carefully
BrowserMatch "^gnome-vfs/1.0" redirect-carefully
BrowserMatch "^XML Spy" redirect-carefully
BrowserMatch "Dreamweaver-WebDAV-SCM1" redirect-carefully
BrowserMatch "MSIE" AuthDigestEnableQueryStringHack=On
</IfModule>
</IfModule>
</IfModule>
</IfModule>
```

4. Starten Sie den Apache neu.

Ihre WebDAV-Konfiguration in der Übersicht:

Verzeichnis	/opt/lampp/webdav
URL	http://localhost/webdav
Benutzer	[von Ihnen gewählt]
Passwort	[von Ihnen gewählt]
Passwort-Datei (mit <i>htpasswd</i>)	/opt/lampp/htpasswd.webdav

3.4.5 Die Beispielkonfiguration im XAMPP für Windows

Der XAMPP liegt in diesem Beispiel erneut unter C:\Programme vor. Für alle älteren XAMPP-Versionen haben Sie, wie eben unter Linux beschrieben, alle notwendigen Apache-Module geladen und die `httpd-dav.conf` in der `httpd.conf` aktiviert.

1. Erstellen Sie eine Passwort-Datei mit *htpasswd*.

```
C:\Programme\xampp\apache\bin\htpasswd -c ↵
C:\Programme\xampp\security\htpasswd.webdav username
```

2. Passen Sie die *httpd-dav.conf* an.

```
<IfModule dav_module>
<IfModule dav_fs_module>
<IfModule setenvif_module>
<IfModule authn_file_module>
DavLockDB "D:/Programme/xampp/tmp/DavLock"
Alias /webdav "D:/Programme/xampp/webdav"
<Directory "D:/Programme/xampp/webdav">
    Dav On
    Order allow,deny
    Allow from all
    AuthName "XAMPP with WebDAV"
    AuthType Basic
    AuthUserFile "D:/Programme/xampp/security/htpasswd.webdav"
    <LimitExcept GET HEAD OPTIONS>
        require valid-user
    </LimitExcept>
</Directory>
BrowserMatch "Microsoft Data Access Internet Publishing Provider" redirect-
carefully
BrowserMatch "MS FrontPage" redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
BrowserMatch "WebDAVFS/1.[0123]" redirect-carefully
BrowserMatch "gnome-vfs/1.0" redirect-carefully
BrowserMatch "XML Spy" redirect-carefully
BrowserMatch "^Dreamweaver-WebDAV-SCM1" redirect-carefully
BrowserMatch "MSIE" AuthDigestEnableQueryStringHack=On
</IfModule>
</IfModule>
</IfModule>
</IfModule>
```

3. Starten Sie den Apache neu.

Ihre WebDAV-Konfiguration in der Übersicht:

Verzeichnis	<xampp>\webdav
URL	http://localhost/webdav
Benutzer	[von Ihnen gewählt]
Passwort	[von Ihnen gewählt]
Passwort-Datei (mit htpasswd)	<xampp>\security\htpasswd.webdav

3.4.6 AuthType Digest

An dieser Stelle soll noch eine Beispielkonfiguration mit `htdigest` vorgestellt werden.

1. Erstellen Sie eine Passwort-Datei.

Linux:

```
/opt/lampp/bin/htdigest -c "/opt/lampp/httpdigest.webdav" "XAMPP with WebDAV" 
username
```

Windows:

```
C:\Programme\xampp\apache\bin\htdigest -c 
"C:\Programme\xampp\security\httpdigest.webdav" "XAMPP with WebDAV" username
```

2. Editieren Sie die `httpd-dav.conf`.

Linux:

```
<IfModule dav_module>
<IfModule dav_fs_module>
<IfModule auth_digest_module>
<IfModule setenvif_module>
<IfModule authn_file_module>
DavLockDB "/tmp/DavLock"
Alias /webdav "/opt/lampp/webdav"
<Directory "/opt/lampp/webdav">
    Dav On
    Order allow,deny
    Allow from all
    AuthName "XAMPP with WebDAV"
    AuthType Digest
    AuthDigestDomain / http://localhost/
    AuthUserFile "/opt/lampp/httpdigest.webdav"
    <LimitExcept GET HEAD OPTIONS>
        require valid-user
    </LimitExcept>
</Directory>
BrowserMatch "Microsoft Data Access Internet Publishing Provider" 
redirect-carefully
BrowserMatch "MS FrontPage" redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
BrowserMatch "WebDAVFS/1.[0123]" redirect-carefully
BrowserMatch "^gnome-vfs/1.0" redirect-carefully
BrowserMatch "XML Spy" redirect-carefully
BrowserMatch "Dreamweaver-WebDAV-SCM1" redirect-carefully
BrowserMatch "MSIE" AuthDigestEnableQueryStringHack=On
</IfModule>
</IfModule>
</IfModule>
</IfModule>
</IfModule>
```


Windows:

```

<IfModule dav_module>
<IfModule dav_fs_module>
<IfModule auth_digest_module>
<IfModule setenvif_module>
<IfModule authn_file_module>
DavLockDB "D:/Programme/xampp/tmp/DavLock"
Alias /webdav "D:/Programme/xampp/webdav"
<Directory "D:/Programme/xampp/webdav">
    Dav On
    Order allow,deny
    Allow from all
    AuthName "XAMPP with WebDAV"
    AuthType Digest
    AuthDigestDomain / http://localhost/
    AuthUserFile "D:/Programme/xampp/security/htdigest.webdav"
    <LimitExcept GET HEAD OPTIONS>
        require valid-user
    </LimitExcept>
</Directory>
BrowserMatch "Microsoft Data Access Internet Publishing Provider" ↩
    redirect-carefully
BrowserMatch "MS FrontPage" redirect-carefully
BrowserMatch "^WebDrive" redirect-carefully
BrowserMatch "^WebDAVFS/1.[0123]" redirect-carefully
BrowserMatch "gnome-vfs/1.0" redirect-carefully
BrowserMatch "XML Spy" redirect-carefully
BrowserMatch "^Dreamweaver-WebDAV-SCM1" redirect-carefully
BrowserMatch "MSIE" AuthDigestEnableQueryStringHack=On
</IfModule>
</IfModule>
</IfModule>
</IfModule>
</IfModule>

```

3. Starten Sie den Apache neu.

3.4.7 Anwendungsmöglichkeiten mit WebDAV

Dreamweaver MX

Bitte beachten Sie: *Dreamweaver MX* kann **nicht** mit einer Digest-Authentifizierung umgehen. Also müssen Sie in diesem Fall mit *AuthType Basic* und *htpasswd* arbeiten. Dann sollte auch Dreamweaver MX problemlos funktionieren.

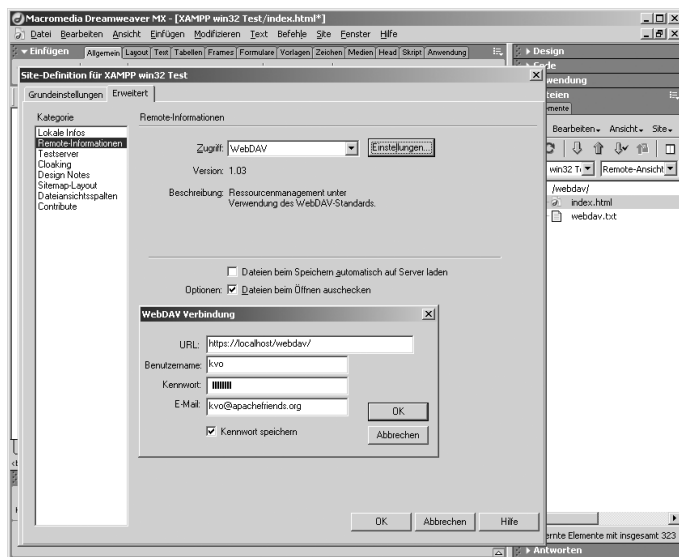


Abbildung 3.4: WebDAV-Konfiguration mit Dreamweaver MX

GoLive CS2

Adobe GoLive CS2 konnte in unseren Versuchen sowohl mit der Authentifizierung über Digest als auch mit Basic problemfrei umgehen. Der Kontakt mit WebDAV erfolgt hier über das Menü unter DATEI > SERVER > MIT WEBDAV VERBINDEN. Sie können wie auch beim *Dreamweaver MX* mehrere Server oder Verzeichnisse angeben.

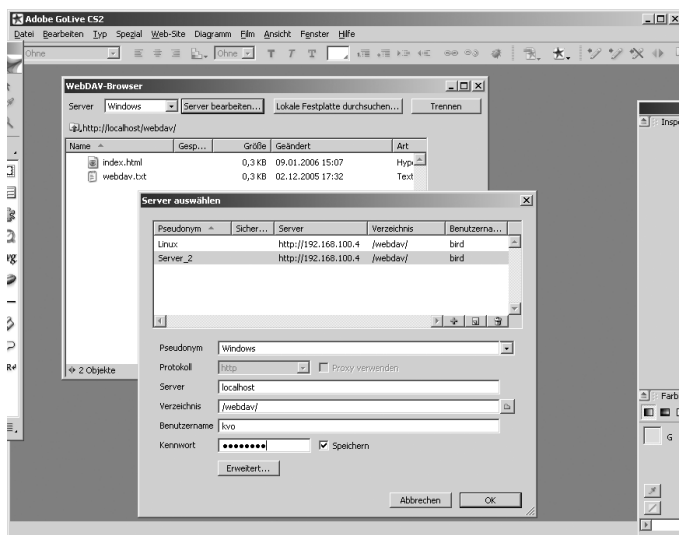


Abbildung 3.5: Adobe GoLive CS2 mit WebDAV

WebDAV als Netzwerkordner unter Windows XP

Gehen Sie in die NETZWERKUMGEBUNG, um sich über NETZWERKRESSOURCE VERBINDEN mit dem WebDAV-Netzwerkordner zu verbinden.

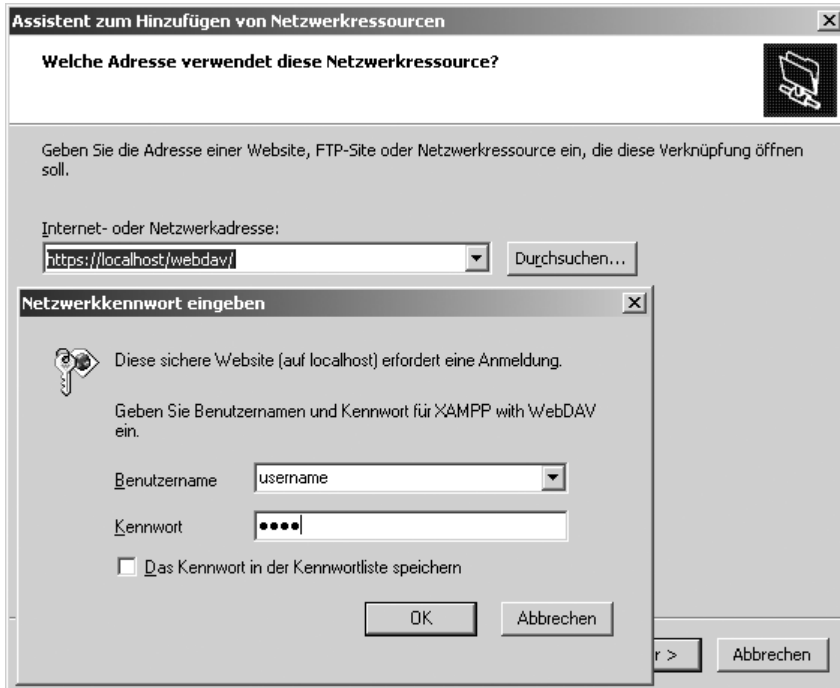


Abbildung 3.6: Anmelden am WebDAV-Ordner

Alles erfolgreich eingerichtet? Dann testen Sie Ihren neuen Webordner im Bereich NETZWERKUMGEBUNG.

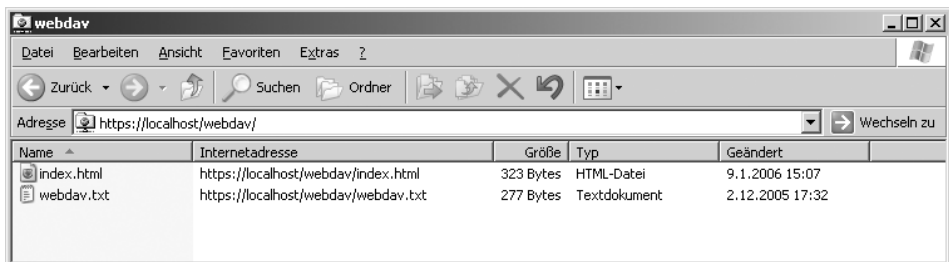


Abbildung 3.7: Der neue Netzwerkordner



4 PHP

In diesem Kapitel widmen wir uns verschiedenen Aspekten von PHP, unter anderem dem dynamischen Generieren von PDFs und den Schnittstellen, die PHP für die Arbeit mit XML-Daten bietet. Zunächst aber folgt ein Blick auf das Thema *Entwicklungsumgebungen*.

4.1 Entwicklungsumgebungen

4.1.1 Zend Studio 5

Zend Studio 5 der Firma *Zend Technologies Ltd.* (<http://www.zend.com>) ist wohl eine der bekanntesten PHP-Entwicklungsumgebungen. Immerhin ist hier die gesamte PHP-Entwicklung ansässig, seitdem die PHP3-Gründer Andi Gutmans und Zeev Suraski das Unternehmen aufgebaut haben. Das ist auch ein Grund für XAMPP, sich den Anforderungen des kommerziellen Zend Studios zu stellen.

Neben den typischen Elementen eines umfangreichen PHP-Editors besticht die Version 5 durch folgende Highlights:

- Debug-Sitzungen intern oder wahlweise über Server
- Code-Optimizer
- Encoder zum Schutz eigener Quellcodes
- Unterstützung für FTP auch mit SSL
- Unterstützung von CVS und Subversion
- PHP-Benchmark und Profiler
- WSDL-Generator usw.

Tipp



Bitte überprüfen Sie zu Beginn, ob die PHP-Version im XAMPP schon einen entsprechenden Zend Optimizer und Debugger anbietet. XAMPP ist immer sehr aktuell und enthält in der Regel die zuletzt veröffentlichte Zend Engine inklusive des Optimizers. Trotzdem kann es schon einmal vorkommen, dass zwar ein neues PHP-Release veröffentlicht und im XAMPP enthalten ist, es aber noch nicht vom Zend Studio Server unterstützt wird. Zudem führt in einem solchen Fall die Installation des Servers zu einer kompletten Deaktivierung der PHP-Funktionalität im XAMPP. Das Problem tritt nur mit dem Zend Studio Server auf, nicht mit dem Studio Client. Tipp: Testen Sie ggf. eine ältere XAMPP-Version als Plattform beispielsweise mit einer Evaluation-Version des Zend Studios und am besten auf einem produktionsunabhängigen Rechner.

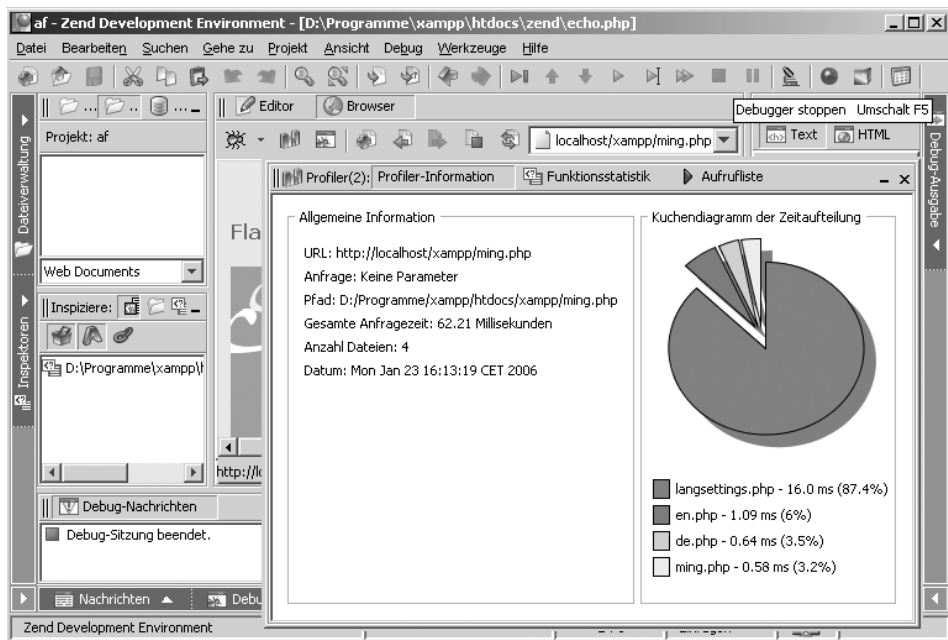


Abbildung 4.1: Zend Studio Client mit einer bestechenden Analyzer-Funktion

Es gibt drei Varianten des Zend Studios mit unterschiedlicher Ausstattung. Innerhalb dieser Varianten variieren die Preise zudem mit der Dauer der ausgestellten Lizenzen.

Name	Features
Zend Studio™ – Standard	Editor mit internem Debugger
Zend Studio™ – Professional	Editor, Debugger, Analyzer, Optimizer
Zend Studio™ – Enterprise	Zend Studio Professional und QA/TestPlattform

Tabelle 4.1: Funktionsumfang der verschiedenen Zend Studio-Versionen

Sowohl die Professional- als auch die Enterprise-Version teilen sich in einen Client, der die eigentliche Entwicklungsumgebung enthält, und den Zend Server. Der Zend Server besteht hauptsächlich aus dem Zend Debug Server, dem Zend Server Center sowie dem Zend Optimizer. Außerdem besitzt der Zend Server den Apache und natürlich auch PHP, sollte beispielsweise XAMPP nicht auf dem System vorliegen. In der Enterprise-Version bietet Zend zusätzlich die sog. »Zend Platform« an, die sich mit einer verbesserten PHP-Performance und optimierten Analysefunktionen sowie weiteren Synchronisationsoptionen auch über mehrere Server weitestgehend an speziellen Bedürfnissen in Unternehmen orientiert.

4.1.2 Installation für XAMPP unter Windows adaptieren

Für dieses Beispiel besorgten wir uns die Evaluation-Pakete `ZendStudio-5_1_0a.exe` (rund 60 Mbyte) und `ZendStudioServer-5.1.0a-Windows-i386.exe` (rund 20 Mbyte) über www.zend.com. Der hierbei benutzte XAMPP lag in der Version 1.5.0 vor und enthielt PHP in der Version 5.0.5.

Installation des Zend Studio Clients

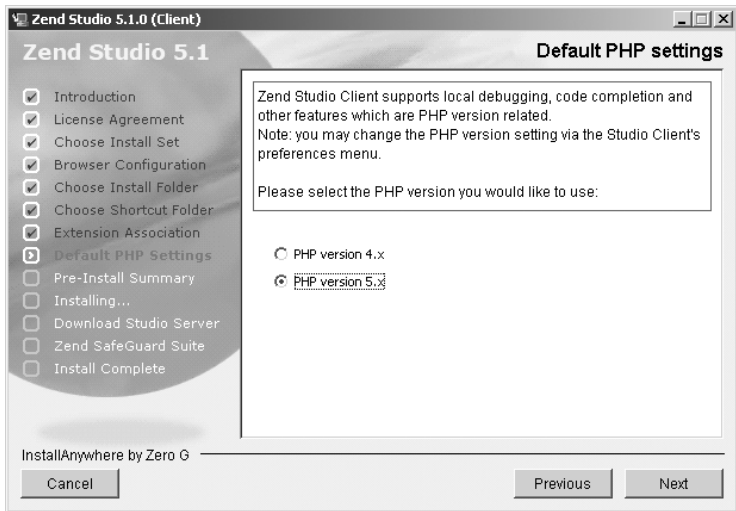


Abbildung 4.2: Auswahl der PHP5-Unterstützung im Setup des Zend Studios 5.1

Die Installation des Zend Studio Clients verlief unkompliziert. Als interner Debugger enthält der Client Minipakete von PHP4 und PHP 5 inklusive einer `ZendDebuggerLocal.dll`. Damit läuft der Client so weit unabhängig von einer PHP-Installation oder eben einer XAMPP-Plattform.

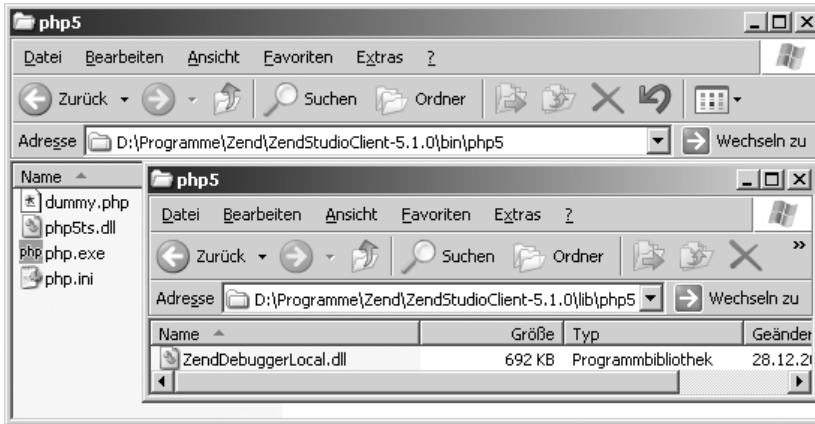


Abbildung 4.3: Der Zend Client enthält einen eigenen PHP4- und PHP5-Debugger

Installation des Zend Studio Servers

1. Richten Sie den Apache im XAMPP als Dienst ein. Sie können zum Installieren des Dienstes die hierfür vorgesehene Batch-Datei benutzen:

```
<xampp>\apache\apache_installservice.bat
```

2. Im nächsten Schritt definieren Sie Ihre Apache-Version (Apache 2) sowie Ihre PHP-Version (meistens PHP 5).
3. Danach erfragt das Setup den absoluten Pfad zu Ihrer `php.ini`. Bitte geben Sie hierfür unbedingt die Datei `<xampp>\apache\bin\php.ini` an.
4. Danach definieren Sie im Zend Server-Setup den Pfad zur `httpd.conf`.
5. Es folgt die Definition des Pfads zum `htdocs`-Verzeichnis.
6. Als Nächstes legen Sie die Hosts und ein Passwort für den Zugriff auf die Zend Server-Konsole fest.
7. Schließen Sie die Installation mit einem Neustart ab.

Sie erreichen nun die webbasierte Konfiguration des Zend Studio Servers über Ihren lokalen Rechner mit der URL <http://localhost/ZendStudioServer/>.

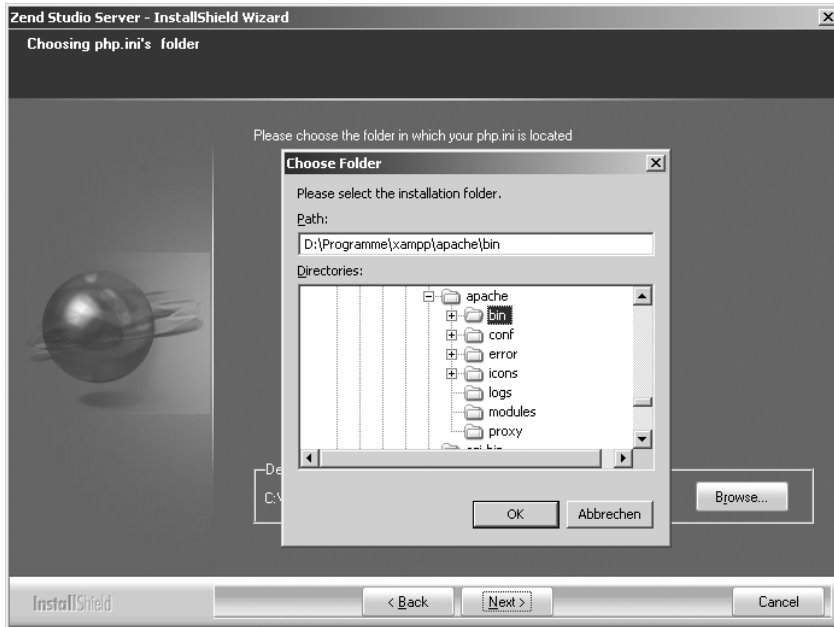


Abbildung 4.4: Wählen Sie über das Setup die `php.ini` im Ordner `<xampp>\apache\bin` aus.

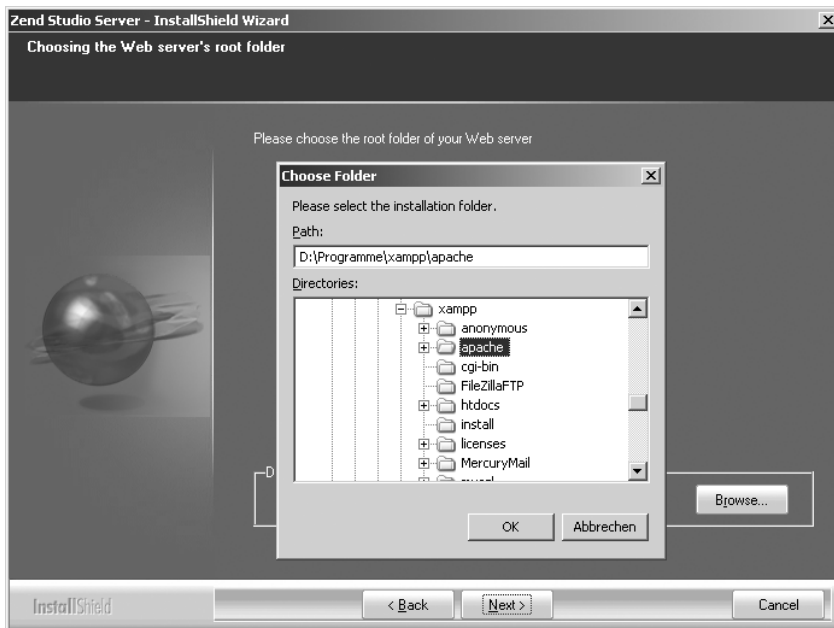


Abbildung 4.5: Der Apache im XAMPP befindet sich im Ordner `xampp\apache`.

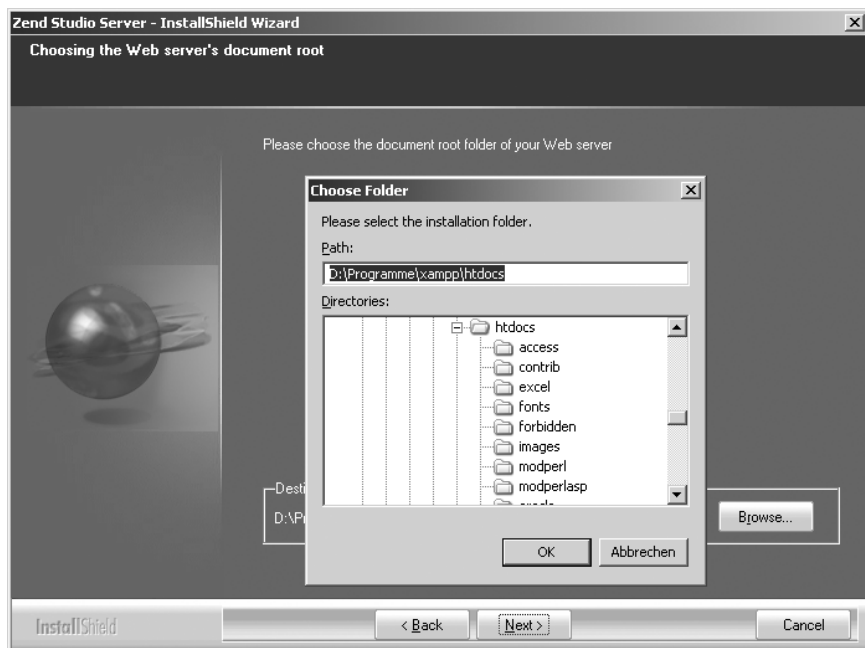


Abbildung 4.6: Der korrekte Pfad lautet xampp\htdocs.

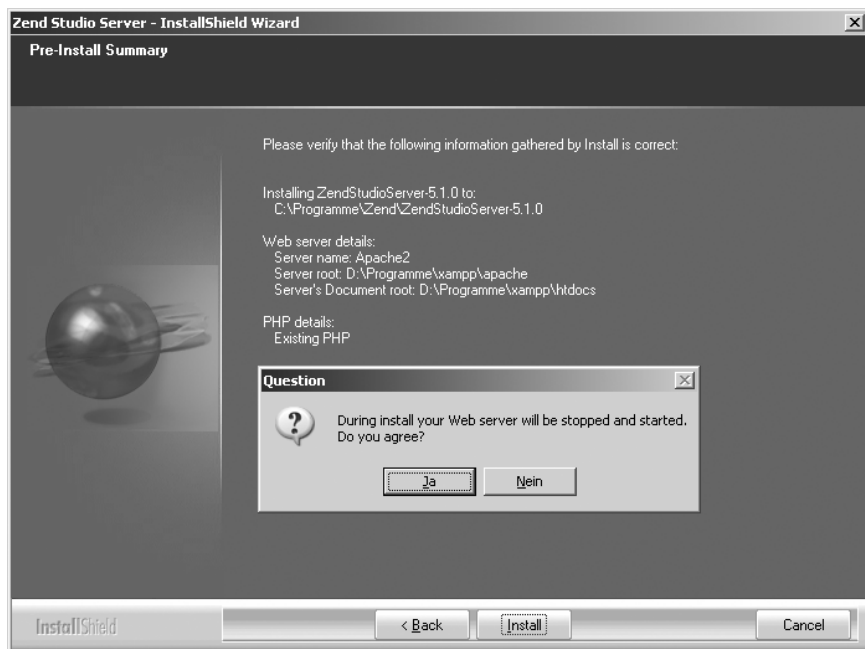


Abbildung 4.7: Der Abschluss der Installation endet mit einem Neustart des Apache 2.

Lösung möglicher Probleme mit dem Zend Optimizer

Gelegentlich kommt beim Aufruf des Studio Servers die Meldung *Zend Optimizer not installed*. Das bedeutet generell, dass PHP nicht das Optimizer-Modul geladen hat oder laden konnte. Manchmal reicht hier schon eine Aktivierung des Moduls im [Zend]-Abschnitt der `php.ini` aus:

```
zend_optimizer.enable_loader = 1
```

Dadurch, dass XAMPP in der Regel bereits einen Optimizer enthält, kann es auch hier zu Problemen mit der Kompatibilität kommen. Als Lösung können Sie in diesem Fall die entsprechenden Module in der Zend Studio Server-Installation verwenden. In dem folgenden Beispiel lag die Installation des Studio Servers im Verzeichnis `C:\ZendStudioServer-5.1.0` vor. Die `php.ini` könnte in diesem Fall wie folgt modifiziert werden.

Konfigurationsdatei: `<xampp>\apache\bin\php.ini`

```
[Zend]
zend_extension_ts="C:\ZendStudioServer-5.1.0\lib\ZendExtensionManager.dll"
zend_extension_manager.optimizer_ts="C:\ZendStudioServer-5.1.0\lib\Optimizer-2.6.1"
zend_optimizer.enable_loader = 1
zend_optimizer.optimization_level=15
;zend_optimizer.license_path =
; Local Variables:
; tab-width: 4
; End:
zend_extension_manager.debug_server_ts="C:\ZendStudioServer-5.1.0\lib\Debugger-5.1.0"
studio.install_dir="C:\ZendStudioServer-5.1.0"
zend_debugger.expose_remotely=allowed_hosts
zend_gui.ini_modifier="C:\ZendStudioServer-5.1.0\sbin\ini_modifier.exe"
zend_debugger.deny_hosts=
zend_gui_password=4d27c0881c2c9e4f57f20160abcef268
zend_debugger.allow_hosts=127.0.0.1/32
zend_ini_file="C:\ZendStudioServer-5.1.0\etc\zend.ini"
```

Es kann aber auch sein, dass aufgrund des zumeist sehr aktuellen PHP-Release im XAMPP gerade der dort zur Verfügung stehende *Zend Optimizer* aktueller und somit kompatibler ist. Bei möglichen Problemen heißt das für Sie: Probieren geht (manchmal) über studieren. Die Zend-Entwickler sind gelegentlich langsamer im Hinblick auf die Aktualisierung des *Zend Studios* im Vergleich zu ihren zahlreichen PHP-Neuerscheinungen. Wenn also alles nicht funktionieren will, kann schon die Verwendung einer älteren XAMPP-Version ausreichen:

http://sourceforge.net/project/showfiles.php?group_id=61776

Wiederherstellen der XAMPP-Konfiguration unter Windows

Um die XAMPP-Konfiguration unter Windows wiederherzustellen, reicht die Etablierung der Sicherheitskopien der `php.ini` und der `httpd.conf` vom Studio Server.

Verzeichnis: `<xampp>\apache\bin`

```
cp php.ini zendserver_php.ini
cp php.ini.ZendStudioServer-5.1.0_bakphp.ini
```

Verzeichnis: `<xampp>\apache\conf`

```
cp httpd.conf zendserver_httpd.conf
cp httpd.conf.ZendStudioServer-5.1.0_bak httpd.conf
```

Achtung



Der Zend Server macht in der Registry einen Schlüssel mit dem Namen `ath` auf. Leider hatte das auf unserem System zur Folge, dass beim Starten eines alternativen XAMPP-Serverpakets immer jene Module geladen wurden, die durch die `php.ini` im Schlüssel `IniFilePath` definiert waren.



Abbildung 4.8: Typische Nachricht des Apache beim fehlerhaften Laden der Module

Ein Beispiel in der Registry:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\PHP]
"IniFilePath"="D:\\Programme\\xampp\\apache\\bin\\"
```

Als auf dem System XAMPP mit einer aktuelleren PHP-Version, nämlich 5.1.2, gestartet werden sollte, benutzte unser Server die älteren PHP-Erweiterungen, wie sie (in diesem Fall) in der `D:\\Programme\\xampp\\apache\\bin\\php.ini` als Standard übergeben werden. Das bedeutet, dass der Registry-Wert `IniFilePath` jederzeit übergeordnet die aktive `php.ini` bestimmt. Und das sogar, wenn zeitweise ein anderer XAMPP zum Einsatz kommt.

Der Einsatz mehrerer XAMPP-Versionen auf einem Rechner ist in der Regel ohne Probleme möglich. Einzige Einschränkung: Die spezifischen Server dürfen hier eben nicht parallel laufen. Doch das wurde leider mit dem Wert `IniFilePath` obsolet, weshalb wir uns entschlossen, diesen Wert kompromisslos zu löschen. Der Zend Server lief übrigens auch ohne diesen Registry-Eintrag über das zuvor bestimmte XAMPP-Bündel. Wir müssen hier jedoch fairerweise anmerken, dass wir nach dem Löschvorgang nicht alle Eventualitäten getestet haben.

Den Registry-Wert `IniFilePath` finden Sie auch in der PHP-Installer-Version des Zend-Unternehmens. Dieser Wert ist damit oft Ursache für Fehlfunktionen bei gemeinsamer Nutzung von XAMPP und PHP-Installer auf einem System.

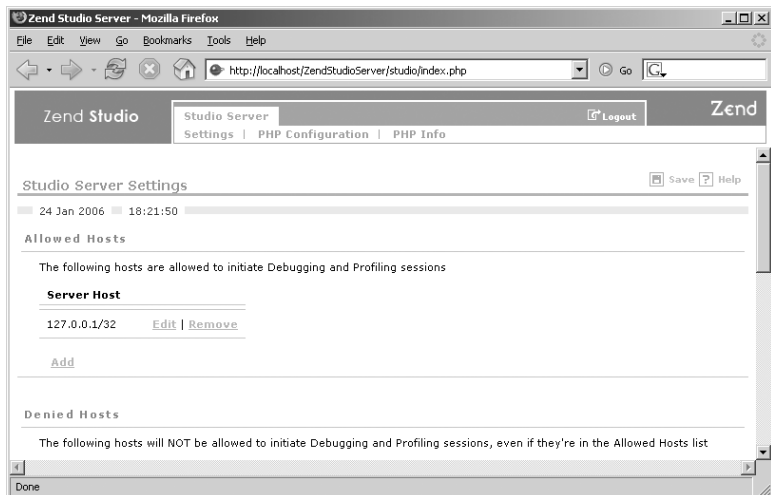


Abbildung 4.9: Der Studio Server lief auch ohne den Registry-Wert `IniFilePath`

4.1.3 Installation für XAMPP unter Linux adaptieren

Unter Linux wurden auf der Grundlage der *XAMPP 1.5.0*-Version die Pakete *ZendStudio-5_1_0.tar.gz* sowie *ZendStudioServer-5.1.0a-linux-glibc21-i386.tar.gz* aufgesetzt. Als Distribution stand uns *SUSE 9.2* mit dem *Kernel 2.6.8* zur Verfügung.

Installation des Zend Studio Clients

Für die Installation des Zend Studio Clients benötigen Sie *root*-Rechte.

1. Entpacken Sie das *ZendStudio*-Paket mit folgendem Befehl:

```
tar xvfz ZendStudio-5_1_0.tar.gz
```

2. Führen Sie das Binärpaket mit *root*-Rechten vor Ort aus:

```
./ZendStudio-5_1_0.bin
```

3. Führen Sie das Setup aus. Sie können die vom Setup vorgeschlagenen Standards übernehmen. Bei der Standard-PHP-Einstellung entschieden wir uns für die Version 5.x.

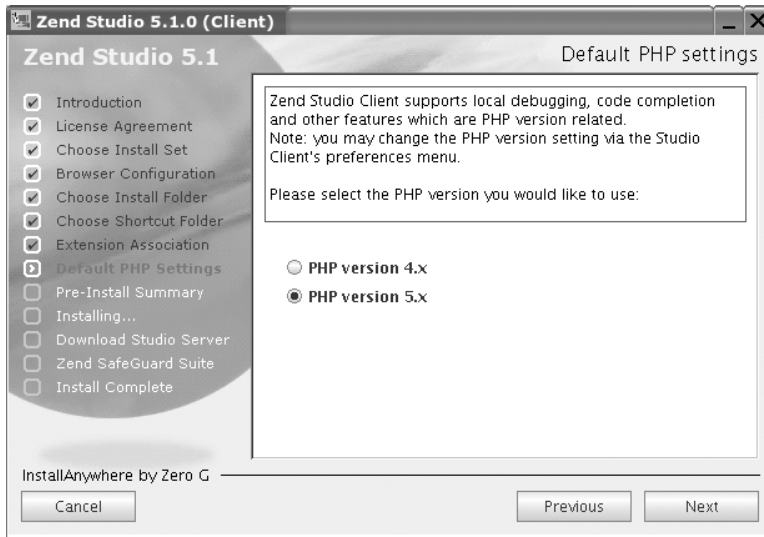


Abbildung 4.10: PHP 5-Auswahl im Setup des Zend Studio Clients

4. Im nächsten Fenster wird die *Zend SafeGuard Suite* als Option ausgewählt.



Abbildung 4.11: Das separate Setup des Zend SafeGuard

5. Nach erfolgreicher Installation kann der Client mit normalen Benutzerrechten gestartet werden:

```
/usr/local/Zend/ZendStudioClient-5.1.0/bin> ./runStudio_unix.sh
```

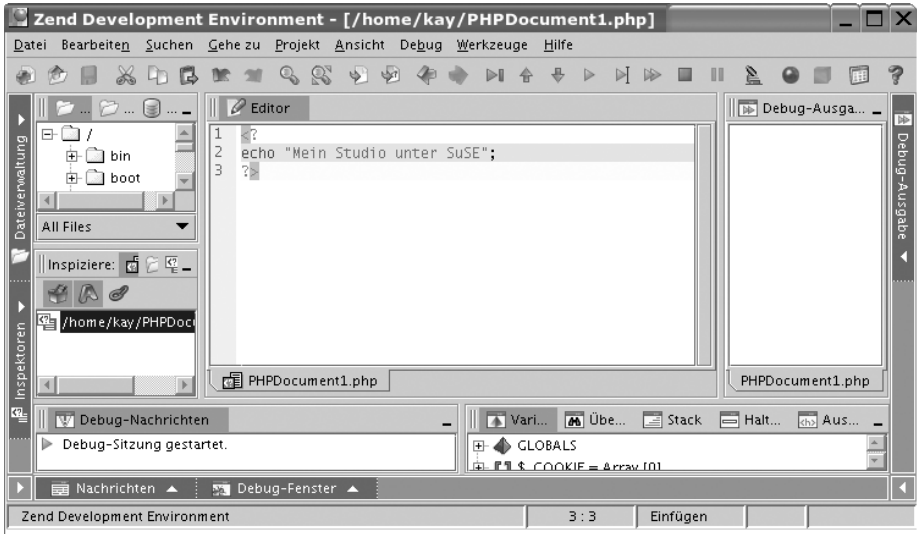


Abbildung 4.12: Der Zend Studio Client unter Linux

Installation des Studio Servers

Für die Installation des Zend Studio Server benötigen Sie *root*-Rechte.

1. Entpacken Sie das Paket mit folgendem Befehl:

```
tar xvfz ZendStudioServer-5.1.0a-linux-glibc21-i386.tar.gz
```

2. Wechseln Sie ins Zielverzeichnis und starten Sie das Setup:

```
cd ZendStudioServer-5.1.0a-linux-glibc21-i386
```

Wir führen das Setup auf Kommandoebene aus:

```
./install-tty
```

Falls Sie eine grafische GUI bevorzugen, wählen Sie:

```
./install
```

3. Folgende Angaben sind im Setup zu machen:

Welche Apache-Installation soll benutzt werden?
Der Pfad zur Apache Control Utility?

Die Eigene („Existing“)
/opt/lamp/bin/apachectl

```

Das Verzeichnis mit der php.ini lautet?      /opt/lampp/etc
Das htdocs-Verzeichnis lautet?              /opt/lampp/htdocs
Welche Hosts sind am Zend Server erlaubt?   127.0.0.1 (als Beispiel)
PHP-Skripte laufen im Apache als User?     nobody

```

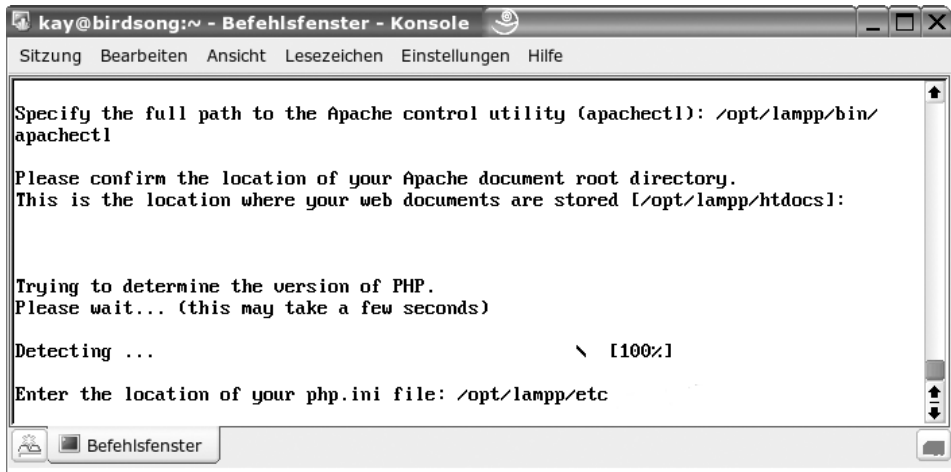


Abbildung 4.13: Das Setup des Studio Servers klassisch über die Konsole

Zusätzlich legen Sie hier noch Ihr administratives Passwort zum Login in die Konfigurationsoberfläche des Zend Servers fest. Das Setup beendet sich schließlich nach einem erfolgreichen Neustart des Apache HTTPD. Den Server erreichen Sie danach auf dem lokalen Rechner über die URL <http://localhost/ZendStudioServer/>.

Wiederherstellung der XAMPP-Konfiguration unter Linux

In manchen Fällen wird die Restauration der ursprünglichen `php.ini` nötig. Hierfür reicht das Überschreiben des vom Zend Server angelegten symbolischen Links mit der Sicherheitskopie `php.ini-zend_studio_server.bak`:

```
cp /opt/lampp/etc/php.ini-zend_studio_server.bak /opt/lampp/etc/php.ini
```

Starten Sie danach den Apache neu:

```
/opt/lampp/bin/apachectl restart
```

4.1.4 Freie Alternativen

Das Zend-Projekt bietet nicht die einzige Entwicklerumgebung für PHP an. Leider wurden aber aus vielen der zuvor freien Tools kostenpflichtige Softwareangebote. Darüber hinaus sind viele dieser Programme nur für Windows-Betriebssysteme verfügbar.

Eine Empfehlung für den einen oder anderen Editor oder ein Entwicklerstudio unterlassen wir hier besser. Zu groß ist die Gefahr, einen quasi »ideologischen« Streit über die beste Entwicklerumgebung auszulösen.

4.1.5 Weaverslave

Einen Editor für Windows möchten wir hier aber doch kurz ansprechen: *Weaverslave* (<http://www.weaverslave.ws>), der eine Reihe von Programmiersprachen unterstützt, darunter auch PHP. Besonders ansprechend fanden wir die Tatsache, dass das Programm im entsprechenden ZIP-Paket direkt gestartet werden konnte. Auch die einfache Einbindung von XAMPP als Server zur Ausführung der Scripts überzeugte.

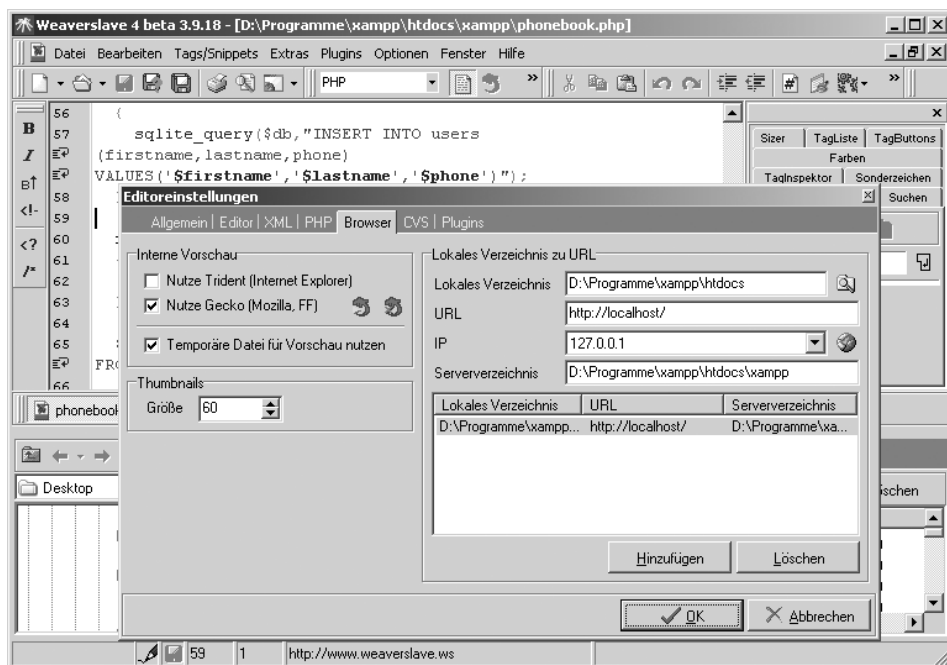


Abbildung 4.14: Der freie Editor Weaverslave für Windows

Für die Programmierer unter Linux gibt es hier leider weniger Auswahl. Ein Editor mit ähnlicher Funktionalität wie der eben vorgestellte Weaverslave ist der freie Allrounder *Bluefish* (<http://bluefish.openoffice.nl>).

4.2 PDF generieren

HTML-Seiten mit PHP zu generieren ist schön, einfach und auch genau das, wofür PHP entwickelt wurde. Aber HTML stößt schnell an seine Grenzen, wenn man den Online-Bereich verlässt: Will man HTML-Dateien ausdrucken, erreicht man selten ein brauch-

bares Ergebnis. Für den privaten Bereich genügt es vielleicht, aber eine Rechnung oder ein Produktblatt aus HTML zu drucken macht keinen professionellen Eindruck: Die Ränder sind ungleichmäßig, das Logo sitzt nicht genau an der richtigen Stelle usw.

Eine Lösung zu diesem Problem besteht darin, PHP so zu nutzen, dass man damit auch PDFs generieren kann.

PHP ist von sich aus nicht in der Lage, PDFs zu erzeugen, und daher muss diese Fähigkeit PHP erst noch nachträglich beigebracht werden. Dazu steht etwa ein halbes Dutzend Erweiterungen zur Verfügung.

Unterteilen kann man diese Erweiterungen in zwei Gruppen: Erweiterungen, die fest in PHP eingebunden werden und meist in C geschrieben wurden, und Erweiterungen, die in PHP geschrieben wurden. Mehr zu den Erweiterungsmöglichkeiten von PHP finden Sie in Kapitel 20, »PHP erweitern«.

In XAMPP sind zwei Bibliotheken enthalten, die es PHP ermöglichen, PDFs zu generieren: `pdf class` und `FPDF`.

Die Entscheidung, in diesem Fall auf Bibliotheken und nicht auf Erweiterungen zurückzugreifen, basiert auf folgenden Überlegungen:

- Keine der PDF-Erweiterungen für PHP ist komplett frei erhältlich.
- Kein normaler Provider bietet diese Erweiterungen an: Eine Migration des unter XAMPP entwickelten PHP-Programms auf den Providerserver würde also fehlschlagen.
- PHP-Bibliotheken lassen sich leicht überall hinzufügen: Um PHP-Erweiterungen zu installieren, benötigt man Administratorrechte.

4.2.1 pdf class

Die *pdf class* von Wayne Munro ist schon seit den ersten Tagen von XAMPP mit in diesem Paket enthalten. In dem CD-Verwaltungs-Beispielprogramm wird diese Bibliothek verwendet, um aus den CD-Beständen eine ausdrückbare Liste im PDF-Format zu generieren.

Der Programmcode ist recht einfach und befindet sich in den ersten Zeilen 2 bis 28 der Datei `htdocs/xampp/cds.php` (bzw. `htdocs\xampp\cds.php` im Falle von Windows).

Gleich am Anfang findet sich folgende Zeile:

```
if($_REQUEST['action']=='getpdf')  
{
```

Wird die CD-Verwaltungsdemo mit dem URL-Parameter `action=getpdf` gestartet, dann soll sie nicht wie gewohnt die HTML-Seite produzieren, sondern die CD-Daten als PDF ausgeben.

```
mysql_connect("localhost","root","");  
mysql_select_db("cdcol");
```

Hier wird lediglich die Datenbankverbindung zu MySQL hergestellt und die Datenbank `cdcol` ausgewählt.



Hinweis

Standardmäßig hat der MySQL-Benutzer *root* in XAMPP kein Passwort gesetzt, daher ist in der `mysql_connect()`-Funktion oben auch der dritte Parameter leer. Sollten Sie aber dem MySQL-*root* inzwischen ein Passwort gesetzt haben, dann muss dieses hier an der dritten Stelle angegeben werden.

Im nächsten Schritt wird die PHP-Bibliothek geladen:

```
include ('class.ezpdf.php');
```

In der Linux-Version von XAMPP befindet sich diese Datei unter `/opt/lampp/lib/php/class.ezpdf.php`, bei Windows unter `<XAMPP>\php\pear\class.ezpdf.php`. Ein Pfad muss aber in beiden Fällen nicht angegeben werden.

PDF vorbereiten

Bevor mit der eigentlichen Erstellung des PDFs begonnen werden kann, muss ein Objekt erstellt werden, das das PDF repräsentiert:

```
$pdf = new Cezpdf();
```

Mit dieser Anweisung wird ein neues PDF-Objekt erzeugt. Über den Konstruktor können (müssen aber nicht) zusätzlich noch Seitengröße und Orientierung des PDFs gesetzt werden:

```
$pdf = new Cezpdf($paper='a4', $orientation='portrait');
```

Für die Seitengröße `$paper` sind folgende Werte möglich:

Format	Seitengröße (in cm)
4A0	168,2 x 237,8
2A0	118,9 x 168,2
A0	84,1 x 118,9
A1	59,4 x 84,1
A2	42 x 59,4
A3	29,7 x 42
A4	21 x 29,7

Tabelle 4.2: Von pdf class unterstützte Seitengrößen

Format	Seitengröße (in cm)
A5	14,8 x 21
A6	10,5 x 14,8
A7	7,4 x 10,5
A8	5,2 x 7,4
A9	3,7 x 5,2
A10	2,6 x 3,7
B0	100 x 141,4
B1	70,7 x 100
B2	50 x 70,7
B3	35,3 x 50
B4	25 x 35,3
B5	17,6 x 25
B6	12,5 x 17,6
B7	8,8 x 12,5
B8	6,2 x 8,8
B9	4,4 x 6,2
B10	3,1 x 4,4
C0	91,7 x 129,7
C1	64,8 x 91,7
C2	45,8 x 64,8
C3	32,4 x 45,8
C4	22,9 x 32,4
C5	16,2 x 22,9
C6	11,4 x 16,2
C7	8,1 x 11,4
C8	5,7 x 8,1
C9	4 x 5,7
C10	2,8 x 4
RA0	86 x 122
RA1	61 x 86
RA2	43 x 61
RA3	30,5 x 43

Tabelle 4.2: Von pdfclass unterstützte Seitengrößen (Forts.)

Format	Seitengröße (in cm)
RA4	21,5 x 30,5
SRA0	90 x 128
SRA1	64 x 90
SRA2	45 x 64
SRA3	32 x 45
SRA4	22,5 x 32
LETTER	21,59 x 27,94
LEGAL	21,59 x 35,56
EXECUTIVE	18,41 x 26,67
FOLIO	21,59 x 33,02

Tabelle 4.2: Von pdfclass unterstützte Seitengrößen (Forts.)

Falls diese Formate nicht reichen oder man ein ganz besonderes Papierformat benötigt, kann man als Seitengröße ein Array mit zwei Werten übergeben:

```
$pdf = new Cezpdf($paper=array(12,12),
                 $orientation='portrait');
```

Mit dieser Anweisung wird ein 12 x 12 cm großes quadratisches PDF generiert.

Der zweite Parameter `$orientation` unterscheidet zwischen 'portrait' für Hochkant- und 'landscape' für Querformat.

Lässt man die Parameter weg, werden die Standardwerte 'A4' und 'portrait' benutzt.

Alle Vorgänge, die man nun an dem PDF vornehmen möchte, erfolgen über Methodenaufrufe dieses Objekts.

Schrift auswählen

Damit wäre nun das PDF-Objekt erstellt. Als Nächstes können die ersten Inhalte in das PDF ausgegeben werden. Zunächst muss aber noch ein weiterer Schritt folgen: nämlich die Wahl der zu verwendenden Schrift.

Im CD-Verwaltungsbeispiel von XAMPP geschieht das mit folgender Anweisung:

```
$pdf->selectFont('/opt/lampp/lib/fonts/Helvetica.afm');
```

bzw. in der Windows-Version:

```
$pdf->selectFont('../pdf/fonts/Helvetica');
```

Mit diesem Schritt wird der Zeichensatz Helvetica gewählt. Alles, was ab jetzt in das PDF »gedruckt« wird, erscheint in dieser Schrift.

php class unterstützt die so genannten *Standard Postscript Fonts*: Courier, Helvetica, Times, Symbol und Zapf-Dingbats. Dabei sind die in Tabelle 4.3 angegebenen Bezeichnungen zu verwenden.

Schriften von pdf class

Courier-Bold.afm
Courier-BoldOblique.afm
Courier-Oblique.afm
Courier.afm
Helvetica-Bold.afm
Helvetica-BoldOblique.afm
Helvetica-Oblique.afm
Helvetica.afm
Symbol.afm
Times-Bold.afm
Times-BoldItalic.afm
Times-Italic.afm
Times-Roman.afm
ZapfDingbats.afm

Tabelle 4.3: Diese Schriften unterstützt pdf class standardmäßig. Die Endung .afm ist optional und kann auch weggelassen werden.

Ausgabe von Texten

Um Texte in das PDF auszugeben, bedient man sich der Methode `ezText()` des PDF-Objekts.

```
$pdf->ezText('CD Collection',14);  
$pdf->ezText('@ 2002/2003 Kai Seidler, oswald@apachefriends.org, GPL',10);  
$pdf->ezText('',12);
```

Diese drei Zeilen geben die Überschrift und das Copyright, gefolgt von einer Leerzeile, in das PDF aus. Die `ezText()`-Methode bekommt zwei oder drei Parameter: Der erste Parameter ist der auszugebende Text, der zweite Parameter die Schriftgröße.

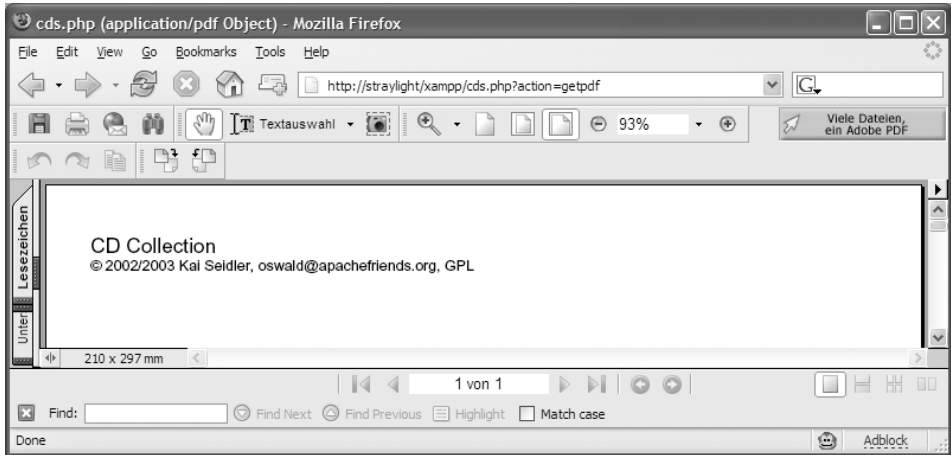


Abbildung 4.15: Erste Textausgabe mit pdf class

Mit Hilfe des optionalen dritten Parameters können auch noch weitergehende Formatierungsoptionen für den Text eingestellt werden:

Formatbezeichner	Bedeutung
'left'	Linker Rand (relativ)
'right'	Rechter Rand (relativ)
'aleft'	Linker Rand (absolut), überschreibt 'left'
'aright'	Rechter Rand (absolut), überschreibt 'right'
'justification'	Formatierung. Möglich sind 'left' (linksbündig), 'right' (rechtsbündig), 'center' oder 'centre' (zentriert) und 'full' für Blocksatz.
'leading'	Absolute Zeilenhöhe (unabhängig von der Schriftgröße), überschreibt 'spacing'
'spacing'	Relative Zeilenhöhe in Abhängigkeit von der Schriftgröße, zum Beispiel: '2' für doppelte Zeilenhöhe

Tabelle 4.4: Absatzformatierungen von pdf class

Zum Beispiel erzeugt folgende Anweisung einen Text, der in Blocksatz mit doppelter Zeilenhöhe ausgegeben wird:

```
$pdf->ezText('Weit hinten, hinter den Wortbergen, fern der Länder Vokalien und Konsonantien leben die Blindtexte.', 10, array('justification' => 'full', 'spacing' => 2));
```

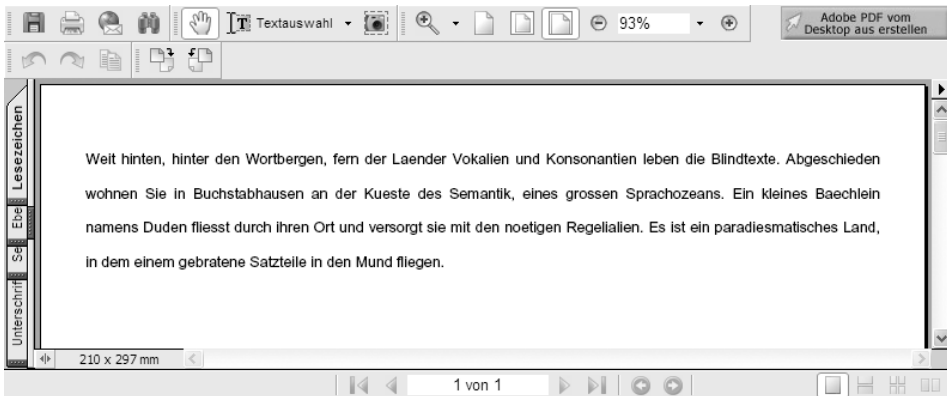


Abbildung 4.16: Blocksatz mit *pdf class*

In fast allen Texten, die in *pdf class* ausgegeben werden, können die beiden HTML-Tags `<b>` und `<i>` verwendet werden, um Textpassagen fett oder kursiv hervorzuheben.

Ausgabe von Tabellen

Aber sehr oft will man nicht nur Text, sondern auch Tabellen ausgeben. Dazu bietet *pdf class* mit der Methode `ezTable()` eine komfortable Möglichkeit. Sehen wir uns zunächst das CD-Verwaltungsbeispiel an.

```
$result=mysql_query("SELECT id,titel,interpret,jahr FROM cds
                    ORDER BY interpret;");

$i=0;
while( $row=mysql_fetch_array($result) )
{
    $data[$i]=array('interpret'=>$row['interpret'],
                    'titel'=>$row['titel'],
                    'jahr'=>$row['jahr']);
    $i++;
}
```

In diesem Abschnitt werden die CD-Daten mit `mysql_query()` aus der Datenbank gelesen und dann Eintrag für Eintrag in der `while()`-Schleife in das Array `$data` geschrieben.

```
$pdf->ezTable($data, "", "", array('width'=>500));
```

Mittels der Methode `ezTable()` wird dann das vorbereitete Array als Tabelle im PDF ausgegeben.

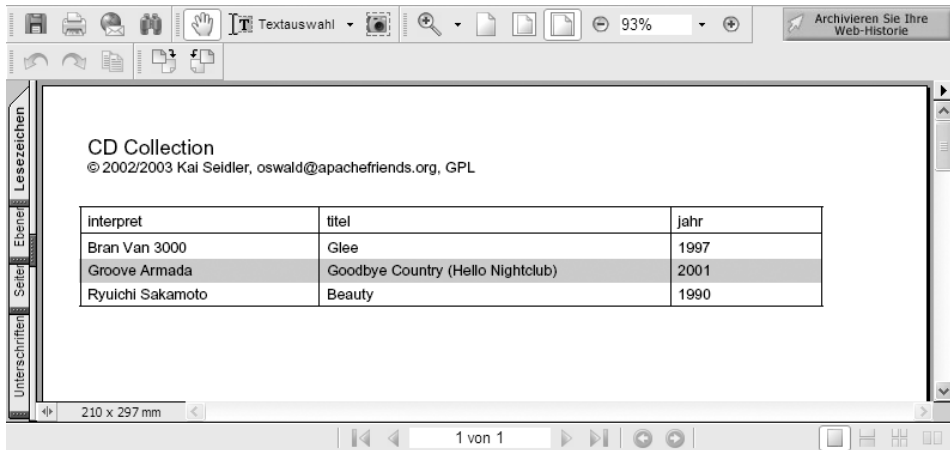


Abbildung 4.17: Einfache Tabelle mit ezTable()

ezTable() hat vier Parameter, wobei nur der erste unbedingt angegeben werden muss und die restlichen drei optional sind:

- Der erste Parameter bestimmt das Array, dessen Daten ausgegeben werden sollen.
- Der zweite Parameter definiert die Spaltentitel.
- Der dritte Parameter enthält die Tabellenüberschrift.
- Und der vierte Parameter bestimmt das Layout der Tabelle.

Ersetzen wir doch einfach den obigen Aufruf von ezTable() durch folgende Zeilen:

```
$title=array('titel'=>'<b>Titel</b>',
            'interpret'=>'<b>Interpet</b>',
            'jahr'=>'<b>Jahr</b>');

$options=array('width'=>500);

$heading="<b><i>CD-Liste</i></b>";

$pdf->ezTable($data, $title, $heading, $options);
```

Jeder Spalte wird hier nun durch das Array \$title ein Titel zugewiesen. Der Tabelle wird über die Variable \$heading eine Überschrift verpasst. In beiden Fällen kann sogar mit ein wenig HTML die Darstellung verändert werden.

Durch die Verwendung der zusätzlichen Variablen \$title, \$options und \$heading ist der Aufruf von ezTable() ein bisschen übersichtlicher geworden. Ansonsten haben sie keine Bedeutung.

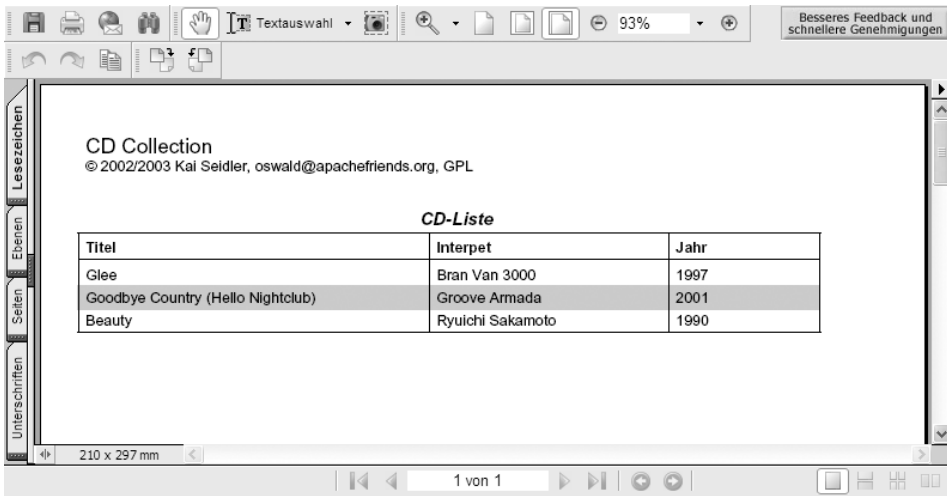


Abbildung 4.18: Leicht veränderte Tabelle mit `ezTable()`

Ausgabe des PDFs

Nachdem alle Ausgaben in das PDF-Dokument gemacht wurden, muss die PDF-Erstellung mit folgendem Befehl abgeschlossen werden:

```
$pdf->ezStream();
exit;
```

Mit `$pdf->ezStream()` wird das PDF-Dokument ausgegeben und anschließend das Beispiel-PHP-Programm beendet.

Hier noch einmal das gesamte Programm im Zusammenhang:

Listing 4.1: Der Teil der CD-Verwaltungsdemo, der für die Generierung des PDFs verantwortlich ist (aus `htdocs/kampp/cds.php`)

```
if($_REQUEST['action']=="getpdf")
{
    mysql_connect("localhost","root","");
    mysql_select_db("cdcol");

    include ('class.ezpdf.php');
    $pdf = new Cezpdf();
    $pdf->selectFont('/opt/lampp/lib/fonts/Helvetica.afm');

    $pdf->ezText('CD Collection',14);
    $pdf->ezText('© 2002/2003 Kai Seidler, oswald@apachefriends.org, GPL',10);
    $pdf->ezText('',12);

    $result=mysql_query("SELECT id,titel,interpret,jahr FROM cds
```

```

ORDER BY interpret;");

$i=0;
while( $row=mysql_fetch_array($result) )
{
    $data[$i]=array('interpret'=>$row['interpret'],
                   'titel'=>$row['titel'],'jahr'=>$row['jahr']);
    $i++;
}

$pdf->ezTable($data,""," ",array('width'=>500));

$pdf->ezStream();
exit;
}

```

Weblinks

Die Bibliothek *pdf class* finden Sie unter folgender URL:

<http://www.ros.co.nz/pdf/>

Die Referenzanleitung zu *pdf class* steht unter folgender Adresse zur Verfügung:

<http://www.ros.co.nz/pdf/readme.pdf>

4.2.2 FPDF

Die FPDF-Bibliothek von Olivier Plathey ist seit XAMPP 1.4.16 mit im Paket enthalten. Ähnlich wie die *pdf class* ist sie ebenfalls freie Software und kann lizenzfrei in jedem Kontext eingesetzt werden.

Wie schon für die *pdf class* gibt es im XAMPP die CD-Verwaltung auch als FPDF-Beispielanwendung. Die folgenden Programmzeilen stammen aus der Datei `htdocs/xampp/cds-fpdf.php` bzw. `htdocs\xampp\cds-fpdf.php`. Die Zeilen, die identisch mit der *pdf class*-Version sind, werden hier allerdings weggelassen und nicht noch einmal erläutert.

PDF vorbereiten

Genauso wie bei *pdf class* muss FPDF zunächst geladen werden:

```
include ('fpdf.php');
```

Durch diesen Befehl wird die FPDF-Klasse in PHP eingebunden, so dass im nächsten Schritt das PDF-Objekt erstellt werden kann:

```
$pdf = new FPDF();
```

Auf der Basis dieses Objekts wird nun das PDF-Dokument erstellt. Ähnlich wie bei `pdf class` können auch hier über den Konstruktor Seitengröße und Seitenorientierung eingestellt werden:

```
$pdf = new FPDF('P','mm','A4');
```

Der erste Parameter bestimmt das Seitenorientierung: 'P' für Hochformat (*portrait*) und 'L' für Querformat (*landscape*).

Der zweite Parameter legt fest, welche Einheiten für dieses PDF verwendet werden sollen. Möglich sind hier:

Parameter	Bedeutung
pt	1/72 mm
mm	1 mm
cm	1 cm
in	1 Zoll = 2,54 cm

Tabelle 4.5: Diese Maßeinheiten versteht FPDF

Der dritte Parameter bestimmt die Seitengröße und kann folgende Werte annehmen:

Format	Seitengröße (in cm)
A3	29,7 x 42
A4	21 x 29,7
A5	14,85 x 21
LETTER	21,59 x 27,94
LEGAL	21,59 x 35,56

Tabelle 4.6: Von FPDF unterstützte Seitengrößen

Man sieht hier deutlich, dass im Gegensatz zu `pdf class` die FPDF-Bibliothek viel weniger vordefinierte Seitengrößen anbietet. Allerdings kann man auch hier ganz einfach sein individuelles Seitenformat definieren, indem man als dritten Parameter einfach ein Array mit zwei Feldern übergibt:

```
$pdf=new FPDF('P','mm', array(120,120));
```

Mit diesem Konstruktor wird eine 12 x 12 cm große Seite definiert. Die Werte in dem Größen-Array beziehen sich auf die Einheit, die als zweiter Parameter angegeben wurde, hier also 'mm'.

Macht man beim Konstruktor keine Angaben, werden die Standardwerte benutzt: Hochformat, mm und A4.

Schrift auswählen

Anders als bei der *pdf class* muss bei FPDF zunächst eine leere Seite »erstellt« werden, bevor mit den Ausgaben in das PDF begonnen werden kann:

```
$pdf->AddPage();
```

Im nächsten Schritt sollte dann die Schriftart festgelegt werden:

```
$pdf->SetFont('Helvetica','',14);
```

Mit dieser Anweisung werden die Schriftart Helvetica und die Schriftgröße 14 Punkt gewählt.

FPDF unterstützt standardmäßig folgende Schriften:

- 'Courier'
- 'Helvetica' (synonym 'Arial')
- 'Times'
- 'Symbol'
- 'ZapfDingbats'

Der zweite Parameter (hier leer gelassen) definiert den Schrifttyp:

Parameter	Bedeutung
' ' (leer)	normale Schrift
'B'	fett
'I'	kursiv
'U'	unterstrichen

Tabelle 4.7: Schrifttypen in FPDF

Bei der Wahl des Schrifttyps sind auch Kombinationen möglich. So ergibt 'BUI' eine fette, unterstrichene und kursive Ausgabe.

Ausgabe von Texten

Nachdem die Schrift ausgewählt wurde, kann mit der Ausgabe von Inhalten in das PDF begonnen werden:

```
$pdf->Write(5, 'CD Collection');
```

Mit dieser Anweisung wird der Text 'CD Collection' in das PDF ausgegeben.

Achtung



Der erste Parameter definiert nicht – wie man zunächst vermuten könnte – die Schriftgröße, sondern die Zeilenhöhe, und zwar in dem Maß, das im Konstruktor definiert wurde, in diesem Fall also 5 mm.

Durch die `Write()`-Methode wird kein Zeilenumbruch ausgelöst. Möchte man also nach einem `Write()` am Anfang der nächsten Zeile weitermachen, muss man einen Zeilenumbruch befehlen:

```
$pdf->Ln();
```

Über einen Parameter kann hier noch die Höhe des Zeilenumbruchs angegeben werden. Fehlt diese Angabe, wie in diesem Beispiel, dann wird die Höhe der letzten Ausgabe verwendet, also die 5 mm von der `$pdf->Write()`-Anweisung oben.

```
$pdf->SetFontSize(10);  
$pdf->Write(5, '@ 2002/2003 Kai Seidler,  
oswald@apachefriends.org, GPL');  
$pdf->Ln();
```

In diesem Beispiel wird die Schriftgröße auf 10 gesetzt und zugleich werden der Copyright-Hinweis und ein Zeilenumbruch ausgegeben. Zur besseren Optik folgt danach noch eine Leerzeile:

```
$pdf->Ln(5);
```

Im nächsten Teil des Programms wird die Tabelle mit den CD-Daten ausgegeben.

Ausgabe von Tabellen

Anders als bei *pdf class* existiert bei FPDF keine extra Funktion, um Tabellen auszugeben. Tabellen werden dadurch generiert, dass man einfach mit der `Cell()`-Methode nacheinander Textzellen ausgibt, die dann eine Tabelle bilden.

```
$pdf->SetFont('Helvetica','B',10);  
$pdf->Cell(40,7,'interpret',1);  
$pdf->Cell(80,7,'titel',1);  
$pdf->Cell(40,7,'jahr',1);  
$pdf->Ln();
```

In der ersten Zeile werden Angaben zur Schrift gemacht, in unserem Beispiel wählen wir Helvetica, Fettdruck und Schriftgrad 10. Diese Schrift soll für den Titel unserer

Tabelle benutzt werden. Danach werden hintereinander – ohne einen Zeilenumbruch – mit der `Cell()`-Methode drei Textzellen ausgegeben.

Die `Cell()`-Methode bekommt vier Parameter (wobei nur der erste Pflicht ist, alle weiteren sind optional):

- Der erste Parameter definiert die Breite einer Zelle. In unserem Fall haben wir Millimeter als Maßeinheit für FPDF gewählt, folglich ist dieser Wert hier in mm anzugeben.
- Der zweite Parameter bestimmt die Höhe einer Zelle. Ebenso wie der letzte Parameter wird in diesem Fall der Wert in mm angegeben.
- Der dritte Parameter enthält den Text, der ausgegeben werden soll.
- Der vierte Parameter gibt an, ob ein Rahmen um diese Zelle gezeichnet werden soll. 0 bedeutet keinen Rahmen und 1 bedeutet, dass ein Rahmen gezeichnet werden soll. Anstelle von 1 kann hier auch eine Kombination aus den Buchstaben 'L' (left), 'T' (top), 'R' (right) oder 'B' (bottom) angegeben werden. Bei 'LT' wird so zum Beispiel nur links und oben ein Rahmen gezeichnet.

Hinweis



Die `Cell()`-Methode hat noch vier weitere optionale Parameter, auf die hier allerdings nicht näher eingegangen wird.

Nachdem der Tabellenkopf ausgegeben wurde, wird die Schrift auf ein normales Helvetica gesetzt:

```
$pdf->SetFont('Helvetica','',10);
```

Im nächsten Schritt wird die SQL-Abfrage an die Datenbank geschickt und die CD-Daten werden gelesen:

```
$result=mysql_query("SELECT id,titel,interpret,jahr FROM cds
                    ORDER BY interpret;");
```

Danach werden die CD-Daten Zeile für Zeile ausgegeben:

```
while( $row=mysql_fetch_array($result) )
{
    $pdf->Cell(40,7,$row['interpret'],1);
    $pdf->Cell(80,7,$row['titel'],1);
    $pdf->Cell(40,7,$row['jahr'],1);
    $pdf->Ln();
}
```

Mittels einer `while()`-Schleife, die genauso lange läuft, wie der `mysql_fetch_array()`-Befehl Daten in dem Ergebnis der Datenbank findet, wird jeweils mit drei `Cell()`-Aufrufen eine Zeile der Tabelle ausgegeben. Am Ende jeder Tabellenzeile wird mit der `Ln()`-Methode an den Anfang der nächsten Zeile gesprungen.

Ganz zum Schluss wird mit der `output()`-Methode FPDF angewiesen, das PDF-Dokument nun endlich auszugeben:

```
$pdf->output();
```

Im Browser sieht das Ganze dann so aus:

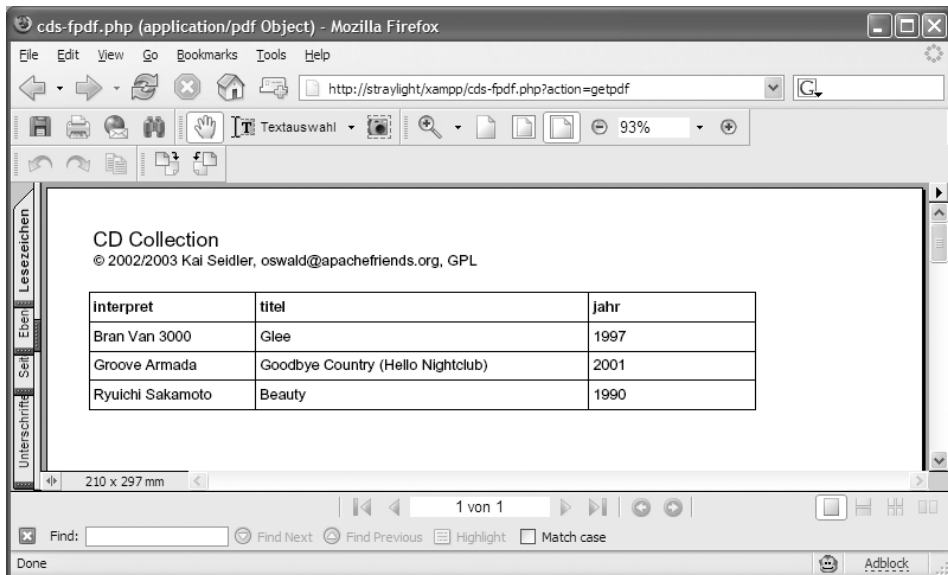


Abbildung 4.19: Das Ergebnis der CD-Verwaltung mit FPDF

Weblinks

Die FPDF-Bibliothek kann unter folgender URL heruntergeladen werden:

<http://www.fpdf.org/>

Weitere Informationen mit zahlreichen Add-Ons stehen auf der deutschsprachigen FPDF-Seite zur Verfügung:

<http://www.fpdf.de/>

4.3 XML/RSS

XML, die *Extensible Markup Language*, ist eines der großen Buzzwords der letzten Jahre. Inzwischen hat sie sich aber tatsächlich in vielen Bereichen als Datenaustauschformat etabliert und verdient eine genauere Betrachtung.

Wie immer in diesem Buch werden wir nicht umfassend auf das Thema XML als solches eingehen können, sondern es nur im Zusammenhang mit XAMPP betrachten. Trotzdem werden wir aber auch hier nicht ganz um trockene Theorie herumkommen.

- Möglichkeiten der Arbeit mit XML
- Lesen eines XML-Dokuments mit SAX
- Lesen eines XML-Dokuments mit DOM
- Generieren von XML aus einer MySQL-Datenbank
- Erstellen eines RSS-Feeds

4.3.1 DOM und SAX

Um mit XML zu arbeiten, existieren zwei Haupttechniken oder -wege: DOM und SAX. (Techniker scheinen wirklich einfach eine große Vorliebe für 3-buchstabile Abkürzungen zu haben.)

DOM

DOM steht für *Document Object Model* und beschreibt den Zugriff auf ein XML-Dokument über hierarchisch angeordnete Objekte. Es gilt als die erste Wahl, um mit XML zu arbeiten. Zum einen, weil es ein vom *World Wide Web Consortium* (W3C) entwickelter Standard ist, und zum anderen, weil es – anders als SAX (siehe unten) – auch die Möglichkeit bietet, XML-Dokumente zu erzeugen. Nachteilig an DOM ist allerdings, dass es sich überhaupt nicht für sehr große XML-Dokumente eignet, da immer zuerst das gesamte XML-Dokument geladen werden muss, damit es verarbeitet werden kann.

Neben der Möglichkeit, XML zu erstellen, hat DOM aber auch noch einen weiteren Vorteil: Mit dem sog. *DOM HTML* können (X)HTML-Seiten mittels DOM generiert werden.

SAX

Die Alternative zu DOM ist SAX. SAX steht für *Simple API for XML* und wie DOM bietet es die Möglichkeit, XML-Dokumente zu lesen. Allerdings bietet es wirklich nur diese Möglichkeit. Erstellen kann man mit SAX keine XML-Dokumente.

SAX funktioniert eventbasiert, d. h., man definiert, dass zu bestimmten Ereignissen bestimmte Aktionen ausgeführt werden. Solche Ereignisse können zum Beispiel das Öffnen oder Schließen eines XML-Tags oder das Auftreten von Daten in einem XML-Dokument sein.

Da SAX nicht wie DOM das gesamte XML-Dokument auf einmal liest, sondern es quasi wie einen Stream behandelt, ist SAX in der Lage, auch mit sehr großen Dokumenten umzugehen.

Auch wenn das auf den ersten Blick recht kompliziert klingt, sieht die Praxis zum Glück einfacher aus, als man zunächst vermuten würde.

Werfen wir zunächst einmal einen genaueren Blick auf SAX.

4.3.2 SAX

Wie schon weiter oben erwähnt, behandelt SAX ein XML-Dokument wie einen Strom von Daten, der Element für Element von SAX verarbeitet wird. Wie SAX diesen XML-Strom bearbeitet, wird über so genannte Events (Ereignisse) bestimmt.

Die drei wichtigsten Events dürften die folgenden sein:

- Das *startElement*-Event tritt ein, sobald ein XML-Tag geöffnet wird.
- Das *endElement*-Event tritt ein, sobald ein XML-Tag geschlossen wird.
- Und das Event *characterData* tritt in dem Moment ein, in dem Daten im XML-Dokument gefunden werden.

Diese Events treten also ganz automatisch ein, wenn der *SAX-Parser* das XML-Dokument durchgeht. Um nun mit diesen Events auch etwas anfangen zu können, werden pro Event so genannte Callback-Funktionen definiert. So kann man zum Beispiel bestimmen, dass, wenn das *characterData*-Event eintritt, die Funktion `verarbeiteDaten()` aufgerufen werden soll.

Um aber mit XML zu arbeiten, benötigen wir zunächst ein kleines XML-Beispiel.

Ein XML-Beispiel

In XAMPP befindet sich unter `/opt/lampp/htdocs/xampp/contrib/cds.xml` bzw. `<xampp>\xampp\contrib\cds.xml` ein kleines Beispiel für ein einfaches XML-Dokument. Wer bereits die Abschnitte über Datenbanken in diesem Kapitel gelesen hat, wird den Inhalt dieser Datei wiedererkennen: Es sind die drei CDs aus dem CD-Verwaltungsbeispiel. Demjenigen, der dieses Beispiel nicht mehr sehen kann, sei gesagt: Es sind wirklich gute CDs. Ich hoffe, das tröstet ein wenig.

Hinweis



Am besten sieht man sich XML-Dateien übrigens in einem Webbrowser wie Firefox oder IE an. Vorteil ist hier: Browser überprüfen nebenbei, ob das XML-Dokument auch tatsächlich korrekt ist, und außerdem stellen sie es in einer übersichtlichen, eingerückten Form dar.

Übrigens ist der IE bei der Überprüfung ein wenig penibler als der Mozilla Firefox. Wer also einen Fehler in einem XML-Dokument vermutet, sollte es mit dem IE überprüfen und nicht mit Firefox.

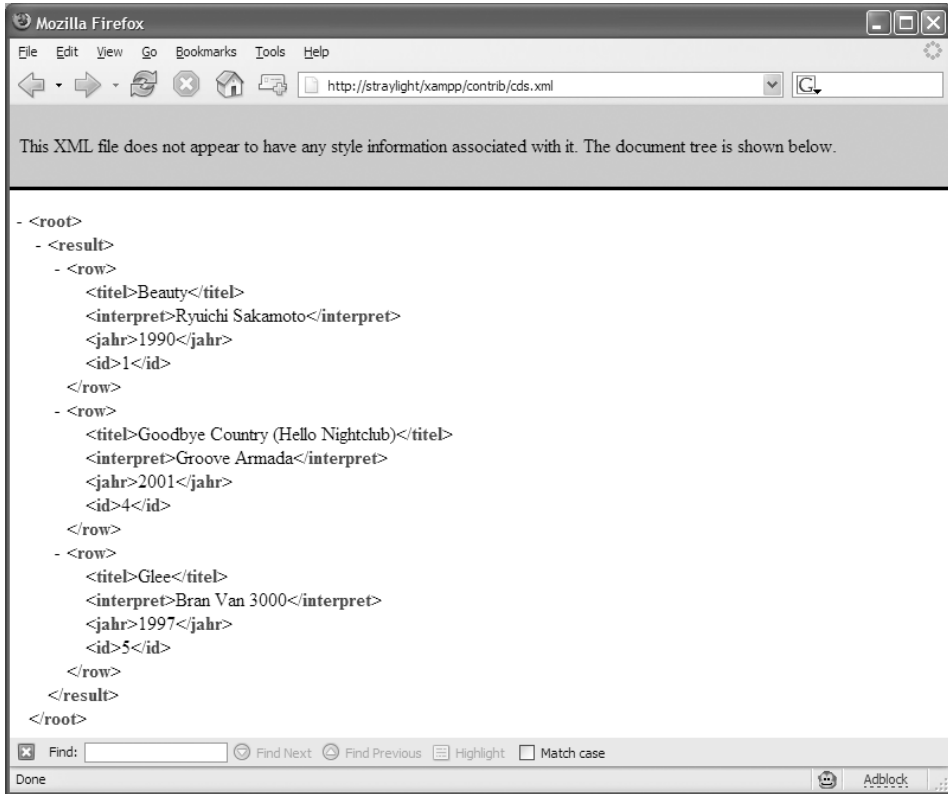


Abbildung 4.20: Kein originelles, aber praktisches Beispiel für ein XML-Dokument

Unter `/opt/lampp/htdocs/xampp/contrib/sax.php` bzw. `<xampp>\htdocs\xampp\contrib\sax.php` findet sich ein kleines Beispiel für eine einfache SAX-Implementierung, das wir uns auf den nächsten Seiten etwas genauer anschauen werden.

Den XML-Parser einrichten und Callback-Funktionen bestimmen

Ziemlich zu Beginn des Programms werden zunächst der SAX-Parser eingerichtet und Callback-Funktionen für die Events `startElement`, `endElement` und `characterData` definiert.

```

$parser = xml_parser_create();
xml_set_element_handler($parser, 'startE', 'endE');
xml_set_character_data_handler($parser, 'characterD');

```

Mit der Funktion `xml_set_element_handler()` werden für den XML-Parser die Callback-Funktionen für das `startElement`- und `endElement`-Event definiert: Der erste Parameter

ist der Handler des XML-Parsers; der zweite Parameter definiert die Callback-Funktion `startE()` für das `startElement`-Event und der dritte Parameter die Callback-Funktion `endE()` für das `endElement`-Event.

Die Funktion `xml_set_character_data_handler()` legt im nächsten Schritt auf ähnliche Weise die Callback-Funktion `characterD()` für das `characterData`-Event fest.

Zusammenfassend sehen die Zuweisungen von Events und Callback-Funktionen wie folgt aus:

Event	Callback-Funktion
<code>startElement</code> -Event	<code>startE()</code>
<code>endElement</code> -Event	<code>endE()</code>
<code>characterData</code> -Event	<code>characterD()</code>

Tabelle 4.8: Die Zuweisung von Callback-Funktionen und Events. Um eine Verwechslung mit den Events zu vermeiden, wurden die Namen der Callback-Funktionen abgekürzt.

Die Callback-Funktionen definieren

Nachdem nun festgelegt wurde, dass bestimmte Funktionen zu bestimmten Ereignissen aufgerufen werden sollen, müssen diese Funktionen jetzt allerdings noch programmiert werden.

```
function startE ($parser, $name, $attribs)
{
    if($name=="ROW")
    {
        echo "<tr>";
    }
    else if($name=="TITEL" || $name=="INTERPRET" ||
           $name=="JAHR" || $name=="ID")
    {
        echo "<td>";
    }
}
```

Die Funktion `startE()` ist die Callback-Funktion für das `startElement`-Event. Sie wird also aufgerufen, sobald der XML-Parser ein sich öffnendes XML-Tag findet. Im zweiten Parameter `$name` wird der Funktion der Name des sich öffnenden XML-Tags mitgeteilt. Hat das XML-Tag zusätzliche Attribute, werden diese im dritten Parameter `$attribs` an die Callback-Funktion übergeben.

In der Callback-Funktion wird nun anhand der Variablen `$name` unterschieden, welcher HTML-Code ausgegeben werden soll. Beim XML-Tag `ROW` wird `<tr>` ausgegeben und bei den XML-Tags `TITEL`, `INTERPRET`, `JAHR` und `ID` `<td>`.

Die folgende Callback-Funktion für das *endElement*-Event macht genau das Gleiche, nur umgekehrt.

```
function endE ($parser, $name)
{
    if($name=="ROW")
    {
        echo "</tr>";
    }
    else if($name=="TITEL" || $name=="INTERPRET" ||
           $name=="JAHR" || $name=="ID")
    {
        echo "</td>";
    }
}
```

Findet der XML-Parser ein sich schließendes Tag, wird je nach Tag-Name entweder ein `</tr>` oder ein `</td>` ausgegeben.

Als Nächstes wird die Callback-Funktion für das *characterData*-Event definiert:

```
function characterD ($parser, $data)
{
    print $data;
}
```

Die `characterD()`-Funktion interpretiert sich ganz einfach: Trifft der XML-Parser auf irgendwelche Daten in dem XML-Dokument, dann werden sie einfach von dieser Funktion ausgegeben.

Im nächsten Schritt wird nun die XML-Beispieldatei gelesen und an den XML-Parser übergeben:

```
$fp = fopen('cds.xml', 'r');
while ($data = fread($fp, 1024))
{
    $result = xml_parse($parser, $data);
}
fclose($fp);
```

Das Ergebnis

Der XML-Parser geht nun das XML-Dokument Element für Element durch und ruft die Callback-Funktionen zu den bestimmten Events auf. Heraus kommt dann Folgendes:



Abbildung 4.21: Die Ausgabe von `sax.php`

Hier noch einmal die gesamte `sax.php` im Überblick:

Listing 4.2: Die komplette `sax.php`

```
<html>
<head>
<title>PHP und SAX</title>
</head>
<body>
<h1>PHP und SAX</h1>
<table border="1">
<tr>
    <th>Titel</th>
    <th>Interpret</th>
    <th>Jahr</th>
    <th>ID</th>
</tr>
<?php
    $parser = xml_parser_create();
    xml_set_element_handler($parser, 'startE','endE');
    xml_set_character_data_handler($parser, 'characterD');

    function startE($parser, $name, $attrs)
    {
        if($name=="ROW")
        {
            echo "<tr>";
        }
        else if($name=="TITEL" || $name=="INTERPRET" ||
            $name=="JAHR" || $name=="ID")
        {
```

```

        echo "<td>";
    }
}

function endE ($parser, $name)
{
    if($name=="ROW")
    {
        echo "</tr>";
    }
    else if($name=="TITEL" || $name=="INTERPRET" ||
           $name=="JAHR" || $name=="ID")
    {
        echo "</td>";
    }
}

function characterD ($parser, $data)
{
    print $data;
}

$fp = fopen('cds.xml', 'r');
while ($data = fread($fp, 1024))
{
    $result = xml_parse($parser, $data);
}
fclose($fp);
?>
</table>
</body>
</html>

```

Die Datei `sax.php` finden Sie im XAMPP für Linux unter `/opt/lampp/htdocs/xampp/contrib/sax.php` oder im XAMPP für Windows unter `<XAMPP>\htdocs\xampp\contrib\sax.php`.

4.3.3 DOM

Wie schon in der Einleitung zu diesem Abschnitt erwähnt, ist einer der großen Vorteile von DOM gegenüber SAX die Möglichkeit, auch XML-Dokumente zu erstellen. Auf den folgenden Seiten werden wir uns ein Beispiel aus XAMPP ansehen, in dem DOM verwendet wird, um aus einer MySQL-Datenbank ein XML-Dokument zu erstellen.

Das Beispielskript befindet sich in der Linux-Version von XAMPP unter `/opt/lampp/htdocs/xampp/contrib/dom.php` und in der Windows-Version im XAMPP-Installationsverzeichnis unter `htdocs\xampp\contrib\dom.php`.

Achtung

Dieses Beispiel funktioniert erst ab PHP 5.



DOM-Dokument einrichten

Ganz am Anfang des Programms wird zunächst ein Objekt der Klasse `DOMDocument` erzeugt.

```
$dom = new DOMDocument('1.0', 'iso-8859-1');
```

Mit den beiden Parametern `'1.0'` und `'iso-8859-1'` werden die XML-Version und das *character encoding* (Zeichentabelle) des XML-Dokuments gesetzt. In diesem Fall wird das von DOM generierte XML-Dokument also mit folgender Zeile beginnen:

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

Im nächsten Schritt wird dem XML-Dokument das *root*-Element hinzugefügt.

```
$root = $dom->createElement('cds');  
$dom->appendChild($root);
```

Mit der Methode `createElement()` wird ein XML-Element mit dem Namen `cds` erzeugt und dann mit `appendChild()` dem XML-Dokument hinzugefügt. Damit hat unser XML-Dokument sein *root*-Element erhalten.

Das XML-Dokument würde bisher so aussehen:

```
<?xml version="1.0" encoding="iso-8859-1"?>  
<cds/>
```

Verbindung zur Datenbank herstellen

Da das XML-Dokument mit Daten aus einer MySQL-Datenbank gefüllt wird soll, wird nun die Verbindung zur Datenbank hergestellt und die Datenbank `cdcol` ausgewählt.

```
mysql_connect("localhost","root","");  
mysql_select_db("cdcol");
```

Danach wird mit `mysql_query()` eine SQL-Abfrage an die Datenbank geschickt. Diese Abfrage haben wir schon sehr oft im Datenbankabschnitt dieses Kapitels gesehen.

```
$result=mysql_query("SELECT id,titel,interpret,jahr FROM cds  
ORDER BY interpret;");
```


Den DOM-Objektbaum aufbauen

Nun folgt das Herz des kleinen DOM-Programms: In einer `while()`-Schleife werden die Ergebnisse der SQL-Abfrage durchgegangen und für jede einzelne Zeile ein Objektbäumchen erstellt. Am Ende eines Schleifendurchlaufs wird dann dieses Objektbäumchen zum großen Objektbaum des XML-Dokuments hinzugefügt.

```
while( $row=mysql_fetch_array($result) )
{
    $cd = $dom->createElement('cd');
    $cd->setAttribute('id', $row['id']);
```

Mit `createElement()` wird ein XML-Element namens `cd` erstellt und danach wird diesem Element das Attribut `id` mit dem Wert aus der Variablen `$row['id']` zugewiesen.

In XML könnte das Ergebnis etwa so aussehen:

```
<cd id="5"/>
```

Als Nächstes wird ein XML-Element mit dem Namen `titel` erstellt und um einen Textknoten mit dem Wert aus der Variablen `$row['titel']` erweitert.

```
$titel = $dom->createElement('titel');
$titel->appendChild($dom->createTextNode($row['titel']));
```

In XML könnte das Ergebnis der letzten beiden Zeilen etwa so aussehen:

```
<titel>Glee</titel>
```

Nun wird dieses XML-Element dem `cd`-Element hinzugefügt:

```
$cd->appendChild($titel);
```

Das Ergebnis sollte dann so aussehen:

```
<cd id="5">
    <titel>Glee</titel>
</cd>
```

Im nächsten Schritt der `while()`-Schleife passiert genau das Gleiche mit dem `interpret`-Element:

```
$interpret = $dom->createElement('interpret');
$interpret->appendChild(
    $dom->createTextNode($row['interpret']));
$cd->appendChild($interpret);
```

Das XML sollte danach so aussehen:

```
<cd id="5">
    <titel>Glee</titel>
    <interpret>Bran Van 3000</interpret>
</cd>
```

Danach wird noch das `jahr`-Element hinzugefügt.

```
$jahr = $dom->createElement('jahr');  
$jahr->appendChild($dom->createTextNode($row['jahr']));  
$cd->appendChild($jahr);
```

Dies erzeugt nun folgendes XML:

```
<cd id="5">  
  <titel>Glee</titel>  
  <interpret>Bran Van 3000</interpret>  
  <jahr>1997</jahr>  
</cd>
```

Am Ende der Schleife wird dann dieser Minibaum genommen und dem `root`-Element des XML-Dokuments hinzugefügt.

```
$root->appendChild($cd);  
}
```

Die Schleife wiederholt genau diese Schritte jetzt für jede Zeile aus der Datenbank und füllt so das XML-Dokument mit den Daten. Sind alle Einträge der Datenbank hinzugefügt, wird das XML-Dokument generiert und ausgegeben:

```
header("Content-Type: text/xml;");  
print $dom->saveXML();
```

Das komplette Ergebnis kann man im folgenden Screenshot sehen:

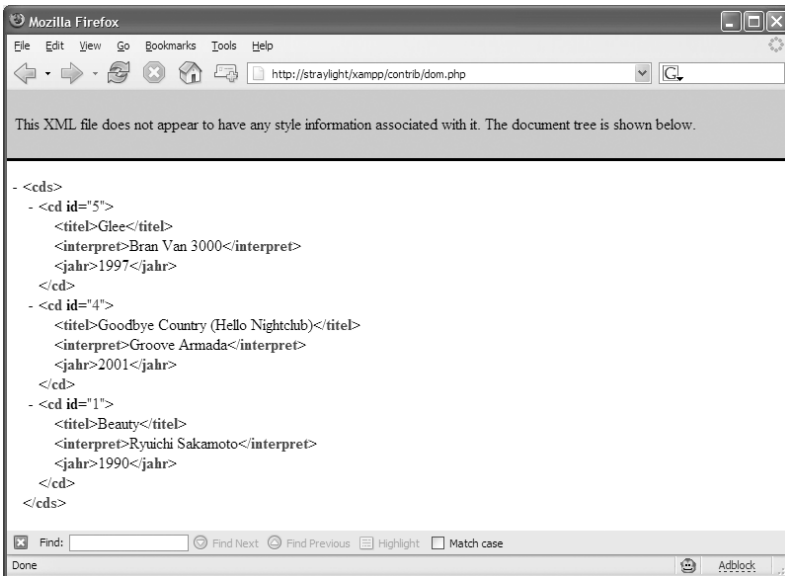


Abbildung 4.22: Das Ergebnis von `dom.php`

Listing 4.3: Das komplette dom.php-Beispiel

```

<?php

$dom = new DOMDocument('1.0', 'iso-8859-1');

$root = $dom->createElement('cds');
$dom->appendChild($root);

mysql_connect("localhost","root","");
mysql_select_db("cdcol");

$result=mysql_query("SELECT id,titel,interpret,jahr FROM cds
                    ORDER BY interpret;");

while( $row=mysql_fetch_array($result) )
{
    $cd = $dom->createElement('cd');
    $cd->setAttribute('id', $row['id']);

    $titel = $dom->createElement('titel');
    $titel->appendChild($dom->createTextNode($row['titel']));
    $cd->appendChild($titel);

    $interpret = $dom->createElement('interpret');
    $interpret->appendChild(
        $dom->createTextNode($row['interpret']));
    $cd->appendChild($interpret);

    $jahr = $dom->createElement('jahr');
    $jahr->appendChild($dom->createTextNode($row['jahr']));
    $cd->appendChild($jahr);

    $root->appendChild($cd);
}

header("Content-Type: text/xml;");
print $dom->saveXML();
?>

```

4.3.4 XML und RSS

RSS (ausgeschrieben meist *Really Simple Syndication*) ist ein auf XML basierendes Datenaustauschformat, mit dem man zum Beispiel auf sehr einfache Weise Daten von der einen Website in eine andere integrieren kann. Leider gibt es mindestens drei verschiedene Formate, die für RSS verwendet werden können und jeweils auch ihre eigene Interpretation der Abkürzung liefern:

- Rich Site Summary (RSS 0.91)
- RDF Site Summary (RSS 0.9 und 1.0)
- Really Simple Syndication (RSS 2.0)

Im folgenden Beispiel wird die PEAR-Bibliothek XML_RSS verwendet, um auf den RSS-Feed – so wird die RSS-Quelle genannt – der Heise-News zuzugreifen.

Das Beispielskript finden Sie in der Linux-Version von XAMPP unter `/opt/lampp/htdocs/xampp/contrib/xmlrss.php` und in der Windows-Version im XAMPP-Installationsverzeichnis unter `htdocs\xampp\contrib\xmlrss.php`.

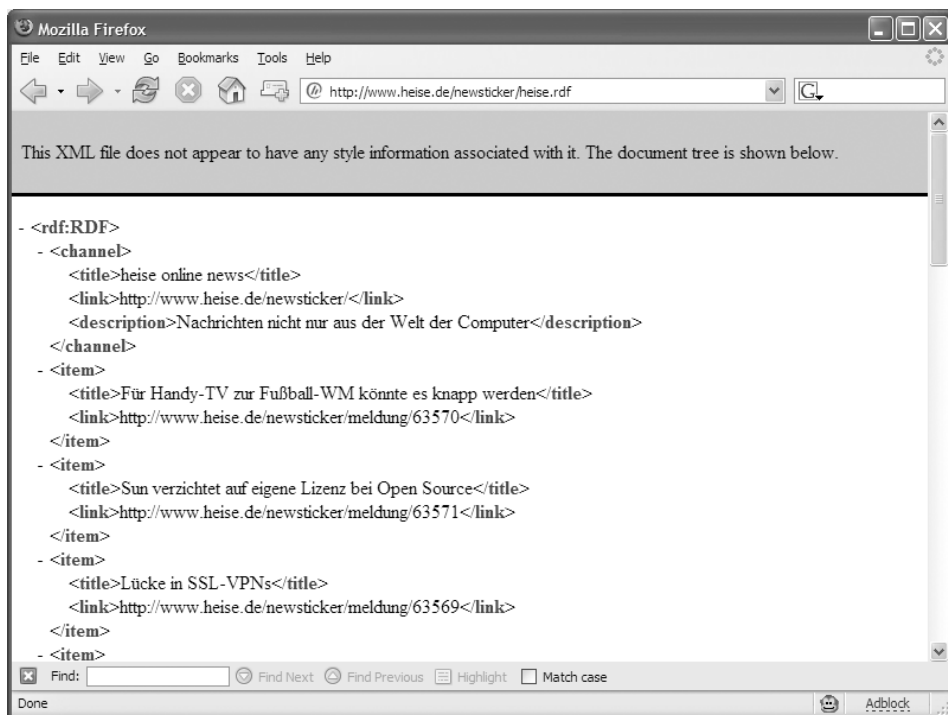


Abbildung 4.23: So sieht ein RSS-Feed aus, wenn man ihn im Browser betrachtet. Man erkennt deutlich die XML-Struktur.

Das Skript fängt mit einer untypischen Zeile an:

```
<?php
    header("Content-Type: text/html; charset=utf-8;");
?>
```

Mit diesem Header-Befehl wird dem Browser angezeigt, dass diese HTML-Seite in UTF-8 kodiert sein wird. Dies ist notwendig, da der Heise-News-Feed – wie eigentlich fast jeder RSS-Feed – selbst in UTF-8 kodiert ist und ein Browser die Daten nor-

malerweise noch in ISO 8859-1 (auch Latin1 genannt) erwartet. Latin1 war früher, als das WWW noch von Europa und Nordamerika dominiert war, eine gute Wahl. Heute, wo die Präsenz asiatischer Länder immer stärker wird, ist UTF-8 die bessere Wahl und setzt sich auch immer mehr durch.

Es folgt der übliche HTML-Header:

```
<html>
<head>
<title>PHP und XML_RSS</title>
</head>
<body><?php
```

Als nächste Aktion wird die PEAR-XML_RSS-Klasse geladen.

```
include_once("XML/RSS.php");
```

Nun wird das neue Objekt `$rss` von der XML_RSS-Klasse instanziiert. Im Konstruktor dieser Klasse wird die URL des Feeds übergeben. In diesem Fall ist es die URL des Heise-Tickers:

```
$rss = new XML_RSS('http://www.heise.de/newsticker/heise.rdf');
```

Schon gleich beim Erzeugen des Objekts wird eine Verbindung zum RSS-Feed aufgebaut. Gibt es die URL nicht, dann wird jetzt schon ein Fehler auftreten.

Mit der Methode `parse()` werden im nächsten Schritt die Daten des Feeds gelesen und parsiert.

```
$rss->parse();
```

Die Methode `getChannelInfo()` zieht nun Metainformationen aus den Feed-Daten, die danach als Kopf in der HTML-Seite ausgegeben werden.

```
$meta=$rss->getChannelInfo();
echo "<h1>{"$meta['title']}</h1>";
echo $meta['description'];
echo "<br>";
echo "<a href=\"{"$meta['link']}\">mehr...</a>";
echo "<p>";
```

In der folgenden Schleife werden nun alle Einträge (in diesem Fall News-Beiträge) des RSS-Feeds durchgegangen und einer nach dem anderen wird ausgegeben.

```
foreach ($rss->getItems() as $item)
{
    echo "<a href=\"{"$item['link']}\">{"$item['title']}</a><br>";
}
```

Zum Schluss noch das obligatorische Ende der HTML-Seite:

```
?>
</body>
</html>
```

Und das war's auch schon.

Um einen besseren Überblick zu bekommen, wird im Folgenden das vollständige Skript aufgelistet.

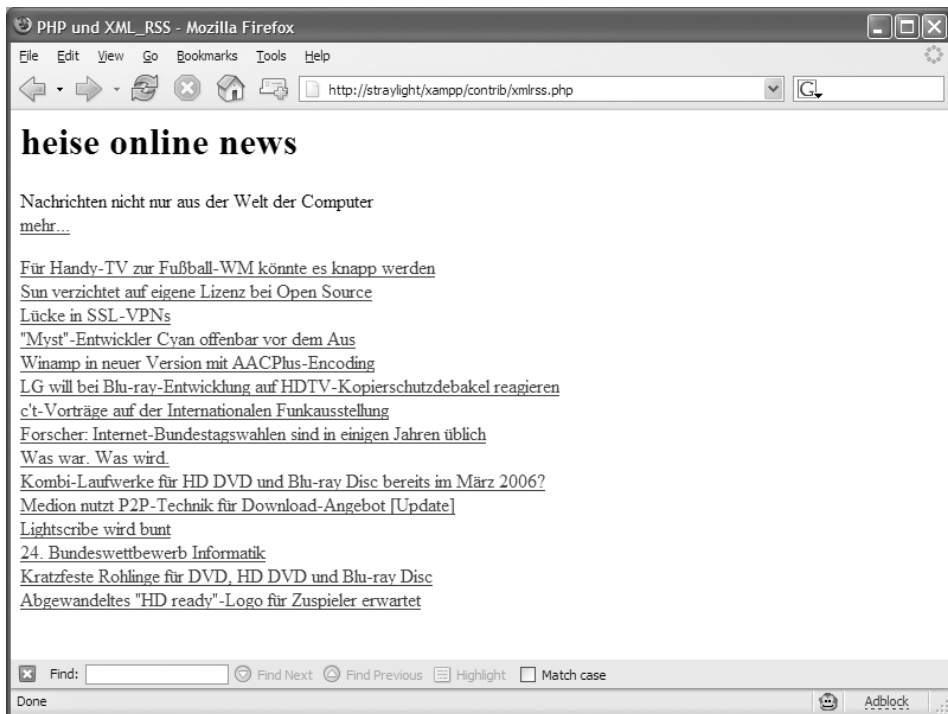


Abbildung 4.24: Mit nur wenigen Zeilen bringt man die Heise-News in die eigene Website

Listing 4.4: Hier noch einmal das gesamte Listing von *xml/rss.php*.

```
<?php
    header("Content-Type: text/html; charset=utf-8;");
?>
<html>
<head>
<title>PHP und XML_RSS</title>
</head>
<body><?php
```

```

include_once("XML/RSS.php");

$rss = new XML_RSS('http://www.heise.de/newsticker/heise.rdf');

$rss->parse();

$meta=$rss->getChannelInfo();

echo "<h1>{$meta['title']}</h1>";
echo $meta['description'];
echo "<br>";
echo "<a href=\"{$meta['link']}\">mehr...</a>";
echo "<p>";

foreach ($rss->getItems() as $item)
{
    echo "<a href=\"{$item['link']}\">{$item['title']}</a><br>";
}

?>
</body>
</html>

```

4.3.5 XML_sql2xml

Ob man sie wirklich braucht, sei dahingestellt, aber es ist einfach eine wunderschöne PEAR-Klasse und ihre Verwendung ist so herrlich leicht: Mit `XML_sql2xml` kann man ohne weiteres aus einer SQL-Anfrage XML machen.

Das folgende Beispiel finden Sie wie immer im Verzeichnis `contrib` unter `htdocs/xampp` bzw. `htdocs\xampp`. Es trägt den Dateinamen `sql2xml.php`.

Ganz am Anfang wird die `sql2xml`-Klasse geladen.

```

<?php
    include_once("XML/sql2xml.php");

```

Danach wird die Verbindung zur Datenbank aufgebaut. `sql2xml` benutzt die PEAR DB-Klasse als Abstraktionsschicht, um auf die Datenbank zuzugreifen. D. h., Sie können hier auf alle Datenbanken zugreifen, die auch PHP bzw. PEAR DB unterstützen.

```

$sql2xml = new xml_sql2xml("mysql://root@localhost/cdcol");

```

In diesem Beispiel wird davon ausgegangen, dass der Benutzer *root* kein Passwort hat. Sollten Sie dem Benutzer bereits ein Passwort gesetzt haben, dann geben Sie es durch Doppelpunkt getrennt nach dem Benutzernamen an. Zum Beispiel:

```

mysql://root:geheim@localhost/cdcol

```

Im nächsten Schritt wird die SQL-Anfrage an die Datenbank abgeschickt und das Ergebnis der Anfrage im XML-Format in der Variablen `$xml` gespeichert.

```
$xml = $sql2xml->getxml ("select * from cds");
```

Danach wird der HTTP-Header auf XML gesetzt und die Variable `$xml` ausgegeben.

```
header("Content-Type: text/xml;");
echo $xml;
```

```
?>
```

Das war's. Das erstaunliche Ergebnis dieses Fünfzeilers sehen Sie im nächsten Screen-shot.

Hier noch einmal das gesamte Listing:

Listing 4.5: Noch einfacher geht es nicht: `sql2xml.php`.

```
<?php
include_once("XML/sql2xml.php");

$sql2xml = new xml_sql2xml ("mysql://root@localhost/cdcol");
$xml = $sql2xml->getxml ("select * from cds");

header("Content-Type: text/xml;");
echo $xml;

?>
```

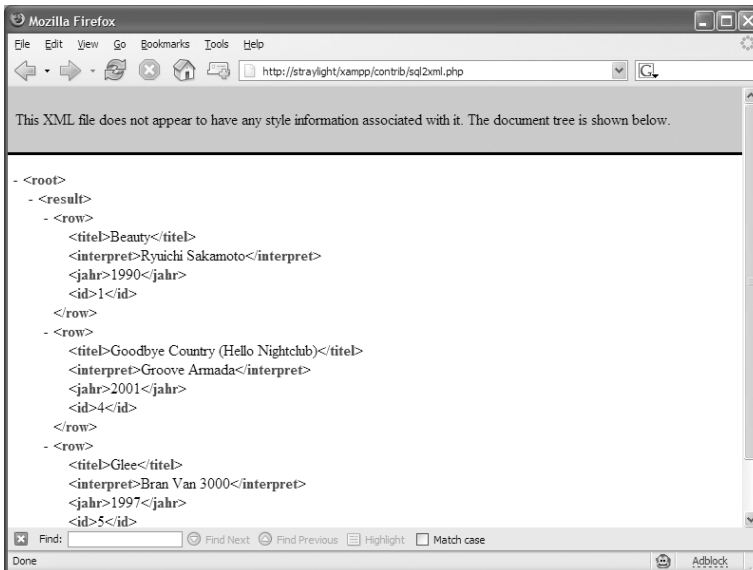


Abbildung 4.25: XML fast direkt aus einer SQL-Datenbank

4.3.6 SOAP

SOAP ist ein Protokoll zum Austausch von Daten zwischen zwei Systemen und zum Ausführen von Aktionen auf einem entfernten System. Ganz einfach erklärt: SOAP ist eine Mischung aus RSS und RPC.

Ursprünglich war SOAP die Abkürzung für Simple Object Access Protocol. Da SOAP aber weder einfach noch ausschließlich dem Zugriff auf Objekte dient, wurde diese ursprünglich ausgeschriebene Bezeichnung fallen gelassen und nur die Abkürzung hat überlebt. Heute muss man also sagen: SOAP bedeutet nichts.

Im Beispiel dieses Abschnitts wird die PHP-SOAP-Unterstützung verwendet, um auf die SOAP-API der Suchmaschine von Google zuzugreifen.

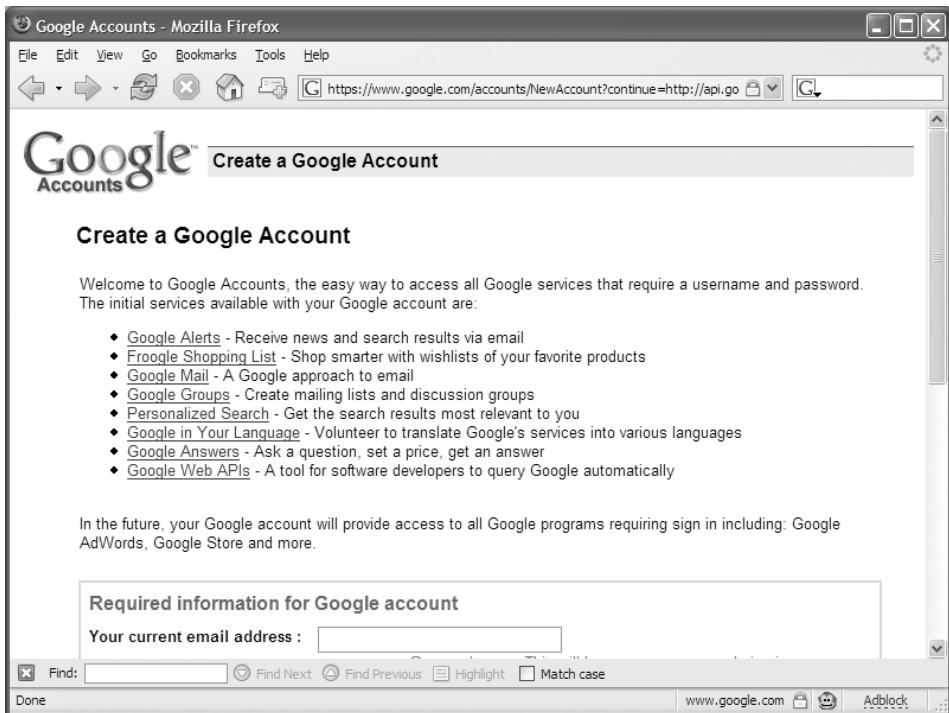


Abbildung 4.26: Bevor man mit SOAP auf die Google-API zugreifen kann, muss man sich einen API-Key bei Google besorgen.

Damit das Beispielprogramm allerdings funktioniert, sind zwei Voraussetzungen zu erfüllen:

1. Sie benötigen einen sog. API-Key von Google, um die SOAP-API der Suchmaschine verwenden zu können. Diesen Key können Sie sich kostenlos unter <http://www.google.com/apis/> beschaffen.

2. Die SOAP-Unterstützung gibt es nur unter PHP 5. D. .h., wenn Sie XAMPP auf PHP 4 geschaltet haben, müssen sie es erst wieder auf PHP 5 umschalten, um dieses Programm laufen lassen zu können.

Das Beispielprogramm finden Sie unter dem Namen `soap.php` im Verzeichnis `htdocs/xampp/contrib` bzw. `htdocs\xampp\contrib`.

Als Erstes wird dem Browser übermittelt, dass die Daten, die er zu erwarten hat, UTF-8-kodiert sein werden.

```
<?php
    header("Content-Type: text/html; charset=utf-8;");
?>
```

Es folgt der übliche HTML-Header:

```
<html>
<head>
<title>PHP und SOAP</title>
</head>
<body>
<?php
```

Nun werden die Vorbereitungen für den SOAP-Zugriff durchgeführt. Im ersten Schritt wird das Objekt `$client` erzeugt, über das später der Zugriff stattfinden wird:

```
$wsdlurl='http://api.google.com/GoogleSearch.wsdl';
$client = new SoapClient($wsdlurl);
```

Im nächsten Schritt werden der API-Key und das Beispielsuchwort für die Google-Suche gesetzt:

```
$apikey="apikey hier einsetzen";
$searchfor="xampp";
```

Geben Sie hier keinen gültigen API-Key an, erhalten Sie später die folgende Fehlermeldung:

Fatal error: *Uncaught SoapFault exception: [SOAP-ENV:Server] Exception from service object: Invalid authorization key: apikey hier einsetzen in .../xampp/contrib/soap.php:16*
Stack trace: #0

Nun findet der eigentliche SOAP-Zugriff statt: Auf dem Server von Google wird die Methode `doGoogleSearch()` aufgerufen:

```
$result = $client->doGoogleSearch($apikey, $searchfor, 0, 10, false, '', true, '
'', '', '');
```

Danach wird die Anzahl der gefundenen Seiten ausgegeben.

```
echo "Gefundene Seiten: ".$result->estimatedTotalResultsCount;
```

Innerhalb einer `foreach()`-Schleife werden die Ergebnisse der Suchanfrage in einer nummerierten Liste ausgegeben:

```
echo "<ol>";
foreach ($result->resultElements as $hit)
{
    echo "<li><a href=\"". $hit->URL. "\">". $hit->title. "</a>";
    echo "<br>". $hit->snippet;
}
echo "</ol>";
?>
```

Jetzt werden noch die notwendigen Ende-Tags der HTML-Seite gesetzt:

```
</body>
</html>
```

Das war's. Über die URL *http://localhost/xampp/contrib/soap.php* können Sie nun den SOAP-Zugriff testen.

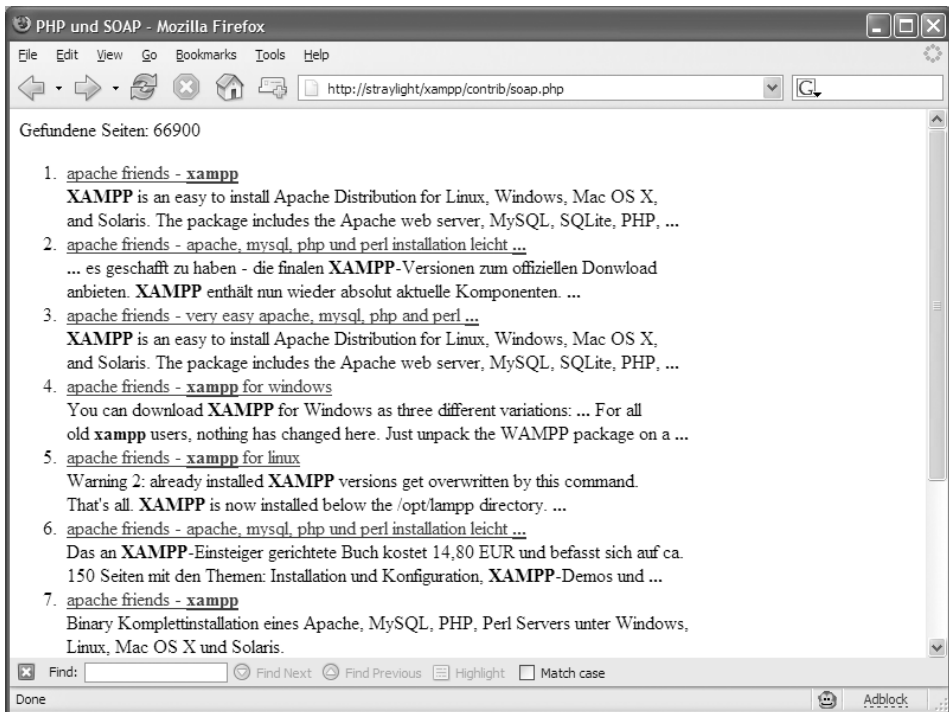


Abbildung 4.27: soap.php in Aktion

Hier noch einmal das vollständige SOAP-Beispiel im Überblick:

Listing 4.6: Das soap.php-Skript für den Zugriff auf Google via SOAP

```
<?php
    header("Content-Type: text/html; charset=utf-8;");
?>
<html>
<head>
<title>PHP und SOAP</title>
</head>
<body>
<?php

    $wsdlurl='http://api.google.com/GoogleSearch.wsdl';

    $client = new SoapClient($wsdlurl);

    $apikey="apikey hier einsetzen";
    $searchfor="xampp";

    $result = $client->doGoogleSearch($apikey, $searchfor, 0, 10,
                                      false, '', true, '', '', '');

    echo "Gefundene Seiten: ".$result->estimatedTotalResultsCount;

    echo "<ol>";
    foreach ($result->resultElements as $hit)
    {
        echo "<li><a href=\"".$hit->URL."">".$hit->title."</a>";
        echo "<br>".$hit->snippet;
    }
    echo "</ol>";
?>
</body>
</html>
```

4.4 PHP und Flash

Auch wenn man es auf den ersten Blick vielleicht nicht so sieht, so haben PHP und Flash doch so einiges gemeinsam.

Zum einen kann man tatsächlich Flash-Filme mit PHP erstellen und zum anderen – und das dürfte sicherlich die größere Verbindung zwischen den beiden sein – bildet PHP sehr gerne oder sehr oft das Backend für Flash-Frontends.

Auf den folgenden Seiten werden wir uns drei verschiedene Anwendungen mit PHP und Flash anschauen:

Als Erstes werfen wir einen Blick in den Code des Flash Art-Beispiels von XAMPP: Wir beschäftigen uns also mit dem Thema *Animationen*.

Danach wenden wir uns einem in PHP/Flash programmierten und auch animierten Balkendiagramm zu – eine etwas ernstere Anwendung als das einfach nur hübsche *Flash Art*.

Und zuletzt beschäftigen wir uns damit, wie man ganz geschickt Daten von PHP zu Flash übertragen kann.

Die Ming-Library

Die Ming-Library von Dave Hayden ist das Herzstück der Flash- oder SWF-Unterstützung von PHP. Im Internet ist sie unter <http://ming.sourceforge.net/> zu finden. Dort findet sich auch die einzige (größere) Anleitung zu Ming. Leider ist sie – wie so oft bei Open-Source-Projekten – nicht sehr überzeugend.

Obwohl die Ming-Unterstützung in PHP zum Core-Paket gehört, hat diese Erweiterung leider noch keine große Bekanntheit erlangt und es gibt bisher nur wenige Hoster, die sie in ihrem Leistungspaket mit anbieten.

In XAMPP ist sie schon seit den ersten Versionen enthalten, vor allem deswegen, weil es einfach eine wirklich coole Sache ist, mit PHP on-the-fly Flash-Filme zu erzeugen.



Hinweis

MING oder Ming? Letzteres ist definitiv richtig. So steht dazu extra auf der Homepage von Ming: *Ming is not an acronym*. Was aber Ming genau bedeutet, darüber schweigt sich die Homepage aus. Lediglich in den FAQ gibt es einen Hinweis auf die Bedeutung:

Aren't Ming and Flash enemies?

Maybe, I dunno. I liked the name. Maybe there's a homoerotic subtext there. (Actually, all I remember from the movie was somebody putting their hand in a rock and green blood. And that awful, awful song.) If that bothers you, just think of vases and ancient China.

Mit Ming kann also nur Kaiser Ming, der Widersacher von Flash Gordon, gemeint sein – was auch eine schöne versteckte Aussage über die Beziehung zwischen Ming und Flash ist.

Quelle: <http://www.opaque.net/wiki/index.php?MingFAQ>

4.4.1 Flash Art

Flash Art ist der – nicht wirklich originelle, aber auch nicht ernst gemeinte – Name des PHP/Ming-Demos in XAMPP. In der Animation drehen sich vier Textzeilen mal im und mal entgegen dem Uhrzeigersinn, mal schneller, mal langsamer und mal ein wenig mehr rechts und mal ein wenig mehr links.

Zum Beweis, dass es sich um einen dynamisch erzeugten Flash-Film handelt, kann über ein kleines HTML-Formular der Text in der Animation geändert werden.



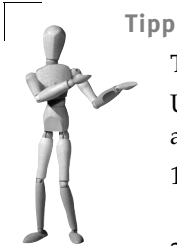
Abbildung 4.28: Leider kann man die Bewegung der Animation hier nicht wirklich erkennen. Die Schriftzeilen drehen sich mal mit und mal entgegen dem Uhrzeigersinn, wobei die dunkleren Zeilen den helleren folgen ...

Auch wenn es nur eine Animation ist, eignet sich dieses Beispiel doch sehr gut, um die Funktionsweise von Ming zu veranschaulichen. Es gibt sicherlich bessere Ming-Programme, aber die sind dann meist gleich so komplex, dass sie als Beispiel ungeeignet sind.

Richten Sie Ihre Aufmerksamkeit auf den Sourcecode von `/opt/lampp/htdocs/xampp/mingswf.php`:

```
$f = new SWFFont("AnkeCalligraph.fdb");
```

Mit dieser Anweisung wird ein neues Objekt erzeugt, das die im Konstruktor angegebene Schrift repräsentiert. Im Parameter wird auf eine Font-Datei im FDB(*Font Definition Block*)-Format verwiesen. Dieses Format ist recht unüblich und selten. Allerdings ist es möglich, TrueType-Fonts in dieses Format umzuwandeln. Mehr dazu im folgenden Kasten.



Tipp

TrueType nach FDB

Um einen TrueType-Font ins FDB-Format umzuwandeln, gehen Sie am besten so vor:

1. Zunächst wandeln Sie mit `ttf2fft` den Font in das FFT-Format um. Siehe dazu: <http://www.opaque.net/wiki/index.php?ttf2fft>.
2. Danach können Sie mit `makefdb` den FFT-Font ins FDB-Format umwandeln. Siehe dazu: <http://www.opaque.net/wiki/index.php?makefdb>.

Im nächsten Schritt wird der Flash-Film erzeugt bzw. das Objekt, das den Flash-Film repräsentiert. Vielleicht erinnern Sie sich an den Abschnitt zu PHP/PDF. Dort ging es auch immer mit einem vergleichbaren Schritt los.

```
$m = new SWFMovie();
$m->setRate(24.0);
$m->setDimension(520, 320);
$m->setBackground(251, 121, 34);
```

In der ersten Zeile wird das Flash-Objekt angelegt und dann werden über Methodenaufrufe folgende Merkmale des zukünftigen Flash-Films festgelegt:

- Mit `setRate()` wird die Bildfrequenz bestimmt. Je höher dieser Wert, desto weicher sehen die Animationen aus, desto mehr Last erzeugen sie allerdings auch und viele Rechner kommen hier mit großen Werten nicht klar. Der Wert 24 ist ein typischer und eigentlich recht brauchbarer Wert.
- `setDimension()` legt die Größe des Flash-Films fest. Achtung: Wird Flash in eine HTML-Seite eingebunden, dann muss dort ebenfalls die Größe des Flash-Films angegeben werden. Die HTML-Größe hat Vorrang gegenüber der Angabe hier. Generell und um optimale Ergebnisse zu erzeugen, sollten aber beide Angaben identisch sein.
- Mit `setBackground()` wird – und das überrascht nicht – die Hintergrundfarbe des Flash-Films gesetzt. Die Methode bekommt drei Parameter für die Farben Rot, Grün und Blau, jeweils im typischen Zahlenbereich von 0 bis 255.

Als Nächstes wird eine kleine Hilfsfunktion definiert, die das Ausgeben (oder eher Zeichnen) von Textzeilen erleichtern soll.

```
function text($r, $g, $b, $a, $rot, $x, $y, $scale, $string)
{
```

Die Funktion `text()` verlangt neun Parameter:

- Rot-, Grün- und Blauanteil der Schriftfarbe (jeweils im Bereich 0 bis 255),
- den Alpha-Wert (im Bereich von 0 bis 100, wobei 0 ganz durchsichtig und 100 undurchsichtig ist),
- den Winkel, in dem der Text dargestellt werden soll (im Bereich von 0 bis 360, wobei 0 quasi waagrecht bedeutet, 180 auf dem Kopf und 360 wieder waagrecht),
- die x- und y-Position des Textes,
- die Größe des Textes,
- und – last, but not least – den Text selbst.

Als Nächstes werden die Variablen `$f` (für den Font) und `$m` für das Film-Objekt als globale Variablen deklariert. (Sonst hätten auch noch diese beiden Variablen als Parameter mit übergeben werden müssen.)

```
global $f, $m;
```

Der Text in einem MingFlash-Film wird durch ein Objekt von der Klasse `SWFText` repräsentiert. Im nächsten Schritt wird also ein solches Objekt erzeugt:

```
$t = new SWFText();  
$t->setFont($f);  
$t->setColor($r, $g, $b, $a);  
$t->setHeight(96);
```

Dem Text-Objekt wird nach der Erzeugung eine Schrift zugewiesen (die Schrift, die ganz am Anfang des Programms mit `SWFFont()` geladen wurde), die Schriftfarbe wird gesetzt und die Schriftgröße festgelegt.

Hinweis



Die Schriftgröße ist hier zwar fest auf 96 eingestellt, da später aber der gesamte Text noch in Abhängigkeit von `$scale` skaliert wird, entstehen unterschiedlich große Texte, auch wenn die Schriftgröße eigentlich bei allen Texten gleich ist.

Mit der folgenden Anweisung wird der Text zentriert:

```
$t->moveTo(-($t->getWidth($string))/2, 32);
```

Mit `getWidth($string)` wird die Breite des Textes errechnet und dann wird der Text um die Hälfte der Breite nach links verschoben.

Danach wird endlich der Text ausgegeben:

```
$t->addString($string);
```

Oder genauer: Er wurde nicht ausgegeben, sondern dem Text-Objekt hinzugefügt. Denn das Ausgeben erfolgt eigentlich eher im folgenden Schritt:

```
$i = $m->add($t);
```

Mit der `add()`-Methode wird das Text-Objekt dem Film-Objekt hinzugefügt. Jetzt ist der Text quasi im Film. In der Variablen `$i` wird eine Referenz auf dieses neue Objekt im Film gespeichert.

```
$i->rotateTo($rot);
$i->moveTo($x,$y);
$i->scale($scale, $scale);
```

Der Text wird nun in die gewünschte Lage rotiert und mit `moveTo()` an die gewünschte Position gebracht. Danach erhält er noch mit `scale()` die gewünschte Größe.

Am Ende der `text()`-Funktion wird die Referenz auf das Text-Objekt in dem Flash-Film als Rückgabewert an die aufrufende Stelle zurückgegeben.

```
    return $i;
}
```

In der folgenden Kolonne werden die Grautöne für die vier Textzeilen auf recht umständliche Art und Weise gesetzt.

```
$colorr[1]=255*0.85;
$colorg[1]=255*0.85;
$colorb[1]=255*0.85;

$colorr[2]=255*0.9;
$colorg[2]=255*0.9;
$colorb[2]=255*0.9;

$colorr[3]=255*0.95;
$colorg[3]=255*0.95;
$colorb[3]=255*0.95;

$colorr[4]=255;
$colorg[4]=255;
$colorb[4]=255;
```

Im nächsten Abschnitt werden die vier Textzeilen an einer beliebigen Position (0,0 = oben rechts) mit unterschiedlichen Größen und Farben platziert. Die Text-Objekte (bzw. ganz genau: Referenzen auf die Objekte) werden in dem Array `$t[]` gespeichert.

```

$c=1;
$anz=4;
$step=4/$anz;

for($i=0;$i<$anz;$i+=1)
{
    $x=0;
    $y=0;
    $size=($i/5+0.2);
    $t[$i]=text($colorr[$c],$colorg[$c],$colorb[$c],
                0xff,0,$x,$y, $size,$text);
    $c+=$step;
}

```

Zunächst eine kleine Wiederholung der bisherigen Vorgehensweise:

1. Als Erstes wird mit `SWFMovie()` ein Film-Objekt erzeugt.
2. Danach werden mit `SWFText()` einzelne Text-Objekte erzeugt, die dann mittels der `add()`-Methode des Film-Objekts dem Film hinzugefügt werden.
3. Referenzen auf die Text-Objekte werden dabei in dem Array `$t[]` gespeichert.

Jetzt kommt der wichtigste Teil des Programms:

```

$frames=300;
for($j=0; $j<$frames; $j++)
{
    for($i=0; $i<$anz; $i++)
    {
        $t[$i]->moveTo(
            260+round(sin($j/$frames*2*pi()+$k)
                *(50+50*(($i+1)))),
            160+round(sin($j/$frames*4*pi()+$i)
                *(20+20*(($i+1))))
        );
        $t[$i]->rotateTo(round(sin($j/$frames*2*pi()+$i/10)*360));
    }
    $m->nextFrame();
}

```

Hierbei passiert Folgendes:

- In der äußeren Schleife werden die 300 Einzelbilder (*frames*) des Flash-Films erzeugt. Genauso wie ein richtiger Film im Kino, Fernsehen oder auf der Festplatte besteht ein Flash-Film auch aus lauter Bildern, die nacheinander ablaufen. Dieser Film hat 300 von solchen Bildern und diese werden in der äußeren Schleife erstellt.
- Immer dann, wenn in der äußeren Schleife das Ende erreicht wird, wird über `$m->nextFrame()`; der Film um ein Bild weitergeschaltet. Um in der Filmmetapher zu bleiben: Mit `nextFrame()` wird ein Foto von der aktuellen Bühne gemacht.

- Im nächsten Durchlauf der äußeren Schleife wird nun die Bühne ein wenig verändert und dann wird wieder mit `nextFrame()` ein Foto gemacht. Dies wird so lange fortgesetzt, bis 300 Bilder fertig sind.
- In der inneren Schleife wird die jeweilige Bühne geändert. Dort werden die vier Textzeilen immer leicht verschoben – mit `moveTo()` – und leicht gedreht mit `rotateTo()`. Die komplizierten Formeln bei beiden Methoden kann man getrost ignorieren. Kompliziert aus dem Grund, weil bei ihrer Zusammenstellung Wert darauf gelegt wurde, dass a) die Bewegung und die Rotation der Texte möglichst angenehm aussehen (also nicht zu hektisch und auch nicht zu langsam sind) und b) sich die Texte nach 300 Bildern wieder in ihrer Ausgangsposition befinden.

Danach ist der Flash-Film fertig und kann mit der `output()`-Methode des Film-Objekts ausgegeben werden.

```
header('Content-Type: application/x-shockwave-flash');
$m->output(0);
```

Und hier noch einmal das komplette Listing:

```
<?php
```

```
$f = new SWFFont("AnkeCalligraph.fdb");

$m = new SWFMovie();
$m->setRate(24.0);
$m->setDimension(520, 320);
$m->setBackground(251, 121, 34);

function text($r, $g, $b, $a, $rot, $x, $y, $scale, $string)
{
    global $f, $m;

    $t = new SWFText();
    $t->setFont($f);
    $t->setColor($r, $g, $b, $a);
    $t->setHeight(96);
    $t->moveTo(-($t->getWidth($string))/2, 32);
    $t->addString($string);

    $i = $m->add($t);
    $i->rotateTo($rot);
    $i->moveTo($x, $y);
    $i->scale($scale, $scale);

    return $i;
}

$colorr[1]=255*0.85;
```

```

$colorg[1]=255*0.85;
$colorb[1]=255*0.85;

$colorr[2]=255*0.9;
$colorg[2]=255*0.9;
$colorb[2]=255*0.9;

$colorr[3]=255*0.95;
$colorg[3]=255*0.95;
$colorb[3]=255*0.95;

$colorr[4]=255;
$colorg[4]=255;
$colorb[4]=255;

$c=1;
$anz=4;
$step=4/$anz;

for($i=0;$i<$anz;$i+=1)
{
    $x=0;
    $y=0;
    $size=($i/5+0.2);
    $t[$i]=text($colorr[$c],$colorg[$c],$colorb[$c],
        0xff,0,$x,$y, $size,$text);
    $c+=$step;
}

$frames=300;
for($j=0; $j<$frames; $j++)
{
    for($i=0; $i<$anz; $i++)
    {
        $t[$i]->moveTo(
            260+round(sin($j/$frames*2*pi()+$k)
                *(50+50*(($i+1)))),
            160+round(sin($j/$frames*4*pi()+$i)
                *(20+20*(($i+1))))
        );
        $t[$i]->rotateTo(round(sin($j/$frames*2*pi()+$i/10)*360));
    }
    $m->nextFrame();
}

header('Content-Type: application/x-shockwave-flash');
$m->output(0);

?>

```

4.4.2 Balkendiagramm

In diesem Abschnitt folgt eine etwas seriösere PHP/Flash-Anwendung: die Erstellung eines animierten Balkendiagramms.

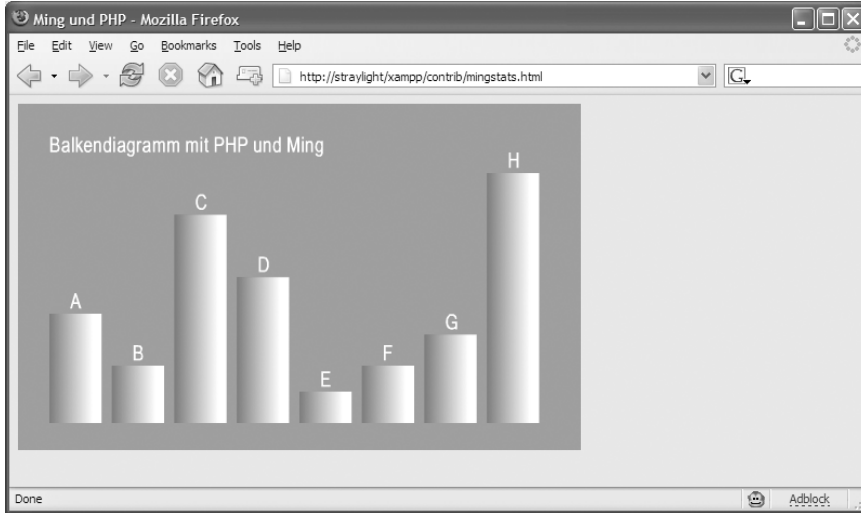


Abbildung 4.29: Die Säulen oder Balken sind animiert: In diesem Beispiel sind sie innerhalb weniger Sekunden auf die hier sichtbare Größe gewachsen.

Sie finden dieses Beispielprogramm in der Datei `mingstats_swf.php` im Verzeichnis `htdocs/xampp/contrib` unter Linux bzw. `htdocs\xampp\contrib` unter Windows.

Ganz am Anfang finden Sie folgenden Block:

```
$values["A"]=rand(20,230);
$values["B"]=rand(20,230);
$values["C"]=rand(20,230);
$values["D"]=rand(20,230);
$values["E"]=rand(20,230);
$values["F"]=rand(20,270);
$values["G"]=rand(20,270);
$values["H"]=rand(20,270);
```

Hier werden die Werte – in diesem Fall Zufallszahlen im Bereich von 20 bis 230 bzw. 270 – und die Bezeichnungen (»A« bis »H«) für die einzelnen Säulen des Graphen definiert.

```
$max=270;

$width=540;
$height=320;
```

In der Variablen `$max` wird eingestellt, welcher der größte Wert ist, der innerhalb des Graphen zu erwarten ist. Die anderen beiden Variablen `$width` und `$height` bestimmen die Größe des Flash-Films.

```
$m = new SWFMovie();  
$m->setDimension($width, $height);  
$m->setBackground(251, 121, 34);  
$m->setRate(30.0);
```

Hier wurde der Flash-Film erzeugt und auf die Größe `$width` x `$height` Pixel skaliert. Der Hintergrund wird mit `setBackground()` in Orange gefüllt. Danach wird die Frame-rate auf 30 Bilder pro Sekunde gesetzt, ein Wert, der für diese Anwendung am geeignetsten erschien.

Danach wird die Schrift für den Text innerhalb des Graphen geladen. Die Schriftdatei `BabelSans-B.fdb` befindet sich auch im `contrib`-Verzeichnis.

```
$font = new SWFFont("BabelSans-B.fdb");
```

Als Nächstes wird ein Verlauf von Schwarz nach Weiß definiert:

```
$g = new SWFGradient();  
$g->addEntry(0.0, 0, 0, 0);  
$g->addEntry(1.0, 0xff, 0xff, 0xff);
```

Dieser Verlauf wird in der Variablen `$g` gespeichert, auf die dann in der folgenden Funktion zurückgegriffen wird:

```
function box($w,$h)  
{  
    global $g;  
  
    $s = new SWFShape();  
    $f=$s->addFill($g, SWFFILL_LINEAR_GRADIENT);  
    $f->scaleTo(0.05);  
    $s->setRightFill($f);  
    $s->movePenTo(0, 0);  
    $s->drawLineTo($w, 0);  
    $s->drawLineTo($w, -$h);  
    $s->drawLineTo(0, -$h);  
    $s->drawLineTo(0, 0);  
    return $s;  
}
```

In Ming gibt es keine extra Funktion, um Rechtecke zu zeichnen. In diesem Fall hilft die `box()`-Funktion weiter. Sie zeichnet ein Rechteck in der Größe der beiden Parameter `$w` und `$h`. Zusätzlich wird dieses Rechteck mit dem Verlauf in der Variablen `$g` gefüllt.

Es folgt eine weitere kleine Hilfsfunktion:

```
function text($string)
{
    global $font;

    $t = new SWFText();
    $t->setFont($font);
    $t->setColor(255, 255, 255);
    $t->setHeight(20);
    $t->addString($string);

    return $t;
}
```

Diese Funktion zeichnet einfach den Text im Parameter `$string` in weißer Farbe und in der Schrift, die in `$font` definiert wurde. Diese Funktion wird für die Bezeichnungen der einzelnen Säulen und die Überschrift verwendet:

```
$t=$m->add(text("Balkendiagramm mit PHP und Ming"));
$t->moveTo(30,40);
```

Nachdem die Überschrift ausgegeben wurde, werden in der folgenden Schleife die Rechtecke und die Beschriftungen für die acht Säulen erzeugt und in den beiden Arrays `$text[]` und `$box[]` gespeichert.

```
$i=0;
reset($values);
while(list($key,$value) = each($values))
{
    $text[$i]=$m->add(text($key));
    $text[$i]->moveTo(50+$i*60,50);
    $box[$i]=$m->add(box(50,1));
    $box[$i]->moveTo(30+$i*60,$height-20);
    $i++;
}
```

In der folgenden Schleife werden die Säulen so lange größer skaliert, bis die maximale Höhe von `$max` erreicht ist. Mittels `scaleTo()` wird die Skalierung der Säulen vorgenommen und mit der `moveTo()`-Methode die Beschriftung entsprechend weiter nach oben geschoben.

```
for($f=0;$f<=$max;$f+=5)
{
    $i=0;
    reset($values);
    while(list($key,$value) = each($values))
    {
        $h=$value;
```

```

        if($h>$f)
        {
            $box[$i]->scaleTo(1,$f);
            $text[$i]->moveTo(50+$i*60,$height-$f-25);
        }
        $i++;
    }
    $m->nextFrame();
}

```

Damit die Animation nicht immer wieder von Neuem beginnt, wird sie mit dem **ActionScript-Befehl** `stop()` angehalten:

```

$m->add(new SWFAction("stop();"));
$m->nextFrame();

```

Letztendlich ist der Flash-Film nun fertig und kann mit der `output()`-Methode ausgegeben und zum Browser des Benutzers geschickt werden.

```

header('Content-type: application/x-shockwave-flash');
$m->output(0);

```

Und hier noch einmal das komplette Listing:

Listing 4.7: Das Flash-Beispiel mingstats_swf.php

```
<?php
```

```

$values["A"]=rand(20,230);
$values["B"]=rand(20,230);
$values["C"]=rand(20,230);
$values["D"]=rand(20,230);
$values["E"]=rand(20,230);
$values["F"]=rand(20,270);
$values["G"]=rand(20,270);
$values["H"]=rand(20,270);

$max=270;

$width=540;
$height=320;

$m = new SWFMovie();
$m->setDimension($width, $height);
$m->setBackground(251, 121, 34);
$m->setRate(30.0);

$font = new SWFFont("BabelSans-B.fdb");

```



```

$g = new SWFGradient();
$g->addEntry(0.0, 0, 0, 0);
$g->addEntry(1.0, 0xff, 0xff, 0xff);

function box($w,$h)
{
    global $g;

    $s = new SWFShape();
    $f=$s->addFill($g, SWFFILL_LINEAR_GRADIENT);
    $f->scaleTo(0.05);
    $s->setRightFill($f);
    $s->movePenTo(0, 0);
    $s->drawLineTo($w, 0);
    $s->drawLineTo($w, -$h);
    $s->drawLineTo(0, -$h);
    $s->drawLineTo(0, 0);
    return $s;
}

function text($string)
{
    global $font;

    $t = new SWFText();
    $t->setFont($font);
    $t->setColor(255, 255, 255);
    $t->setHeight(20);
    $t->addString($string);

    return $t;
}

$t=$m->add(text("Balkendiagramm mit PHP und Ming"));
$t->moveTo(30,40);

$i=0;
reset($values);
while(list($key,$value) = each($values))
{
    $text[$i]=$m->add(text($key));
    $text[$i]->moveTo(50+$i*60,50);
    $box[$i]=$m->add(box(50,1));
    $box[$i]->moveTo(30+$i*60,$height-20);
    $box[$i]->setName("box".$i);
    $i++;
}

for($f=0;$f<=$max;$f+=5)

```

```

{
    $i=0;
    reset($values);
    while(list($key,$value) = each($values))
    {
        $h=$value;
        if($h>$f)
        {
            $box[$i]->scaleTo(1,$f);
            $text[$i]->moveTo(50+$i*60,$height-$f-25);
        }
        $i++;
    }
    $m->nextFrame();
}

$m->add(new SWFAction("stop();"));
$m->nextFrame();

header('Content-type: application/x-shockwave-flash');
$m->output(0);
?>

```

4.5 Benutzerauthentifizierung

Nicht alles, was man selbst ins Internet stellt, soll von jedermann genutzt werden können, und daher muss man die Möglichkeit haben, den Zugang zu bestimmten Funktionen oder Seiten einzuschränken.

Am Beispiel der CD-Verwaltungsdemo von XAMPP werden wir Ihnen zeigen, wie Sie ganz einfach Ihre eigenen Webapplikationen mit einem individuellen Passwort-schutz versehen können.

4.5.1 Benutzerdatenbank mit phpMyAdmin einrichten

Wahrscheinlich wollen Sie Ihre Daten nicht nur mit einem Passwort und einem Benutzernamen schützen, sondern unterschiedlichsten Benutzern Zugänge geben. Richten Sie dazu zunächst eine entsprechende Benutzertabelle in Ihrer Datenbank ein.

In diesem Beispiel verwenden wir dazu die Tabelle `cds_auth`, die Sie am einfachsten in der `cdcol`-Datenbank mittels *phpMyAdmin* anlegen:

1. Rufen Sie phpMyAdmin auf und melden Sie sich als Benutzer *root* an.
2. Wählen Sie die Datenbank `cdcol` aus.
3. Im unteren Bereich haben Sie nun die Möglichkeit, eine neue Tabelle innerhalb der `cdcol`-Datenbank anzulegen:

Create new table on database cdcol

Name: Number of fields:

Go

Abbildung 4.30: Legen Sie die neue Tabelle mit drei Feldern unter dem Namen `cds_auth` an.

4. Definieren Sie die einzelnen Felder für diese Tabelle:

Server: localhost ▶ Database: cdcol ▶ Table: cds_auth

Field	Type ?	Length/Values ¹	Collation
id	INT		
username	VARCHAR	20	
password	VARCHAR	20	

Abbildung 4.31: Die einzelnen Felder für die Minibenutzerdatenbank

Das `id`-Feld sollten Sie noch als `auto_increment` und `primary key` definieren. Für dieses Beispiel ist es zwar nicht notwendig, längerfristig wird dies aber wahrscheinlich von Vorteil sein.

5. Nachdem Sie die Tabelle eingerichtet haben, fügen Sie nun über die Einfügen-Funktion von phpMyAdmin einen ersten Benutzer in die Tabelle ein:

Server: localhost ▶ Database: cdcol ▶ Table: cds_auth

Browse Structure SQL Search Insert Export Import Operations

Field	Type	Function	Null	Value
id	int(11)			
username	varchar(20)			oswald
password	varchar(20)			geheim

Abbildung 4.32: Der erste Benutzer »oswald« mit dem Passwort »geheim«

Insgesamt sollte Ihre `cdcol`-Datenbank in phpMyAdmin nun aussehen wie in Abbildung 4.33.

Möchten Sie später weitere Benutzer hinzufügen, bestehende Benutzer löschen oder ein Passwort ändern, dann können Sie das hier über phpMyAdmin erledigen.

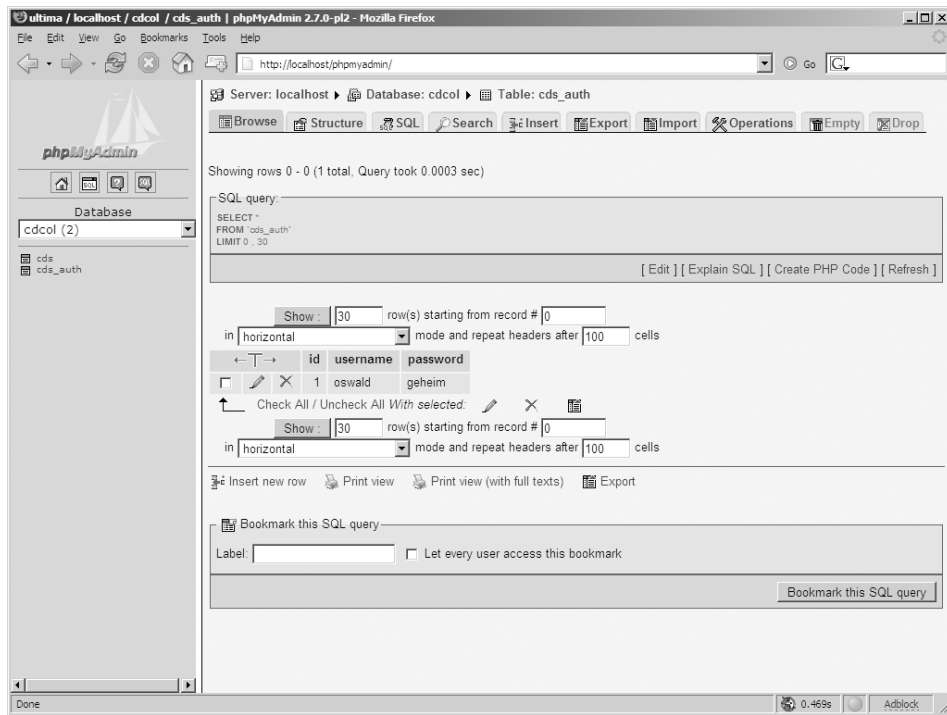


Abbildung 4.33: In der Datenbank *cdcol* gibt es nun die Tabelle *cds_auth* mit den Spalten *id*, *username* und *password*. Beispielhaft wurde der erste Benutzer »oswald« mit dem Passwort »geheim« eingerichtet.

4.5.2 Passwortabfrage programmieren

Mit dem Erstellen der Benutzertabelle haben Sie schon den größten Schritt hinter sich gebracht. Als Nächstes müssen Sie nur noch die Passwortabfrage programmieren. Dies ist wirklich recht einfach: Lassen Sie den Webbrowser die Arbeit machen und verwenden Sie zur Benutzerauthentifizierung HTTP-Auth.

Wechseln Sie in das Verzeichnis *htdocs/xampp*, also dort, wo auch die CD-Verwaltungsdemo von XAMPP liegt. Legen Sie hier eine neue Datei namens *cds_auth.php* an und füllen Sie diese mit folgendem Inhalt:

Listing 4.8: Ganz einfaches HTTP-Auth mit PHP und MySQL-Datenbank

```
<?php

$username=addslashes($_SERVER['PHP_AUTH_USER']);
$password=addslashes($_SERVER['PHP_AUTH_PW']);
mysql_connect("localhost","root","");
mysql_select_db("cdcol");
```

```

$results = mysql_query("SELECT * FROM cds_auth WHERE username='$username' AND
password LIKE '$password'");

if(mysql_num_rows($results)==0)
{
    header( 'WWW-Authenticate: Basic realm="cds" ' );
    header( 'HTTP/1.0 401 Unauthorized' );

    echo "Passwort erforderlich!";
    die();
}
?>

```

Dieses Skript besteht im Großen und Ganzen aus drei Teilen. Der erste Teil sieht folgendermaßen aus:

```

$username=addslashes($_SERVER['PHP_AUTH_USER']);
$password=addslashes($_SERVER['PHP_AUTH_PW']);

```

Hier werden Benutzername und Passwort, die über HTTP-Auth übermittelt wurden, in die Variablen `$username` und `$password` geschrieben. (Was es mit den `addslashes()` auf sich hat, soll hier nicht erklärt werden, darauf wird im Abschnitt »SQL-Injections« in Kapitel 10, »Sicherheit«, genauer und konkret eingegangen.)

Im zweiten Teil wird die Verbindung zur Datenbank aufgebaut und angefragt, ob es einen Benutzer mit diesem Passwort gibt:

```

mysql_connect("localhost","root","");
mysql_select_db("cdcol");

$results = mysql_query("SELECT * FROM cds_auth WHERE username='$username' AND
password LIKE '$password'");

```

Übrigens wird hier absichtlich einmal `=` und einmal `LIKE` verwendet, um die Spalten `username` und `password` zu vergleichen: Der Vergleich mit `=` ist bei MySQL *case-insensitive* und der Vergleich mit `LIKE` ist *case-sensitive*. Beim Benutzernamen wird also nicht zwischen Groß- und Kleinbuchstaben unterschieden, beim Passwort schon. Also so, wie man es meist auch gerne haben möchte.

Im dritten Teil wird nun nur noch der Negativfall abgefangen. Wenn Benutzername und Passwort nicht stimmen, die Anfrage an die Datenbank also keine treffende Zeile in der Datenbank gefunden hat, dann wird der HTTP-Fehlercode 401 (also Unauthorized) an den Browser geschickt und danach beendet sich das PHP-Programm komplett.

```

if(mysql_num_rows($results)==0)
{
    header( 'WWW-Authenticate: Basic realm="cds" ' );

```

```
header( 'HTTP/1.0 401 Unauthorized' );

echo "Paßwort erforderlich!";
die();
}
```

Erhält der Browser den HTTP-Code 401, stellt er automatisch einen Login-Dialog dar und bittet den Benutzer, sich zu authentifizieren. Soviel zum Inhalt dieser Datei.

Nachdem Sie die Datei `cds_auth.php` erstellt haben, ändern Sie noch die Datei `cds.php` so, dass am Anfang die `cds_auth.php`-Datei mit dem `include()`-Befehl eingelesen wird:

```
<?
include("cds_auth.php");
if($_REQUEST['action']=="getpdf")
{
    mysql_connect("localhost","root","");
    mysql_select_db("cdcol");
```

Wenn Sie nun über Ihren Browser auf die CD-Verwaltungsdemo zugreifen, wird Ihnen von Ihrem Browser ein Login-Dialog mit der Aufforderung angezeigt, einen Benutzernamen und ein Passwort einzugeben. Haben Sie sich an das Beispiel weiter oben gehalten, dann geben Sie hier »oswald« als Benutzername und »geheim« als Passwort ein:

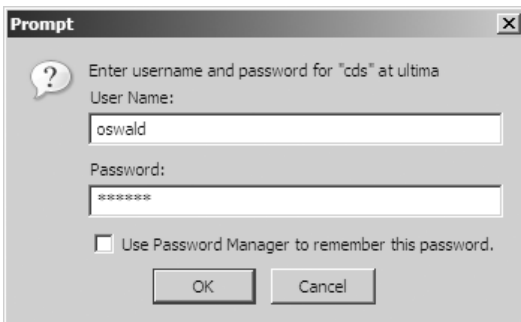


Abbildung 4.34: Die CD-Verwaltung ist nun mit einem Benutzernamen/Passwort geschützt.

Die CD-Verwaltung ist nun also nur noch für authentifizierte Benutzer erreichbar.



Hinweis

Noch zwei Anmerkungen zum Thema Sicherheit an dieser Stelle:

1. Es ist keine wirklich gute Idee, Passwörter im Klartext zu speichern. Das hier verwendete Programm soll nur ein Beispiel sein und wurde absichtlich so einfach wie möglich gehalten. In der Realität empfiehlt es sich, Hash-Algorithmen wie `md5sum()` in PHP oder `PASSWORD()` in MySQL zu verwenden.
2. In diesem Beispiel wird HTTP-Auth (Basic) verwendet, um die Authentifizierung des Benutzers zu ermöglichen. Diese Form von HTTP-Auth überträgt das Passwort quasi im Klartext. Damit ist das Passwort potentiell abhörbar. Aus diesem Grund sollte man HTTP-Auth (Basic) nach Möglichkeit nur in Verbindung mit HTTPS einsetzen.



5 ProFTPD

FTP-Server gibt es für Linux- und Unix-Systeme fast schon wie Sand am Meer. Die Großmutter aller FTP-Server ist mit Sicherheit der WU-FTP. Allerdings muss man auch zugeben, dass der WU-FTP ein Konfigurationsmonster und nicht wirklich leicht zu administrieren ist. Da er obendrein vom Leistungsumfang nicht mehr mit moderneren FTP-Servern mithalten kann, haben wir uns bei der Entscheidung, welchen FTP-Server wir für die Linux-Version mit in XAMPP aufnehmen, recht schnell gegen den WU-FTP und für den kleineren ProFTPD entschieden.

Welcher von den vielen FTP-Servern nun der beste ist, ist objektiv sicherlich schwer zu entscheiden. Wir haben uns für den ProFTPD in XAMPP entschieden, weil er folgende – aus unserer Sicht wichtige – Vorteile bietet:

- Er ist sehr einfach zu konfigurieren.
- Seine Konfigurationsdatei erinnert an die Apache `httpd.conf`.
- Er unterstützt *Virtual Hosts*.
- Er unterstützt die Authentifizierung mittels SQL, LDAP, RADIUS.
- Er enthält eine sehr einfache *Anonymous FTP*-Unterstützung.

Diese Gründe sind dafür verantwortlich, dass seit fast der allerersten XAMPP-Version der ProFTPD mit im XAMPP-Paket enthalten ist.

5.1 XAMPP und ProFTPD

Die Konfigurationsdatei des ProFTPD befindet sich unter `/opt/lampp/etc/proftpd.conf`. Der Inhalt ist recht übersichtlich und die gesamte Datei passt fast auf eine knappe Bildschirmseite. Im Großen und Ganzen entspricht sie der normalen ProFTPD-Konfiguration.

Zwei Besonderheiten sind jedoch in der Konfiguration von XAMPP zu beachten:

- Der ProFTPD ist so konfiguriert, dass sich der Benutzer `nobody` mit dem Passwort `lampp` einloggen kann. (Normalerweise dürfen sich nur normale Benutzer des Linux-Systems mit ihrem eigenen Passwort einloggen.)
- Aus Sicherheitsgründen ist für alle Benutzer der Zugriff nur auf das Verzeichnis `/opt/lampp/htdocs` beschränkt.

Werfen wir daher einen Blick in die Konfigurationsdatei und schauen uns die besonderen Konfigurationen in der `proftpd.conf` genauer an.

```
<Directory /opt/lampp/htdocs/*>  
    AllowOverwrite                on  
</Directory>
```

Mit dieser Konfiguration erlaubt der FTP-Server, dass Dateien in das `htdocs`-Verzeichnis hochgespielt und ggf. bestehende Dateien überschrieben werden dürfen.

```
DefaultRoot /opt/lampp/htdocs
```

Diese Konfigurationsdirektive sorgt dafür, dass alle Benutzer, die sich über FTP anmelden, nur Zugriff auf das `htdocs`-Verzeichnis und die darunter liegenden Unterverzeichnisse haben.

```
UserPassword nobody wRPBu8u4YP0CY
```

Mit dieser Zeile wird das Passwort `lamp` für den Benutzer `nobody` gesetzt. Normalerweise hat der Benutzer `nobody` kein gültiges Passwort, so dass er sich eigentlich niemals anmelden kann. Die Zeichenfolge `wRPBu8u4YP0CY` ist das mit der `crypt()`-Funktion verschlüsselte Passwort `lamp`.

Achtung



Um unerwünschten Zugriff zu vermeiden, sollte das Passwort für den Benutzer `nobody` schnellstens geändert werden. Mehr dazu siehe in Kapitel 10, »Sicherheit«.

```
RequireValidShell off
```

Normalerweise dürfen sich nur Benutzer via FTP anmelden, für die eine gültige Shell eingestellt ist. Da `nobody` normalerweise nicht so eine Shell hat, wird diese Einschränkung hier ausgeschaltet. Erst dadurch ist es möglich, dass sich `nobody` überhaupt anmelden kann.

```
UseFtpUsers off
```

Eigentlich dürfen sich nur Benutzer via FTP anmelden, die nicht(!) in der Datei `/etc/ftpusers` erwähnt werden. Normalerweise steht aber `nobody` in dieser Datei, und damit er sich anmelden kann, muss auch dieses Hindernis überwunden werden.

Wie man sieht, wurde ein großer Aufwand betrieben, um dem Benutzer `nobody` das Einloggen via FTP zu erlauben. Hinter alledem steckt aber ein wichtiger Grund:

XAMPP will das bestehende Linux-System nach Möglichkeit nicht »anfassen« bzw. ändern und trotzdem den Benutzern von XAMPP schnell ein lauffähiges System zur Verfügung stellen.

XAMPP ist daher so konfiguriert, dass *nobody* sofort mit FTP zugreifen kann und auch die Rechte hat, um im *htdocs*-Verzeichnis Dateien zu ändern und hinzuzufügen.



Achtung

Langfristig sollte *nobody* aber nicht zum Arbeiten mit XAMPP benutzt, sondern ein eigener Benutzer verwendet werden. Allerdings muss man sich dann auch näher mit den Zugriffsrechten unter Linux beschäftigen.

Definitiv keine gute Idee ist es, auch hierfür den Benutzer *root* zu verwenden. In der gesamten Unix- und Linux-Literatur wird davor gewarnt, den *root*-Benutzer für Dinge zu verwenden, die nicht ausschließlich der Administration des Systems dienen.

5.2 Virtual Hosts

Vom Apache kennt man die sog. *Virtual Hosts* und auch die Unterscheidung von *name-based virtual hosts* und *IP-based virtual hosts*. Erstere werden anhand des Hostnamens unterschieden, über den sie angesprochen werden, und Letztere über ihre IP-Adresse. Da die *name-based virtual hosts* helfen, IP-Adressen zu sparen, werden beim Apache hauptsächlich diese verwendet.

Bei FTP hat man leider nicht die Wahl zwischen diesen beiden Sorten von Virtual Hosts: Hier gibt es leider nur die *IP-based virtual hosts* und nicht die *name-based virtual hosts*. Das liegt nicht an ProFTPD, sondern am zugrunde liegenden FTP-Protokoll. Dort war nie so etwas wie Virtual Hosts vorgesehen und dies ist ohne aufwändige Protokolländerungen auch unmöglich.

Das bedeutet auch: Wer auf seinem Rechner keine zusätzlichen IP-Adressen besitzt, braucht sich auch gar nicht erst die Mühe zu machen, sich mit virtuellen Hosts unter FTP zu beschäftigen.

Damit wir hier trotzdem zumindest unter Laborbedingungen mit ProFTPD und Virtual Hosts arbeiten können, verwenden wir die besondere Eigenschaft des loopback-Device: Jeder Rechner, der über TCP/IP verfügt, erreicht sich selbst unter allen IP-Adressen, die mit 127 anfangen. Die bekannteste dürfte die 127.0.0.1 sein. Aber auch 127.0.0.2 oder 127.1.2.3 oder 127.200.100.10 sind IP-Adressen des eigenen Rechners.

Was wir also ganz bequem hinbekommen sollten, ist, für eine dieser 127er-Adressen einen funktionierenden Virtual Host zu konfigurieren.

Achtung



Natürlich funktionieren diese 127er-Virtual Hosts nur dann, wenn man von dem gleichen Rechner auf sie zugreift. Will man wirkliche, von außen über das Netzwerk erreichbare Virtual Hosts, muss man dafür sorgen, dass der XAMPP-Server mehrere gültige IP-Adressen bekommt.

Dazu werden einfach die folgenden drei Zeilen ans Ende der ProFTPd-Konfigurationsdatei `/opt/lampp/etc/proftpd.conf` gehängt.

```
<VirtualHost 127.0.0.2>
    DefaultRoot /opt/lampp/htdocs/xampp/lang
</VirtualHost>
```

Damit wird ein Virtual Host für die IP-Adresse 127.0.0.2 definiert. Das Besondere an diesem Host ist, dass das DefaultRoot-Verzeichnis auf `/opt/lampp/htdocs/xampp/lang` geändert wurde.

Nachdem man die `proftpd.conf` geändert hat, muss man den ProFTPd noch anweisen, diese Datei erneut einzulesen. Am einfachsten geht das mit folgendem Befehl:

```
straylight /# /opt/lampp/lampp reloadftp
XAMPP: Reload ProFTPd...
```

Danach kann man seinen Virtual Host gleich testen: Verbinden Sie sich einfach mit dem Linux FTP-Client auf die 127.0.0.2 und melden Sie sich dort als normaler Benutzer an. Wenn alles geklappt hat, sollten Sie sich nach der Anmeldung im `xampp/lang`-Verzeichnis befinden.

Listing 5.1: Protokoll einer FTP-Anmeldung auf der 127.0.0.2

```
straylight /# ftp 127.0.0.2
220 ProFTPd 1.2.10 Server (ProFTPd) [127.0.0.2]
Name (127.0.0.2:oswald): oswald
331 Password required for nobody.
Password: *****
230 User nobody logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
200 EPRT command successful
150 Opening ASCII mode data connection for file list
-rw-r--r--  1 root    root      13127 Apr  1 09:31 de.php
-rw-r--r--  1 root    root      11612 Apr  1 09:31 en.php
-rw-r--r--  1 root    root      11975 Apr  1 09:31 es.php
```

```

-rw-r--r-- 1 root    root      17816 Apr  1 09:52 fr.php
-rw-r--r-- 1 root    root      25100 Apr  1 09:51 it.php
-rw-r--r-- 1 root    root      11909 Apr  1 09:31 nl.php
-rw-r--r-- 1 root    root      25061 Apr 10 17:56 no.php
-rw-r--r-- 1 root    root      21613 Apr  1 09:31 pl.php
-rw-r--r-- 1 root    root      21543 Apr  1 09:31 zh.php
226 Transfer complete.
ftp> quit
221 Goodbye.
straylight /#

```

Zur Sicherheit wird hier auch gleich noch der Gegenbeweis aufgeführt:

Listing 5.2: Protokoll einer FTP-Anmeldung unter 127.0.0.1

```

straylight /# ftp 127.0.0.1
Connected to 127.0.0.1.
220 ProFTPD 1.2.10 Server (ProFTPD) [127.0.0.1]
Name (127.0.0.1:oswald): oswald
331 Password required for oswald.
Password: *****
230 User oswald logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
200 EPRT command successful
150 Opening ASCII mode data connection for file list
drwxr-xr-x  2 root    root      1272 Jan 25  2003 apache
-rwxr-xr-x  1 nobody  root        163 Oct 31  2003 index.html
drwxr-xr-x  2 nobody  root        48 Dec 26  2004 webalizer
drwxr-xr-x  6 root    root      1560 Sep  4 10:34 xampp
226 Transfer complete.
ftp> quit
221 Goodbye.

```

Beim zweiten FTP über die 127.0.0.1 befinden Sie sich nach der Anmeldung – wie gewohnt – im `htdocs`-Verzeichnis von XAMPP. Über die beiden IP-Adressen sind Sie also in unterschiedlichen Verzeichnissen gelandet und somit hat die Konfiguration des Virtual Hosts funktioniert.

5.3 Anonymous FTP

Anonymous FTP stammt noch aus einer Zeit weit vor dem World Wide Web. Es existierten weder HTTP noch P2P. Um Dateien zum Download anzubieten oder auch nur untereinander auszutauschen, gab es den sog. Anonymous FTP-Server.

Anonymous genannt deswegen, weil man sich hier nicht mit einer eigenen Benutzerkennung beim FTP-Server anmelden musste, sondern mit der anonymen Kennung `anonymous` oder `ftp`.

Anonymous FTP-Server sind heutzutage wahrscheinlich noch nicht ganz von der virtuellen Bildfläche verschwunden, weil sie im Gegensatz zum HTTP den Upload von Dateien ganz leicht möglich machen.

Auf den folgenden Seiten werden wir daher zwei Konfigurationsmöglichkeiten eines Anonymous FTP-Servers vorstellen: die eine mit und die andere ohne Upload-Möglichkeit.

Hinweis



Wir verwenden in der folgenden Konfiguration den Benutzer `nobody`, um einen Anonymous FTP-Zugang zu realisieren. Aus historischen Gründen wird hierzu normalerweise der Benutzer `ftp` verwendet. Da es aber den Benutzer `ftp` nicht auf jedem Linux-System gibt – den Benutzer `nobody` hingegen schon –, erscheint es uns sinnvoller, diesen Benutzer zu verwenden. Aus technischer oder auch sicherheitstechnischer Sicht spielt das absolut keine Rolle. Diese Vorgehensweise ist nur wesentlich einfacher.

5.3.1 Anonymous FTP ohne Upload-Möglichkeit

Zunächst sollte man sich entscheiden, welches Verzeichnis man für den Anonymous FTP-Server verwenden will. Für dieses Beispiel nehmen wir das Verzeichnis `/opt/lampp/anonymous` und erstellen es im ersten Schritt mit dem `mkdir`-Befehl:

```
mkdir /opt/lampp/anonymous
```

In dieses Verzeichnis kommt nun noch ein `pub`-Verzeichnis:

```
mkdir /opt/lampp/anonymous/pub
```

Im Falle von ProFTPD besteht eigentlich keine wirkliche technische Notwendigkeit für dieses Verzeichnis, aber es ist bei Anonymous FTP-Servern so gebräuchlich, dass alle Dateien, die über ihn veröffentlicht werden, über das Verzeichnis `pub` zu erreichen sind.

Im nächsten Schritt richten wir die Zugriffsrechte so ein, dass der FTP-Server diese Verzeichnisse auch lesen darf:

```
chmod a+rx /opt/lampp/anonymous  
chmod a+rx /opt/lampp/anonymous/pub
```

In das Verzeichnis `/opt/lampp/anonymous/pub` können dann die Dateien abgelegt werden, die man per FTP veröffentlichen möchte.

Nun wird die Datei `/opt/lampp/etc/proftpd.conf` um folgende Zeilen erweitert:

```
<Anonymous /opt/lampp/anonymous>
User nobody
Group nobody
UserAlias anonymous nobody
UserAlias anonymous ftp

<Limit WRITE>
    DenyAll
</Limit>
</Anonymous>
```

Danach wird ProFTPD angewiesen, die Konfigurationsdatei erneut einzulesen:

```
/opt/lampp/lampp reloadftp
```

Und schon ist der Anonymous FTP-Server bereit. Melden Sie sich jetzt einfach mal mit einem FTP-Client und dem Benutzernamen `ftp` an (beim Passwort wird man gebeten, die eigene E-Mail-Adresse einzugeben. Das kann man, muss man aber nicht befolgen):

```
straylight /# ftp 127.0.0.1
Connected to 127.0.0.2.
220 ProFTPD 1.2.10 Server (ProFTPD) [127.0.0.1]
Name (127.0.0.2:oswald): ftp
331 Anonymous login ok, send your email address as your password.
Password: eg@l
230 Anonymous access granted, restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> dir
200 EPRT command successful
150 Opening ASCII mode data connection for file list
drwxr-xr-x  2 root   root       48 Sep  2 12:00 pub
226 Transfer complete.
ftp> quit
```

Sie sehen, dass es möglich ist, sich mit dem Benutzernamen `ftp` anzumelden und als Passwort irgendetwas einzugeben. Mit dem Befehl `dir` kann man dann den Inhalt des Verzeichnisses `/opt/lampp/anonymous` sehen. Der Anonymous FTP-Server funktioniert also.

5.3.2 Anonymous FTP mit Upload-Möglichkeit

Um dem Anonymous FTP-Server auch noch eine Upload-Möglichkeit zu geben, sind noch ein paar weitere Schritte erforderlich.

Als Erstes benötigen Sie ein extra Verzeichnis, in das die Dateien hochgeladen werden sollen. Gebräuchlich ist hier der Verzeichnisname `incoming`:

```
mkdir /opt/lampp/anonymous/incoming
```

Dieses Verzeichnis weisen Sie im nächsten Schritt dem Benutzer und der Gruppe, unter dem bzw. der der Anonymous FTP-Server läuft, nämlich `nobody` und `nobody`.

```
chown nobody:nobody /opt/lampp/anonymous/incoming/
```

Damit stellen Sie sicher, dass der Upload von Dateien in Bezug auf die Linux-Zugriffsrechte kein Problem darstellt.

Ersetzen Sie nun den `<Anonymous>...</Anonymous>`-Container der bisherigen Anonymous FTP-Konfiguration in der `/opt/lampp/etc/proftpd.conf` durch folgende Zeilen:

```
<Anonymous /opt/lampp/anonymous>
User nobody
Group nobody
UserAlias anonymous nobody
UserAlias anonymous ftp
```

```
<Limit WRITE>
    DenyAll
</Limit>
```

```
<Directory incoming/*>
    <Limit READ>
        DenyAll
    </Limit>
    <Limit STOR>
        AllowAll
    </Limit>
</Directory>
</Anonymous>
```

Weisen Sie danach ProFTPD an, die Konfigurationsdatei erneut einzulesen:

```
/opt/lampp/lampp reloadftp
```

Und im nächsten Schritt testen Sie gleich den FTP-Zugang:

```
straylight /# ftp 127.0.0.1
Connected to 127.0.0.2.
220 ProFTPD 1.2.10 Server (ProFTPD) [127.0.0.1]
Name (127.0.0.2:oswald): ftp
331 Anonymous login ok, send your email address as your password.
Password: eg@l
```


230 Anonymous access granted, restrictions apply.
 Remote system type is UNIX.
 Using binary mode to transfer files.

Das Anmelden mit einem beliebigen Passwort hat funktioniert. Sehen Sie jetzt einmal nach, in welchem Verzeichnis Sie sich befinden.

```
ftp> dir
200 EPRT command successful
150 Opening ASCII mode data connection for file list
drwxr-xr-x  2 root    root          48 Sep  2 12:00 pub
drwxrwxrwx  2 nobody nobody       48 Sep  2 12:00 incoming
226 Transfer complete.
```

Das sieht genau so aus, wie es zu erwarten war. Es gibt ein pub- und ein incoming-Verzeichnis. Versuchen Sie jetzt einmal, eine Datei hochzuladen.

```
ftp> put test.txt
local: test.txt remote: test.txt
200 EPRT command successful
550 test.txt: Permission denied
```

Es hat nicht geklappt. Prima. Denn das sollte es ja auch nicht. Dafür ist nämlich das incoming-Verzeichnis da. Wechseln Sie also in dieses Verzeichnis und probieren Sie dort erneut das Hochladen:

```
ftp> cd incoming
250 CWD command successful
ftp> dir
200 EPRT command successful
150 Opening ASCII mode data connection for file list
226 Transfer complete.
ftp> put test.txt
local: test.txt remote: test.txt
200 EPRT command successful
150 Opening BINARY mode data connection for interbase.php
100% |*****| 437      5.27 MB/s   00:00 ETA
226 Transfer complete.
437 bytes sent in 00:00 (29.47 KB/s)
```

Nun hat es funktioniert. Überprüfen Sie dies mit dem dir-Befehl.

```
ftp> dir
200 EPRT command successful
150 Opening ASCII mode data connection for file list
-rw-r--r--  1 nobody  nobody       437 Sep  2 12:07 test.txt
226 Transfer complete.
ftp> get test.txt
```

```
200 EPRT command successful
550 test.txt: Permission denied
ftp> quit
```

Der Versuch, die Datei mit `get text.txt` herunterzuladen, ist fehlgeschlagen. Das ist auf den ersten Blick seltsam und sieht nach einem Fehler aus. Ist es aber nicht. Verantwortlich dafür sind die folgenden Zeilen, die wir – weiter oben – in die `proftpd.conf` eingefügt haben:

```
<Directory incoming/*>
  <Limit READ>
    DenyAll
  </Limit>
  ...
</Directory>
```

Mit diesen Zeilen wird das Herunterladen (READ) von Dateien aus dem `incoming`-Verzeichnis verboten (`DenyAll`). Diese Einstellung ist absolut wichtig, ohne sie sollte man niemals – niemals(!) – einen Anonymous FTP-Server betreiben. Nur so wird sichergestellt, dass der FTP-Server nicht von Fremden zum Austausch illegaler Dateien missbraucht wird.



6 FileZilla Server

Seit dem Jahr 2001 existiert das Open-Source-Projekt *FileZilla*, das einen FTP-Client sowie einen FTP-Server kostenfrei zur Verfügung stellt. Bei beiden Programmen handelt es sich um reine Windows-Applikationen. Sie liegen sowohl als Installer als auch als Quellcode vor und stehen unter der GPL-Lizenz. Die Homepage von FileZilla lautet <http://filezilla.sourceforge.net>.



Hinweis

Manchmal wird dem Apache HTTPD eine FTP-Server-Funktionalität zugesprochen, was selbstverständlich nicht korrekt ist. Der Apache verwaltet die Zugriffe über das *HyperText Transfer Protocol* (HTTP), ein FTP-Server regelt den Verkehr über das *File Transfer Protocol* (FTP). Einer FTP-Übertragung wird meist dann dem Transfer über dem *HyperText Transfer Protocol* den Vorzug gegeben, wenn es sich um größere Datenmengen handelt oder ein exklusiver Verzeichniszugriff für sog. »Uploads« vorgehalten werden soll.

Beschreibung	Standort
Hauptprogramm	<xampp>\FileZillaFTP\FileZilla Server.exe
Konfigurationsdatei	<xampp>\FileZillaFTP\FileZilla Server.xml
Batch-Datei Setup	<xampp>\filezilla_setup.bat
Batch-Datei Start	<xampp>\filezilla_start.bat
Batch-Datei Stop	<xampp>\filezilla_stop.bat

Tabelle 6.1: Die wichtigsten Dateien von FileZilla Server

Die Installation und Einrichtung als Dienst erfolgen hier über:

```
<xampp>\filezilla_setup.bat
```

Die Deinstallation erfolgt ebenfalls über die *filezilla_setup.bat*. Bei einer Deinstallation fragt FileZilla nach, ob der Server gestoppt werden soll. Erst im zweiten Schritt werden Sie angehalten, über den Dienst zu entscheiden. Haben Sie den Server mit der ersten Frage gestoppt, müssen Sie also die *filezilla_setup.bat* zum zweiten Mal bestätigen, um den FileZilla-Dienst zu entfernen.

Optional zur *filezilla_setup.bat* können Sie natürlich auch direkt die

`<xampp>\FileZillaFTP\filezilla\FileZilla Server.exe`

benutzen.

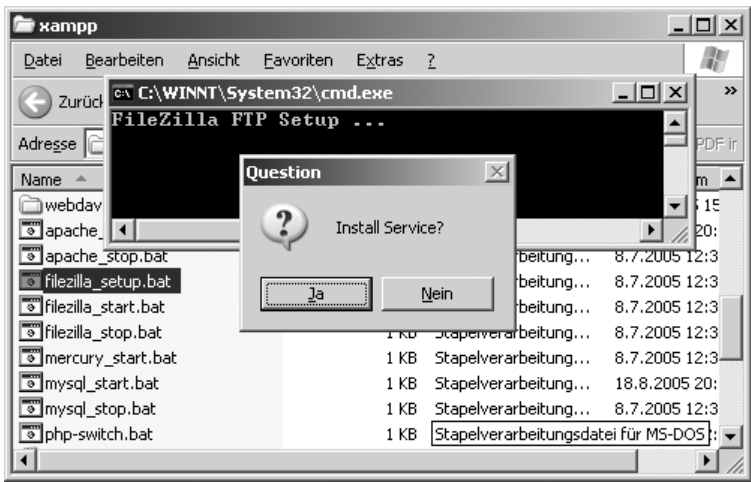


Abbildung 6.1: Das Setup des FileZilla Servers

Nach dem Start von FileZilla können Sie den Server gleich testen. Entweder Sie gehen in den Anonymous-Bereich, den Sie direkt mit allen gängigen Browsern unter der URL *ftp://127.0.0.1* bzw. *ftp://localhost* abrufen können. Oder Sie loggen sich als XAMPP-Testbenutzer *newuser* und mit dem Login *wampp* über einen FTP-Client Ihrer Wahl ein. Sie befinden sich danach automatisch im *htdocs*-Verzeichnis von XAMPP.

Benutzer	Passwort	Ziel
newuser	wampp	<code><xampp>\htdocs</code>
anonymous		<code><xampp>\anonymous</code>

Tabelle 6.2: Benutzer und ihre Zielverzeichnisse

**Achtung**

Aus Gründen der Sicherheit auf Ihrem System empfehlen wir, für den Testbenutzer das Passwort zu ändern oder diesen ganz zu löschen!

6.1 FileZilla Server-Interface

Nach der Einrichtung des Servers stellt das *FileZilla Server-Interface* (kurz: *FileZilla-Interface*) die Oberfläche für die weitere Konfiguration des Servers dar. Hierüber können Benutzer und Gruppen gebildet, Übertragungsmodi festgelegt oder Sicherheitsoptionen verändert werden. Der Pfad zum Interface im XAMPP lautet:

```
<xampp>\FileZillaFTP\FileZilla Server Interface.exe
```



Abbildung 6.2: Auf dem lokalen PC braucht das FileZilla-Interface kein Passwort!

Die Verbindung der GUI (*Graphical User Interface*) zum FileZilla Server kann vom lokalen Rechner ohne Passwort erfolgen. Sollten Sie eine Fernwartung planen, müssen Sie ein Passwort für den Zugang definieren.

Unter EDIT > SETTINGS finden Sie hierzu das Menü ADMIN INTERFACE SETTINGS. Dort definieren Sie das Passwort und die sog. BINDINGS, d.h. den IP-Bereich (Bereich der Internetprotokoll-Nummern im festgelegten Netz) für den Fernzugriff auf Ihren FTP-Server.

Selbstverständlich können Sie so mit einem Interface mehrere FileZilla Server überwachen. Voraussetzung ist die Einrichtung der Remote-Überwachung für jeden dieser Server.

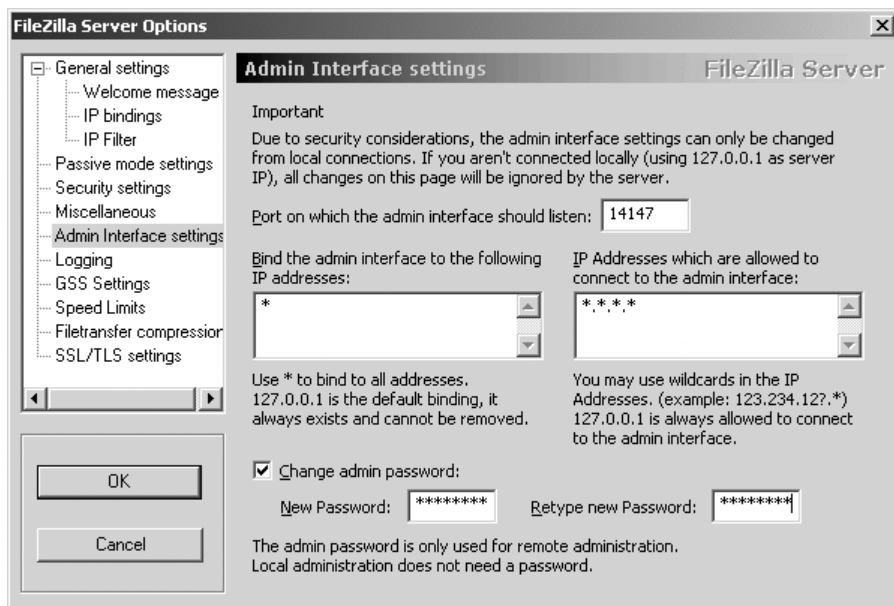


Abbildung 6.3: In diesem Fall dürfen alle IP-Nummern auf das Interface zugreifen.

6.2 Virtual Hosts

Um es gleich zu Beginn zu erwähnen: FileZilla Server der Version 0.9.10a kennt keine Virtual Hosts. Dieser Umstand bedeutet gleichzeitig, dass Sie hier einem Benutzer nur ein einmaliges FTP-Benutzerverzeichnis zuordnen können. Und damit können wir gleich zu der »normalen« Benutzer- und Gruppenverwaltung überleiten.

Hinweis



Über Virtual Hosts ist es möglich, mehrere verschiedene Benutzer mit einem Benutzernamen, beispielsweise *webmaster*, zu verwalten. Der Benutzer *webmaster* der Domain *host1.de* ist eine andere Person mit anderen Zugriffsrechten als der Benutzer *webmaster* der Domain *host2.de*. Sie besitzen zudem unterschiedliche Passwörter und Heimatpfade auf dem Zielrechner. Da ja einzelne Rechner mehrere Hosts und Domains verwalten können, ist ein »Virtual Host«-fähiger FTP-Server wie der ProFTPD unter Linux eine ideale Ergänzung zum Hosting-Management über den Apache HTTPD.

In diesem Beispiel soll dem Benutzer *zaphod* ein FTP-Zugang für seine Webseiten im XAMPP eingerichtet werden. Zunächst richten Sie hierfür über das FileZilla-Interface eine neue Gruppe mit dem Namen *www* ein. Öffnen Sie unter EDIT das *Groups*-Menü und erstellen Sie die neue Gruppe mit dem Stammverzeichnis `<xampp>\htdocs`.



Abbildung 6.4: Auf geht es zur Gruppenverwaltung im FileZilla-Interface.

In dem Verzeichnis legen Sie gleich danach ein reales Verzeichnis für *zaphod* über den Windows-Explorer an, also:

```
<xampp>\htdocs\zaphod
```

Erstellen Sie nun den Benutzer *zaphod* in der Gruppe *www* über das FileZilla-Interface. Vergessen Sie hierbei bitte nicht, das Heimatverzeichnis und die Rechte für das Verzeichnis zu definieren. Zu guter Letzt braucht der Benutzer *zaphod* noch ein Passwort.

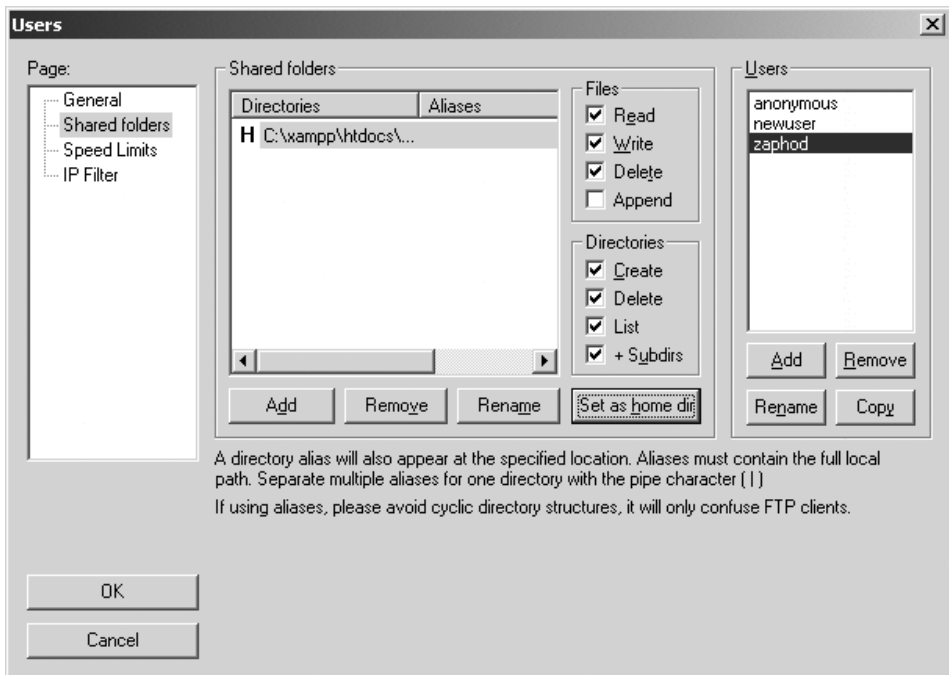


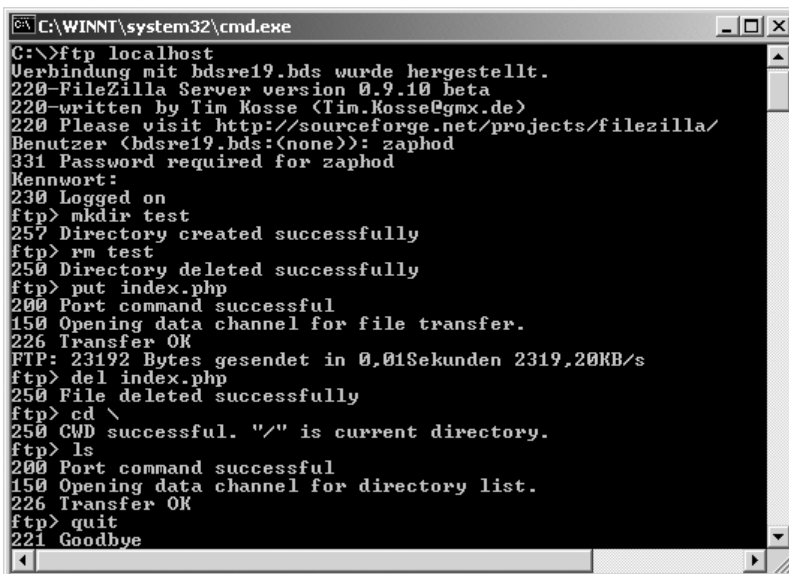
Abbildung 6.5: Das Heimatverzeichnis des neuen Benutzers *zaphod*

Ein Test mit dem neuen Benutzer ist Pflicht! Hierfür nehmen Sie einen von Ihnen favorisierten FTP-Client und überprüfen den Account. In diesem Beispiel benutzen wir das FTP-Programm auf der Eingabeaufforderung.

Beschreibung	FTP-Kommando
Erstellen eines Verzeichnisses test	mkdir test
Löschen des Verzeichnisses test	rm test
Upload einer Datei test.html	put test.html
Löschen einer Datei test.html	del test.html
Wechseln in die oberste Verzeichnisebene	cd /

Tabelle 6.3: Typische FTP-Befehle

Dem Benutzer *zaphod* ist es selbstverständlich nicht erlaubt, aus seinem Heimatverzeichnis `<xampp>\htdocs\zaphod` in andere nicht legitimierte Verzeichnisse zu wechseln! Das gilt es zu überprüfen.



```

C:\>ftp localhost
Verbindung mit bdsre19.bds wurde hergestellt.
220-FileZilla Server version 0.9.10 beta
220-written by Tim Kosse (Tim.Kosse@gmx.de)
220 Please visit http://sourceforge.net/projects/filezilla/
Benutzer (bdsre19.bds:(none)): zaphod
331 Password required for zaphod
Kennwort:
230 Logged on
ftp> mkdir test
257 Directory created successfully
ftp> rm test
250 Directory deleted successfully
ftp> put index.php
200 Port command successful
150 Opening data channel for file transfer.
226 Transfer OK
FTP: 23192 Bytes gesendet in 0,01Sekunden 2319,20KB/s
ftp> del index.php
250 File deleted successfully
ftp> cd \
250 CWD successful. "/" is current directory.
ftp> ls
200 Port command successful
150 Opening data channel for directory list.
226 Transfer OK
ftp> quit
221 Goodbye

```

Abbildung 6.6: Der Test für den neuen Benutzer mit dem FTP-Konsolen-Client

6.3 Anonymous FTP

Bei *Anonymous FTP* kann sich der festgelegte Benutzer *anonymous* mit einer beliebigen E-Mail-Adresse in einem hierfür freigegebenen Verzeichnis einloggen. Neuerdings finden sich immer häufiger FTP-Server im Netz, die den Anonymous-Bereich ohne Passwort freigeben. Im Prinzip ist das auch in diesem Fall dasselbe. Ob nun eine reale oder fiktive E-Mail-Adresse, eine beliebige @-Phrase oder kein Passwort: Anonymous FTP ist der öffentliche FTP-Bereich auf Ihrem Server!

Ein Anonymous-Bereich ist im XAMPP als Standard enthalten. Sie finden den Benutzer *anonymous* ohne Passwort im User-Bereich des FilaZilla-Interface. Das Heimatverzeichnis (*Homedir*) lautet `<xampp>\anonymous`. Der Upload-Bereich lautet `<xampp>\anonymous\incoming`. Die Rechte sind hier so gesetzt, dass der Benutzer im *anonymous*-Hauptverzeichnis lesen und im *incoming*-Unterverzeichnis lesen und schreiben darf. Natürlich können Sie hier Ihr eigenes *anonymous*-Verzeichnis definieren.

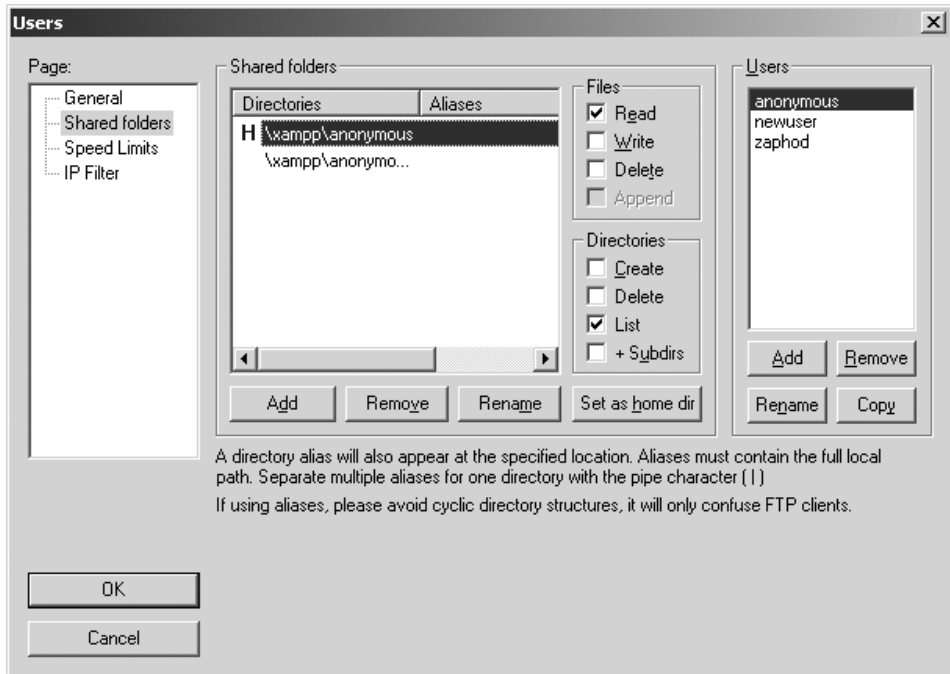


Abbildung 6.7: Das vorgegebene *anonymous*-Konto im XAMPP



Achtung

Der Upload-Bereich (*incoming*) ist ein mögliches Sicherheitsrisiko. Dadurch erhalten Sie auf Ihrem Rechner womöglich ungewünschte Fremdprogramme. Nicht weniger schlimm ist das absichtliche »Vollmüll« durch Dritte. Durch eine solche Attacke versucht der Angreifer, Ihre Festplatte bis zum Limit mit seinen Paketen zu belegen. Gerade Windows reagiert bei einer vollen Festplatte sensibel, häufig auch mit Systemabstürzen. XAMPP enthält zwar den *incoming*-Bereich als Beispiel. Es ist jedoch im produktiven Betrieb dringend empfohlen, diesen Bereich zu sichern oder gar ganz zu entfernen.

6.4 FTP-Beispiele mit PHP & Perl

Mit dem FileZilla Server funktionieren natürlich auch alle gängigen FTP-Funktionen in PHP. Kommen wir also gleich zu einem kleinen Beispiel für den Benutzer *zaphod* in PHP.

Listing 6.1: Das Skript ftp-mit-php.php

```
<?php
$host="localhost";
$user="zaphod";
$password="blabla";

// Herstellen der Basis-Verbindung
$conn_id = ftp_connect($host);

// Einloggen mit Benutzername und Kennwort
$login_result = ftp_login ($conn_id, $user, $password);

// Verbindung überprüfen
if ((!$conn_id) || (!$login_result)) {
    echo "Ftp-Verbindung nicht hergestellt!";
    echo "Verbindung mit $host als Benutzer $user nicht möglich";
    die;
} else {
    echo "Verbunden mit $host als Benutzer $user";
}

print "<br>";
$path="/";
$dir = "";
///// List
$files=ftp_nlist ($conn_id, "$path");
foreach ($files as $file) {
    $isfile = ftp_size ($conn_id, $file);
    if($isfile == "-1") { // Ist ein Verzeichnis
        print $file."/ [DIR]<br>";
    }
    else {
        $filelist[(count($filelist)+1)] = $file; // Ist eine Verzeichnis
        print $file."<br>";
    }
}

///// Upload
// ftp_put($conn_id, $remote_file_path, $local_file_path, FTP_BINARY);
```

6.5 Voraussetzungen für den externen FTP-Zugriff

```

///// Download
// ftp_get($conn_id, $local_file_path, $remote_file_path, FTP_BINARY);

///// Löschen einer Datei
// ftp_delete ($conn_id, $remote_file_path);

// Schließen des FTP-Streams
ftp_quit($conn_id);
?>

```

Solch ein Zugriff funktioniert natürlich auch mit Perl. Sie benutzen hierfür das *NET::FTP Modul*, das im *XAMPP-Perl-Add-On* enthalten ist. Für das folgende Beispiel loggen wir uns über den bereits vorhandenen Testbenutzer *newuser* mit dem Passwort *wampp* ein. Das Skript führen Sie im Verzeichnis `<xampp>\htdocs\modperl` aus.

Listing 6.2: Das Skript *ftp-mit-perl.pl*

```

use Net::FTP;
print "Content-Type: text/html\n\n";
$ftp = Net::FTP->new("localhost");      # Unser Host
die "Konnte keine Verbindung aufbauen $!" unless $ftp;
$ftp->login("newuser", "wampp"); # Hier Benutzername und Passwort eingeben
$ftp->get("index.php");
$ftp->quit;

```

Preisfrage: Und wo befindet sich die mit dem Perl-Skript soeben transferierte *index.php*? Im XAMPP-Hauptverzeichnis! Hätten Sie das gedacht?

6.5 Voraussetzungen für den externen FTP-Zugriff

Häufig treten dann Probleme mit dem Kontakt zu einem FTP-Server auf, wenn der Zugriff nicht vom lokalen Rechner aus erfolgt. Der Grund liegt in der Sicherheitsstruktur der Übertragung. Damit einem mutmaßlichen »Sniffer« das Leben schwer gemacht wird, wechselt der Server im aktiven Modus (*Active Mode*) bei jeder einzelnen Transaktion den Port. Hiefür kann ein FTP-Server die volle Anzahl verfügbarer Ports nutzen: 65535. Sie benutzen eine Firewall? Da haben wir schon das Problem. Denn im Prinzip müssen Sie nun alle Ports auf dem Server öffnen. Aber auch bei einem NAT-Router, der die Rechneradressen umschreibt, um eine gemeinsame Nutzung zu ermöglichen, gelangen die Anfragen nicht zum Ziel.

So schlimm ist das Ganze zum Glück nun auch nicht, denn jeder FTP-Server kann bei entsprechender Anfrage des FTP-Clients in den passiven Modus (*Passive Mode*) wechseln. Dabei baut der Client eine Verbindung über einen festgelegten Port auf, meist Ports jenseits der *Well Known Ports* (Port 0 bis 1023). Darüber läuft dann der gesamte Verkehr bis zum Beenden durch eine *quit*- oder *bye*-Anweisung bzw. infolge eines sog. Timeouts.

Für die Konfiguration des FileZilla Servers wie auch anderer FTP-Server hat das Aktiv/Passiv-Problem keine Bedeutung. Sie besitzen bereits die Fähigkeit zur automatischen Erkennung des vom Client vorgegebenen Modus. Aber Sie können die Spannweite der zur Verfügung stehenden Ports für den passiven Modus innerhalb des FileZilla Servers selbst definieren. Nutzen Sie hier über das FileZilla-Interface das Menü EDIT > SETTINGS > PASSIVE MODE.

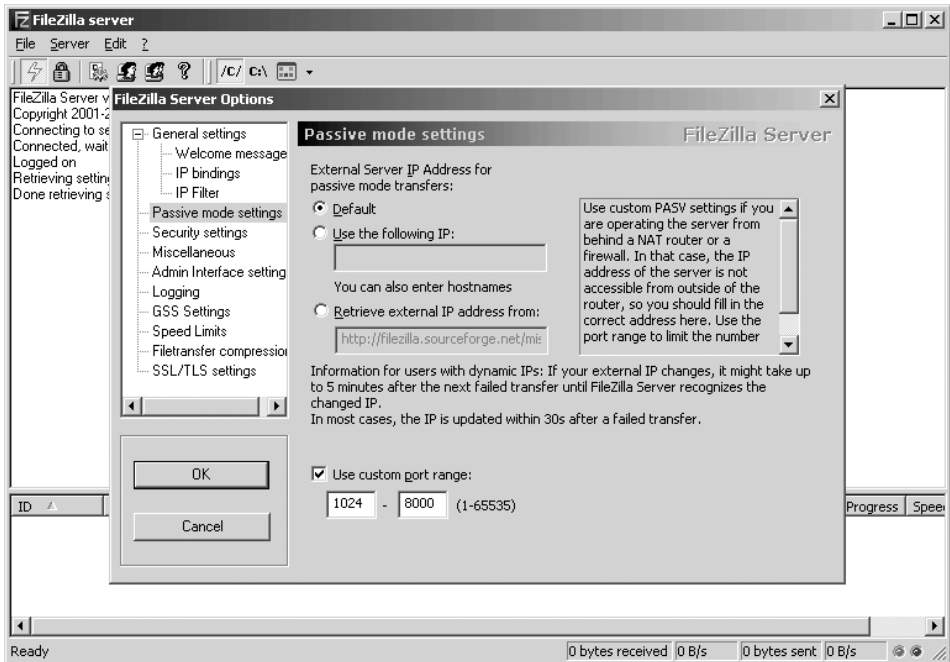


Abbildung 6.8: Die Festlegung der verfügbaren Ports im passiven Modus

Bei Problemen ist also die Überprüfung Ihrer eingesetzten Infrastruktur von entscheidender Bedeutung. Sie benutzen einen Rechner hinter einem Router oder Sie schützen sich mit einer Firewall? Das kann schon Grund sein für das Scheitern eines FTP-Zugriffs. Für Ihren Serverrechner mit FileZilla müssen für einen externen Zugriff – egal in welchem Modus – die folgenden Ports erreichbar sein:

Portnummer	Funktion
21	Port für die Kommandoübertragung
20	Port für die Datenübertragung
22	SSH-Port, FTP über SSH (nur bei SSH)
990	FTPS, FTP über SSL (nur bei SSL/TLS)

Tabelle 6.4: Ports, die üblicherweise von FTP verwendet werden

6.6 Secure-FTP mit FileZilla und SSL/TLS

Zu Beginn des Abschnitts gleich dieser Hinweis: Verschlüsselte Verbindungen unterstützen nicht alle FTP-Clients. Und nicht alle SSL-fähigen Clients verstehen sich mit dem eingesetzten FTP-Server. Ein ähnliches Client-Server-Problem gibt es übrigens auch bei verschlüsselten E-Mail-Verbindungen. Es liegt also nahe, gleich den FileZilla Client zu benutzen, der sich von Natur aus am besten mit dem FileZilla Server versteht.

Sie beginnen mit der Zertifizierung des Servers. Innerhalb des FileZilla-Interface wählen Sie im Menü EDIT > SETTINGS den Abschnitt *SSL/TLS settings* aus. Der Button *Generate new certificate* führt Sie schließlich zu dem Zertifizierungsfenster. Hier geben Sie Ihre Kontaktdaten, den Ländercode und Ihre Domain an. Für den Ländercode benutzen wir einen zweistelligen *ITU-T-Standard*, wie er als Ländervorwahl in der Telekommunikation Verwendung findet. Die Nummer 49 steht für Deutschland. Die USA haben übrigens den Code 01. Als Domain können Sie für einen ersten Test *localhost* eingeben. Zum Schluss speichern Sie über *Browse* Ihr neues Zertifikat in ein (möglichst sicheres) Verzeichnis auf Ihrem Rechner.

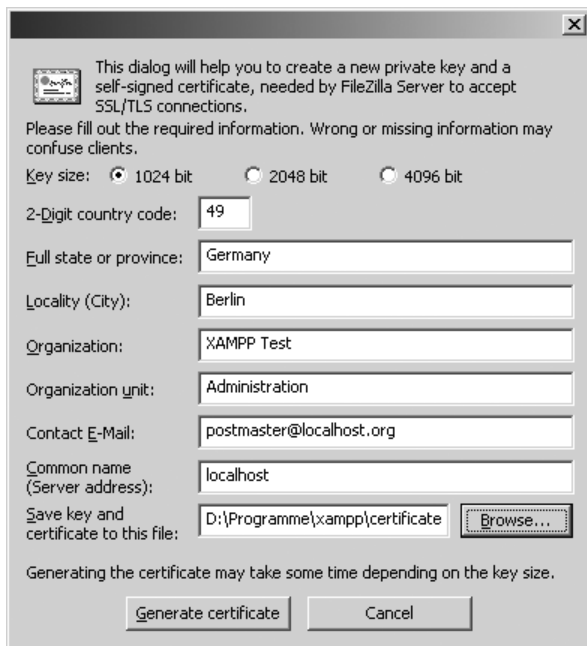


Abbildung 6.9: Ein neues Zertifikat im FileZilla-Interface erstellen

Mit dem neuen Zertifikat können Sie nun die SSL/TLS-Unterstützung im FileZilla Server aktivieren. Achten Sie aber darauf, dass auf Ihrem Rechner der Port 990 für SSL/TLS nicht blockiert ist.

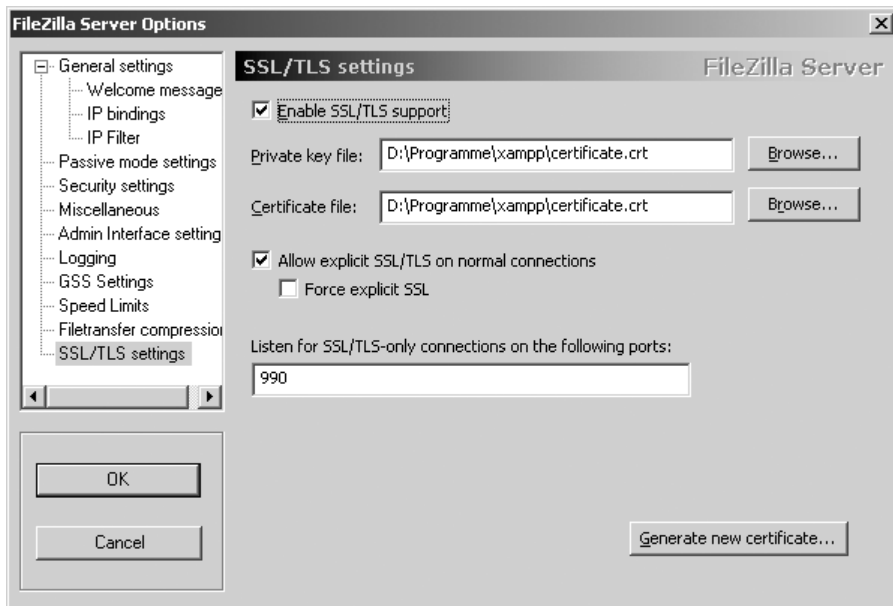


Abbildung 6.10: Die Aktivierung von SSL/TLS mit dem neuen Zertifikat

Jetzt brauchen Sie nur noch eine gesicherte Verbindung zum lokalen Server über den FileZilla Client herzustellen. Hierfür öffnen Sie DATEI > SEITENVERWALTUNG, um eine *Neue Seite* (d.h. neue Verbindung) einzurichten. Wichtig ist dabei die Einstellung des Ports auf die Nummer 990 statt des Ports 21 und des Servertyps *FPT über SSL/TLS (Implizierte Verschlüsselung)* statt des einfachen FTP.

Mit *Verbinden* testen Sie den Zugang. Wenn nun alles richtig eingestellt ist, werden Sie aufgefordert, das zuvor von Ihnen erstellte Zertifikat zu verifizieren. Wollen Sie in Zukunft diese Abfrage vermeiden, können Sie hier mit einem Häkchen Ihrem Zertifikat immer vertrauen.

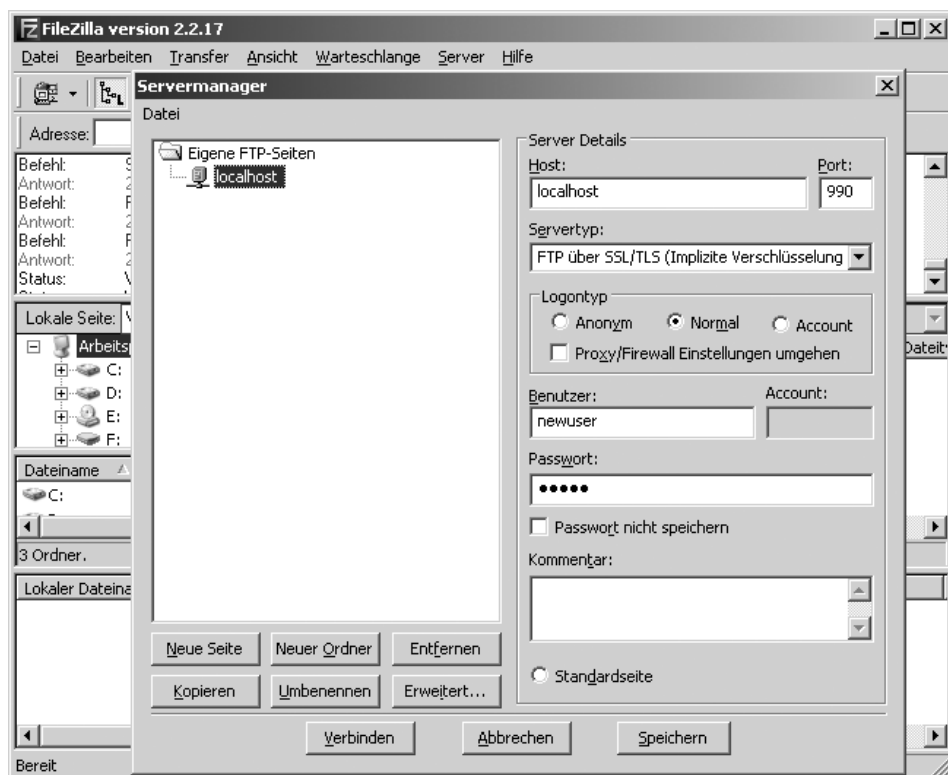


Abbildung 6.11: Die Einrichtung eines neuen FTP-Zugangs über SSL/TLS auf dem Client



7 XAMPP und E-Mail

Warum besitzt eigentlich nur der XAMPP unter Windows einen E-Mail-Server? Diese Frage stellen sich vielleicht viele der eher verwöhnten Linux-Anwender. Sie haben auf dieser Plattform eine üppige Auswahl von freien POP3-, IMAP- und SMTP-Servern. Und damit sind wir schon bei der Crux. Für die normalen Windows-Anwendersysteme gibt es entweder keinen oder nur einen äußerst eingeschränkten E-Mail-, genauer SMTP-Dienst. Allein dieser Umstand ist Grund genug, eine angemessene Alternative im XAMPP bereitzustellen: das *Mercury Mail Transport System*.

In diesem Kapitel geht es ausschließlich um XAMPP für Windows. Unter Linux gibt es zum Thema E-Mail wenig zu sagen: Dort ist der E-Mail-Server ein fester Bestandteil des Betriebssystems. Wenn das E-Mail-System unter Linux läuft, wird es auch von XAMPP aus funktionieren.

7.1 Windows: Mercury Mail Transport System

Das Mercury Mail Transport System ist übrigens das Werk des Programmierers Davis Harris. Sie verfügen damit über einen POP3-, SMTP- und IMAP-Server. Bitte verwechseln Sie das Mercury Mail Transport System nicht mit dem beliebten und ebenfalls frei erhältlichen E-Mail-Client Pegasus Mail vom gleichen Autor.

Den E-Mail-Server können Sie sich unter folgender URL herunterladen:

<http://www.pmail.com>

Weitere Informationen zum Mercury Mail Transport System finden Sie im deutschsprachigen Forum unter folgender Adresse:

<http://www.pmail32.de>

Das Mercury Mail Transport System in der Version 4.01 enthält so viele Features, dass eine Aufzählung den Rahmen dieses Kapitels sprengen würde. Für die Integration in das XAMPP-Paket sprachen die folgenden Argumente:

- Der SMTP-Server lässt sich mit der *sendmail*-Option in der *php.ini* ansprechen.
- Das System versteht relative Pfade und ist damit über einen USB-Stick transportabel.

- Das System läuft auch unter Windows 98.
- Das System bietet volle SSL-Unterstützung.

Aus diesen Gründen hat das Mercury Mail Transport System einen festen Platz in XAMPP.

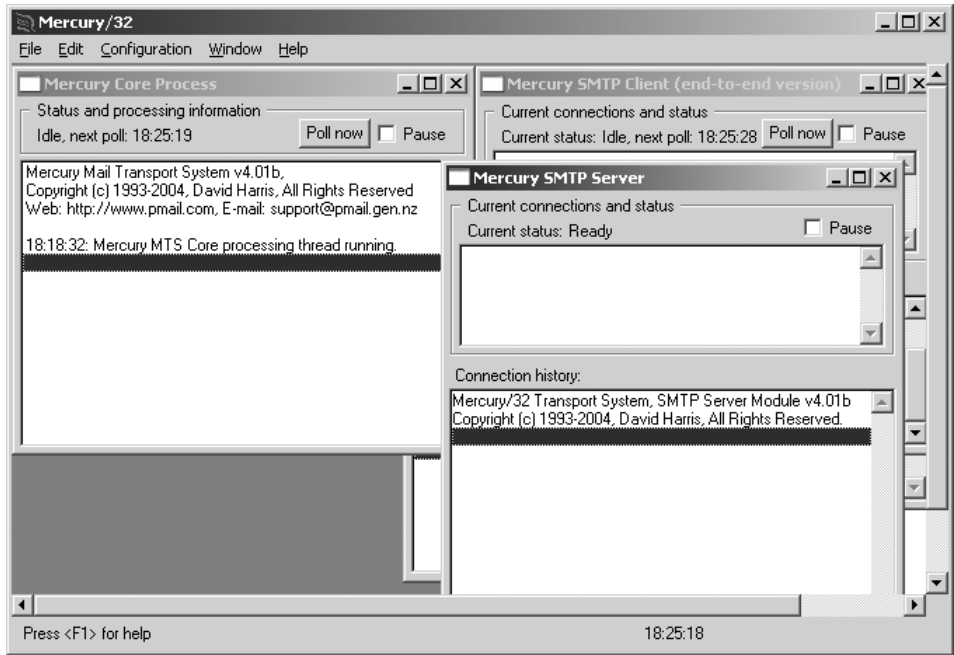


Abbildung 7.1: Das Mercury Mail Transport System als universeller E-Mail-Server

Beschreibung	Standort
Hauptprogramm	<xampp>\MercuryMail\mercury.exe
Konfigurationsdatei	<xampp>\MercuryMail\MERCURY.INI
POP3/IMAP-Postfächer	<xampp>\MercuryMail\MAIL\
Postausgangsordner	<xampp>\MercuryMail\QUEUE\

Tabelle 7.1: Mercury Mail Transport System im XAMPP

Einen kleinen Makel hat der Mercury Mail Server aber: Bisher fehlte eine Dienstinstallation für NT-Systeme. Sie müssen also den Mercury entweder per Hand starten oder Sie binden eine Verknüpfung der mercury.exe in den Autostart ein, um das Programm beim Systemstart automatisch zu laden.

In den folgenden Abschnitten werden wir auf wichtige Hinweise und Beispiele zur Verwendung des Mercury-Servers eingehen. Wir können jedoch lediglich einen Ausschnitt der Möglichkeiten dieses sehr umfangreichen *MTA* (Mail Transport Agent) darstellen.

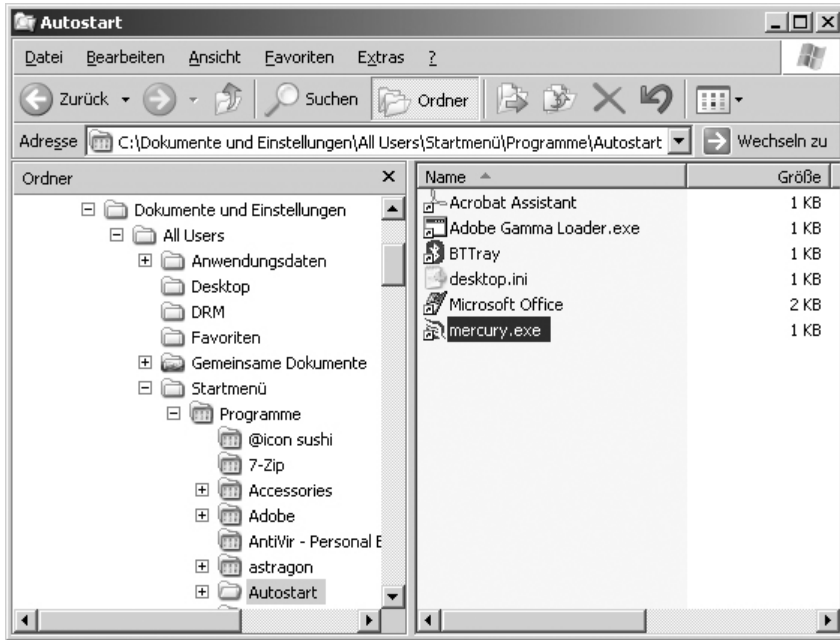


Abbildung 7.2: Eine Verknüpfung der mercury.exe im Startmenü

7.1.1 Eine Frage der Domain

Zunächst einmal stellt sich folgende Frage: Mit welcher Domain möchte ich senden und empfangen? Eigentlich stellt sich diese Frage für die meisten Anwender nicht. Aber vielleicht haben Sie einen ISP (Internet Service Provider) mit Standleitung, eine feste IP-Adresse und einen *FQDN* (Fully Qualified Domain Name). Dann benötigen Sie für einen eigenen Mailserver auf dem DNS-Server (DNS) Ihres ISP einen *MX*-Eintrag für Ihre Domain. Der sog. *MX Record* enthält die Adresse Ihres Mailservers. Damit wird die Zuordnung aller E-Mails Ihrer Domain gewährleistet. Und vielleicht soll es ja ein Mercury-System sein. Das heißt allerdings nicht automatisch, dass es sich auch dafür zuständig fühlt. Neben dem *MX*-Eintrag muss das Mercury Mail Transport System im Hinblick auf seine Zuständigkeit grundlegend konfiguriert werden. Im Abschnitt *CONFIGURATION > MERCURY CORE MODULE* können Sie unter dem Reiter *LOCAL DOMAINS* diese Zuständigkeit einstellen. Dabei ist *LOCAL HOST OR SERVER* der zuständige Rechner und *INTERNET NAME* der Domainname nach dem Muster *Second-Level-Domain.Top-Level-Domain* (also wie *apachefriends.org*).

Die Voreinstellung ist hier *localhost* mit den simultanen Subdomains *localhost.net*, *localhost.org* und *localhost.com*. Diese FQDNs sind notwendig, da viele E-Mail-Clients bei unqualifizierten Domainnamen den Versand verweigern.

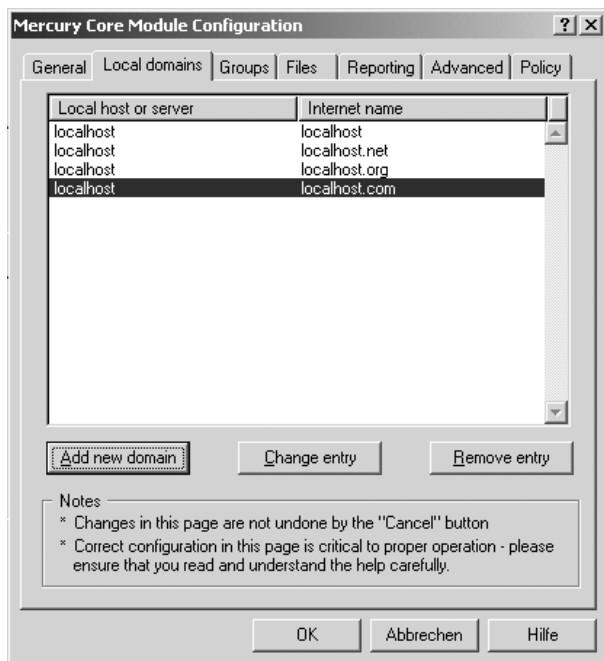


Abbildung 7.3: Die Beispieldomains im Mercury-System

Achtung



Die im XAMPP beispielhaft verwendeten qualifizierten Domainnamen wie *localhost.org* dürfen nicht nach außen verwendet werden. Sie dienen einzig der Simulation auf Ihrem lokalen Rechner. Alles andere stünde unter SPAM-Verdacht und ist nebenbei vom Autor des Mercury-Servers strengstens verboten! Es spricht jedoch nichts dagegen, eine lokale Entwicklungsumgebung auch mit fremden Domains zum Zwecke einer realistischen Simulation – beispielsweise eines eigenen programmierten Warenkorbs mit E-Mail-Benachrichtigung – einzurichten. Nur der E-Mail-Verkehr sollte in diesem Fall auf Ihren lokalen Rechner beschränkt sein.

Mit den folgenden Domains sind Sie immer auf der sicheren Seite:

example.com
example.net
example.org

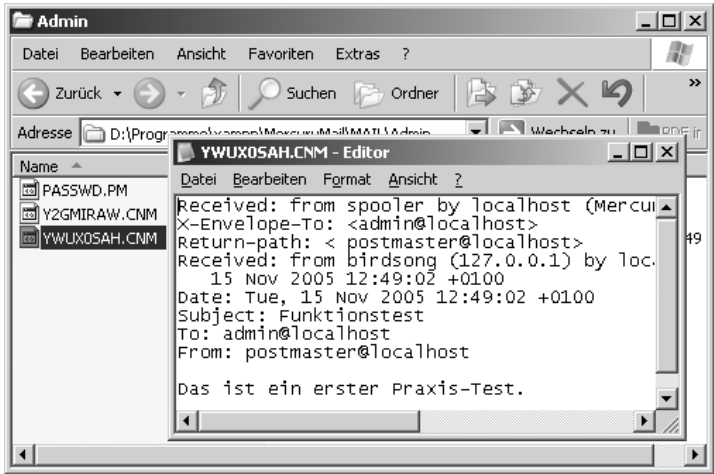


Abbildung 7.4: Direkter Zugriff auf eine empfangene E-Mail via Texteditor

Die Mailprotokolle dieser Domains wurden von der *Internet Assigned Numbers Authority* (IANA) zu Testzwecken reserviert. Damit darf nun jeder seinen E-Mail-Server mit einer fast vollwertigen FQDN nach RFC 2606 unbesorgt testen.

Diese Grundeinstellung im XAMPP reicht in den meisten Fällen aus, um beispielsweise E-Mail-Formulare für das Web auszutesten. Sie können es gleich probieren. Zu diesem Zweck enthält XAMPP ein kleines E-Mail-Formular in PHP.

Nach dem Versand finden Sie Ihre Mail im `MAIL`-Unterordner des Rezipienten, in diesem Fall im Verzeichnis des Benutzers `admin`. Sie können die E-Mail mit jedem beliebigen Texteditor öffnen und den Inhalt kontrollieren. Das reicht schon aus, um das Resultat eigener Versandskripte ohne E-Mail-Client schnellstmöglich zu überprüfen.

7.1.2 Postfächer und Aliase

Postfächer bzw. Benutzer können über das Mercury-Hauptprogramm (kurz: Mercury Console) unter `CONFIGURATION > MANAGE LOCAL USERS` erstellt, geändert oder gelöscht werden. Dabei finden Sie im XAMPP bereits die folgenden Benutzer vor:

Name	Empfänger	Passwort	Rechte
<i>Admin</i>	<i>Admin</i>	(keines)	Administrativ
<i>Postmaster</i>	<i>Admin</i>	(keines)	Administrativ
<i>newuser</i>	<i>newuser</i>	wampp	Benutzer

Tabelle 7.2: Die Standardbenutzer von Mercury unter XAMPP

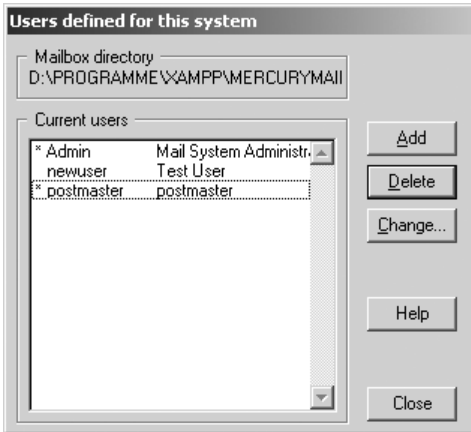


Abbildung 7.5: Bereits vorhandene Postfächer im Mercury Mail Transport System

Die Passwort-Dateien `PASSWD.PM` für die Benutzer befinden sich in den jeweiligen Postfach-Ordern unter `<xampp>\MercuryMail\MAIL`. Die Passwörter liegen hier im Klartext vor.

Achtung



Aus Gründen der Sicherheit ist auch hier die Vergabe von sicheren Passwörtern dringend zu empfehlen.

Aliase legen Sie über das Menü `CONFIGURATION > ALIAS` an. Sollte bei Ihnen dieser Menüpunkt fehlen, richten Sie bitte eine leere Datei namens `ALIAS.MER` im Verzeichnis `<xampp>\MercuryMail\Mercury` ein. Nachdem Sie das Mercury-System neu gestartet haben, sollten Sie den *Alias*-Eintrag im Hauptmenü `CONFIGURATION` finden.

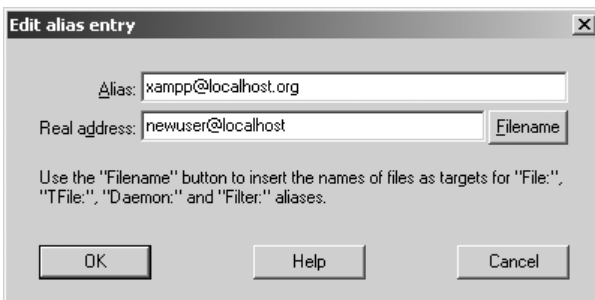


Abbildung 7.6: Ein neues Alias für `newuser@localhost` einrichten

7.1.3 SMTP: Beschränkungen oder Authentifizierung

Versenden darf nicht jeder. Und schon gar nicht mit Ihrem guten Namen. Ob also Ihr SMTP-Server von Dritten oder – wir wollen Ihnen selbstverständlich nichts unterstellen – vom eigenen administrativen Personenkreis missbraucht wird: SPAM bleibt SPAM. Und das hat der Autor des Mercury Mail Transport Systems ausdrücklich untersagt. Also bleibt die Frage: Wie soll der Postausgangsserver geschützt werden?

Dazu stehen Ihnen zwei Möglichkeiten zur Verfügung:

- eine Einschränkung über einen IP-Adressbereich
- eine Einschränkung über die SMTP-Authentifizierung

Als Standard im XAMPP dürfen nur Benutzer des lokalen Rechners (localhost) versenden. Das ist die einfachste und zugleich restriktivste Möglichkeit einer IP-Nummern-Beschränkung. Eine weitere Möglichkeit ist die Angabe einer Spannweite von IP-Nummern, die eine Berechtigung zum Versand erhalten. Hierfür rufen Sie in der Mercury Console das Menü CONFIGURATION > MERCURY'S SMTP SERVER auf. Unter dem Reiter CONNECTION CONTROL können Sie nun das sog. *Relaying* auf Ihr Netzwerk einschränken.

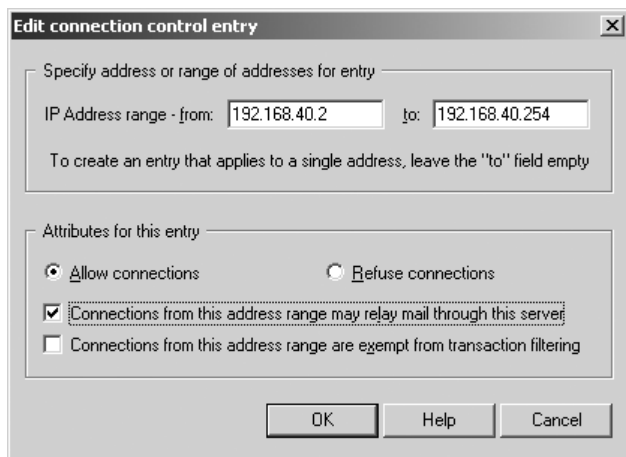


Abbildung 7.7: Eine festgelegte Spannweite von Adressen darf zugreifen.

Die andere Möglichkeit ist *SMTP-Authentifizierung*. Das Mercury Mail Transport System unterstützt zwei SMTP-Anmeldungsverfahren: Das LOGIN im Klartext und die verschlüsselte CRAM-MD5-Authentifizierung. Letztere wird allerdings nicht von allen E-Mail-Clients unterstützt und so beschränken wir uns hier auf die LOGIN-Methode im Klartext. Hierfür legen Sie eine Textdatei mit dem folgenden Inhalt an:

```
<Benutzer> <Passwort>
```

Beispiel:

newuser wampp
Postmaster geheim

Speichern Sie die Datei an einem sicheren Ort ab. Auf der Registerkarte CONNECTION CONTROL unter CONFIGURATION > MERCURY'S SMTP geben Sie den Pfad als AUTH PASSWORD FILE bekannt. Alle Häkchen haben Sie zuvor auf strikte SMTP-Authentifizierung gesetzt.

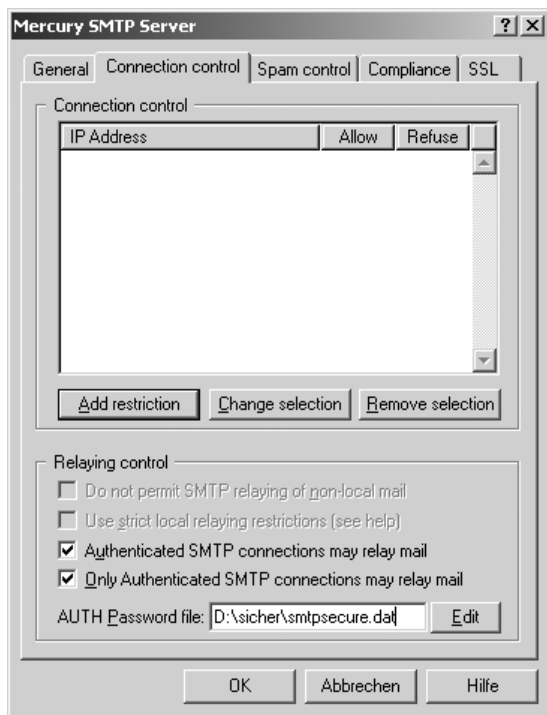


Abbildung 7.8: Eine Passwort-Datei für die Authentifizierung am SMTP-Server

Achtung



Mit strikter SMTP-Authentifizierung könnte das Versenden via PHP, Perl oder Java erheblich eingeschränkt sein. Die einfache `mail()`-Funktion, wie sie beispielsweise PHP bereitstellt, wird nun nicht mehr funktionieren. Vielfach werden allerdings für die einzelnen Interpretersprachen freie Klassen angeboten, die ein Login über Benutzer und Passwort einschließen.

7.1.4 Automatische Antworten

Das Mercury Mail Transport System unterscheidet in der Praxis den Ausdruck *Autoresponder* von dem Ausdruck *Automatische Antworten*.

Der Autoresponder bezeichnet hier eine automatisierte E-Mail-Antwort, die auf einem Alias ohne eigenes Postfach beruht. Das könnte beispielsweise *fag@domain.org* sein, wobei der Ausdruck »fag« weder Postfach noch normales Alias ist. Damit liest zwar niemand jemals die E-Mail an *fag@domain.org*. Aber dafür erhält der Absender eine automatisierte Antwort, worin vielleicht darauf hingewiesen wird, dass es sich hierbei um eine automatische Antwort handelt und wo der Interessent weiterführende Informationen erhält.

Die Einrichtung eines solchen Autoresponders erfolgt wieder im Mercury-Programm im Menü CONFIGURATION > ALAISE. Allerdings enthält die reale Adresse (REAL ADDRESS) nun einen Pfad zur Textdatei mit einem zum Versand bestimmten Inhalt. Eine schlichte Textantwort wird mit dem Attribut FILE: vor dem Pfad initialisiert.

FILE:d:\programme\xampp\mercury\faq.txt

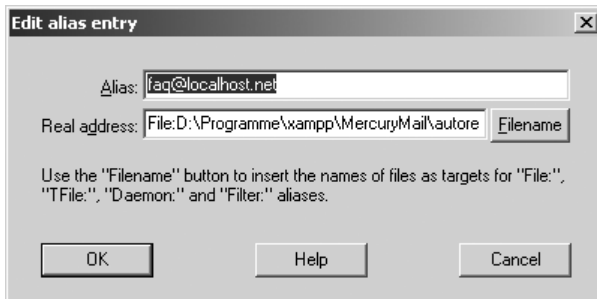


Abbildung 7.9: Ein neuer Autoresponder-Text für die FAQ-Adresse

Eine weitere Möglichkeit ist eine Antwort über eine *Template-Datei*, in der Sie auch *Header-* und *MIME-*Informationen eigenständig definieren können. Template-Dateien werden mit dem Attribut TFILE: gekennzeichnet. Wichtige Hinweise zu der hier verwendeten Syntax finden Sie in der Mercury Online-Hilfe.

TFILE:d:\programme\xampp\mercury\faq.mer

Ob nun im Urlaub oder wegen Krankheit: Viel häufiger als die eben vorgestellten Autoresponder-Methoden werden die bekannten und beliebten Abwesenheitsbenachrichtigungen (*Simple autoreplies*) Verwendung finden. Die Einrichtung im Mercury ist sehr einfach. Sie brauchen nur eine Textdatei mit dem Namen AREPLY.PM in dem Postfach-Ordner der betroffenen Adresse unterzubringen. Dort schreiben Sie Ihren Abwesenheitsvermerk hinein.

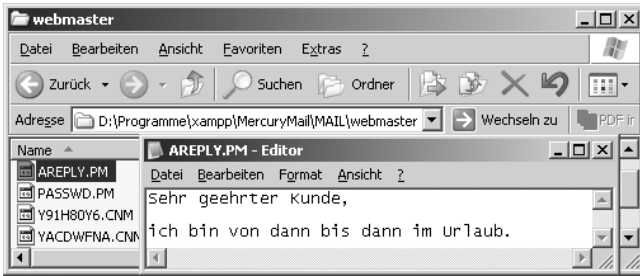


Abbildung 7.10: Eine einfache Abwesenheitsnachricht einrichten



Tipp

Sollten die *Autoreplies* nicht bei Ihnen auf Anhieb funktionieren, hilft meist ein Neustart des Mercury-Servers.

Für weitere Optionen wie die Festlegung der Gültigkeitsdauer der Abwesenheitsnachricht kann zusätzlich eine `AREPLY.CFG` in dem betreffenden Postfach erstellt werden. Die Definition dieser Datei können Sie der Mercury Online-Hilfe im Abschnitt *Programmed autoreplies* entnehmen. Ein Beispiel für eine `AREPLY.CFG` mit einer automatischen Antwort vom Typ 4 (Zeitraum), hier für die Dauer vom 10.05.2006 bis einschließlich dem 13.06.2006, stellt die folgende Zeile dar:

```
4, 20060510, 20060613, 0000, 2400, AREPLY.PM
```

7.2 Fake sendmail

Um einem möglichen Missverständnis von vornherein vorzubeugen, eine kurze Erklärung: *Fake sendmail* ist **nicht** *Sendmail*. Wie der Name bereits andeutet, ist es ein Tool, das für externe Programme wie PHP eine Schnittstelle bereithält, die wie *Sendmail* unter Linux ansprechbar ist. Doch danach schlägt die E-Mail einen ganz anderen Weg ein. Denn *Fake sendmail* sendet nicht selbst, sondern nutzt hierfür einen fremden SMTP-Server. Die Konfiguration hierfür erfolgt in der `sendmail.ini`.

Beschreibung	Standort
Hauptprogramm	<xampp>\sendmail\sendmail.exe
Konfigurationsdatei	<xampp>\sendmail\sendmail.ini

Tabelle 7.3: Die beiden Dateien des *Fake sendmail*

Dort stellen Sie Ihren realen SMTP-Server und falls notwendig die Authentifizierung ein. Folgende Authentifizierungsmethoden sind möglich:

- über den SMTP-Server mit Benutzer und Passwort
- SMTP nach POP3

Mittlerweile benutzen die Postausgangsserver häufig die erste Variante.



Hinweis

Fake sendmail hat doch so manche Vorteile gegenüber Mercury. Erstens braucht man keinen FQDN (Fully Qualified Domain Name). SPAM ist mit Mercury nicht erlaubt! Auch wenn Sie sich mit fremden Federn schmücken und mit falschen Domains nach außerhalb senden, laufen Sie Gefahr, mit Ihrer Adresse auf die sog. »Black Lists« zu geraten. So bleiben Sie aber sauber. Zweitens können Sie mit dieser – wenn auch – »gefakten« sendmail-Schnittstelle selbst Third-Party-Programme umgehen. Erfahrene Entwickler werden das zu schätzen wissen, wenn es darum geht, auf einer Linux-adäquaten Umgebung ihre Software zu testen. Ein durchaus nützliches Tool!

Definieren wir schnell eine beispielhafte Ausgangssituation mit folgenden Daten:

- Ihr Postausgangsserver (SMTP): `smtp.hoster.de`
- Ihr Benutzer: `zaphod`
- Ihr Passwort: `strenggeheim`
- Ihre E-Mail: `zaphod@meinedomain.de`

Die Konfiguration in der `sendmail.ini` lautet damit:

```
[sendmail]
smtp_server= smtp.hoster.de
auth_username= zaphod
auth_password= strenggeheim
```

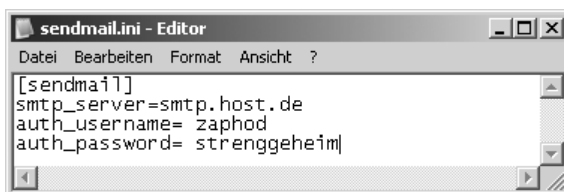


Abbildung 7.11: Ein Beispiel für eine einfache `sendmail.ini`

Das war es schon, denn mehr Informationen benötigt die `sendmail.ini` in diesem Fall nicht. Jetzt werden die Einstellungen getestet, indem Sie eine `test.mail` mit einem Texteditor verfassen. Diese könnte wie folgt aussehen:

```
From: zaphod@meinedomain.de
To: ziel@domain.de
Subject: Test von fake sendmail win32
```

Das ist ein Text für die Testmail.

Speichern Sie die Datei als `test.mail` direkt in den `sendmail`-Ordner. Dann prüfen Sie Ihre Einstellungen auf Kommandoebene.

```
sendmail -t < test.mail
```

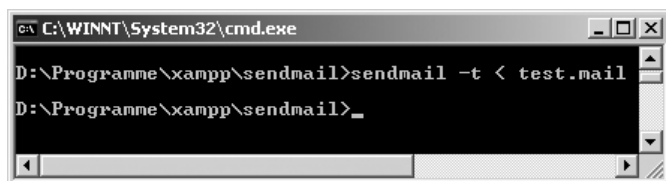


Abbildung 7.12: Eine Test-E-Mail auf der Kommandoebene mit Fake sendmail versenden

Wenn nun keine Fehlermeldung erfolgte, können Sie sich überlegen, *Fake sendmail* als »Standardversender« für PHP zu etablieren. Entkommentieren Sie hierfür die folgende Zeile in der `<xampp>\apache\bin\php.ini`:

```
sendmail_path = "D:\program files\xampp\sendmail\sendmail.exe -t"
```

Kommentieren Sie ebenfalls die folgenden Zeilen im `[mail function]`-Bereich mit Semikolon aus:

```
;SMTP = localhost
;smtp_port = 25
```

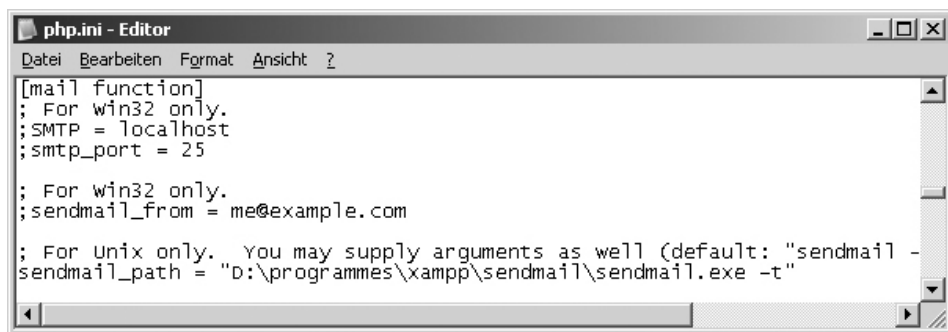


Abbildung 7.13: Die `php.ini` mit Fake sendmail für den Standardversand

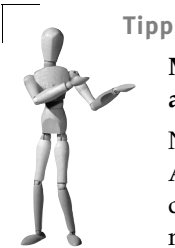
Damit ist die PHP-Unterstützung des *Mercury MTA* im XAMPP deaktiviert. Sie versenden nun bei jeder *PHP-mail()*-Anfrage mit *Fake sendmail*. Ein einfacher Test in PHP wäre das folgende Programm:

Listing 7.1: Ein einfaches PHP-Skript zum Verschicken einer E-Mail

```
<?php
$to      = 'ziel@domain.de';
$subject = 'Test mit PHP';
$message = 'Ein Test aus PHP';
$headers = 'From: zaphod@meinedomain.de' . "\r\n" .
    'Reply-To: zaphod@meinedomain.de' . "\r\n" .
    'X-Mailer: PHP/' . phpversion();

if (@mail($to, $subject, $message, $headers) )
{ echo "Versendet an $to"; }
else { echo "Fehler beim Versenden"; }

?>
```



Tipp

Mit Fake sendmail kann ich aber immer nur eine E-Mail-Adresse angeben!?

Nein, das ist nicht unbedingt richtig. Dazu ein Beispiel: Sie haben die Adresse *info@meinedomain.de*, möchten aber für Ihren Shop noch mit der Adresse *shop@meinedomain.de* über *Fake sendmail* versenden. Als mögliche Lösung bietet sich an: Sie richten bei Ihrem Anbieter weitere Aliase für Ihre Hauptadresse ein. In der Regel akzeptiert der SMTP-Server die Versendung der neuen Aliase mit den Zugangsdaten der Hauptadresse. Aliase werden übrigens auch häufig als E-Mail-Weiterleitungen bezeichnet.



8 XAMPP und XAMPP im Duett

Der XAMPP ist trotz aller unserer Bemühungen der Vereinheitlichung auf den verschiedenen Plattformen ungleich. Die Gründe liegen auf der Hand: Die Betriebssysteme sind unterschiedlich. Die verschiedenen Plattformen bieten dem Benutzer nicht nur eine Auswahl an verschiedener Software. Viel wichtiger ist die Bereitstellung unterschiedlicher Schnittstellen für die Kommunikation zwischen den Programmen, den sog. *Application Programming Interfaces (APIs)*. Um es kurz zu machen: Der Linux-XAMPP kann manches, was der Windows-XAMPP nicht kann, und umgekehrt. Und das kann schließlich der Grund sein, warum in einem bestimmten EDV-Szenario die gleichzeitige Nutzung zweier XAMPPs auf den unterschiedlichen Plattformen notwendig wird.

8.1 Das Front-/Backend-Szenario

Vielleicht kennen Sie ein solches Szenario? Ihre Mitarbeiter oder Kunden wollen eine Microsoft-Applikation ans weltweite Netz hängen. Sie sind aber strikt dagegen, argumentieren beispielsweise mit einer Überführung in *MySQL* und führen Sicherheits- und Performance-Gründe hierfür an. Serienbriefe mit MS Word funktionieren ja problemlos auch über den *MySQL-ODBC-Connector*. Eigentlich sind Ihre Argumente unschlagbar. Trotzdem: Ihre Appelle bleiben ungehört.

Eine weitere Lösung in diesem fiktiven Beispiel ist das Zusammenspiel von XAMPP: Linux quasi als Frontendserver und Windows im Backendbereich. Wobei die eigentliche Last der Apache unter Linux tragen soll. Der Apache unter Windows agiert dagegen nur für wenige ausgewählte COM-Anwendungen und soll dabei nicht als HTTP-Server nach außen erscheinen. Die Zielstellung in diesem Beispiel im Einzelnen:

- Der XAMPP unter Windows führt ein PHP-Programm auf einer ausgewählten Windows-Applikation durch, in diesem Beispiel wahlweise auf einer MS Access-Datenbank oder auf einer MS Excel-Datei.
- Der primäre Webserver unter Linux soll eine festgelegte URL auf den sekundären Webserver, hier einen Apache unter Windows, intern übergeben (via *RewriteRule*).
- Der Apache unter Linux soll darüber hinaus als vermittelnder Proxy die Antwort des Windows-Servers über seine Domain an den Absender weitergeben (*Reverse Proxy*).

In diesem Beispiel lautet die IP-Adresse des Linux-Servers 192.168.100.100, die IP-Adresse des Windows-Servers 192.168.100.50.

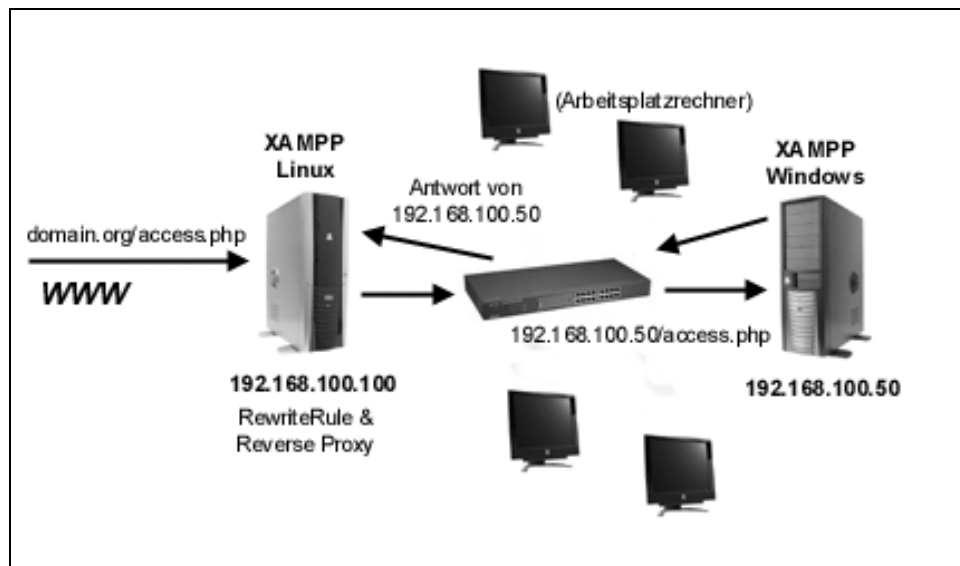


Abbildung 8.1: Die Netzwerktopologie des Beispielszenarios

8.2 PHP-Zugriff auf COM-Komponenten über XAMPP für Windows

Über das *Component Object Model* (COM) können Programmelemente von anderen Applikationen ausgeführt werden. Soweit die Kurzform. Konkret bedeutet das für Sie, dass Sie über die *COM-Funktionen* in PHP auf Elemente von Windows-Programmen zugreifen können. Die PHP-Referenz zu diesem Thema finden Sie unter folgender URL:

<http://www.php.net/manual/de/ref.com.php>

Hinweis



Nun müssen Sie nicht das Rad neu erfinden. Es existieren schon zahlreiche Programme, die viele Funktionen für Windows-Anwendungen bereithalten. Wir stellen Ihnen hier exemplarisch das *Lesen* und *Schreiben* in eine MS Access-Datei sowie das *Lesen* einer MS Excel-Datei vor.

8.2.1 MS ACCESS

Zu Beginn realisieren Sie den Zugriff auf MS Access mit PHP. Hierfür benötigen Sie die *ADODB-Bibliothek*, die im XAMPP für Windows im `php\pear`-Verzeichnis bereits enthalten ist. ADODB stellt eine Bibliothek für den PHP- oder Python-Zugriff auf zahlreiche Datenbanksysteme dar. Es ist konzeptionell an die ADO-Schnittstelle (ActiveX Data Objects) von Microsoft angelehnt. Vielfältige Hilfe erhalten Sie auf den ADODB-Seiten unter <http://adodb.sourceforge.net/#docs>. Eine informative deutschsprachige Dokumentation finden Sie darüber hinaus im Internet unter http://phpleverywhere.com/phpeverywhere/adodb_german.

ADODB Connector	Voraussetzungen	Plattform
<i>access</i>	ODBC	Nur Windows
<i>ado_access</i>	ADO- oder OLEDB-Provider	Nur Windows
<i>ado_mssql</i>	ADO- oder OLEDB-Provider	Nur Windows
<i>db2</i>	DB2 CLI/ODBC-Interface	
<i>vfp</i>	ODBC	Nur Windows
<i>firebird</i>	InterBase-Client	Unix und Windows
<i>borland_ibase</i>	InterBase-Client	Unix und Windows
<i>informix</i>	Informix-Client	Unix und Windows
<i>ldap</i>	LDAP-Extension	
<i>mssql</i>		
<i>mysql</i>	MySQL-Client	Unix und Windows
<i>oci805</i>	Oracle-Client	Unix und Windows
<i>odbc</i>		
<i>odbc_mssql</i>	ODBC	Unix und Windows
<i>odbc_oracle</i>	ODBC	Unix und Windows
<i>odbtp</i>	odbtp	Unix und Windows
<i>odbtp_unicode</i>	odbtp	Unix und Windows
<i>oracle</i>	Oracle-Client	Unix und Windows
<i>pdo</i>	PDO-Treiber	Unix und Windows
<i>postgres</i>	PostgreSQL-Client	Unix und Windows
<i>postgres64</i>	PostgreSQL-Client	Unix und Windows
<i>postgres7</i>	PostgreSQL-Client	Unix und Windows
<i>postgres8</i>	PostgreSQL-Client	Unix und Windows

Tabelle 8.1: Die Unterstützung der wichtigsten Datenbanktypen durch ADODB

ADOdb Connector	Voraussetzungen	Plattform
<i>sapdb</i>	SAP ODBC-Client	
<i>sqlanywhere</i>	SQL Anywhere ODBC-Client	
<i>sqlite</i>		
<i>sybase</i>	Sybase-Client	

Tabelle 8.1: Die Unterstützung der wichtigsten Datenbanktypen durch ADOdb (Forts.)

Sie benutzen als Beispieldatenbank am besten die deutsche Version der *northwind.mdb* mit dem Namen *nwind.mdb*. Es reicht auch die abgespeckte *fnpwind.mdb*, die nur die wichtigsten Tabellen enthält.

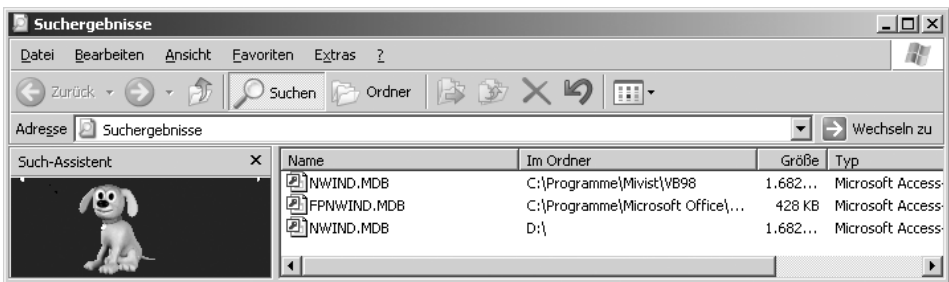


Abbildung 8.2: Manchmal müssen Sie die *nwind.mdb* erst suchen bzw. suchen lassen.

Über ADOdb senden Sie nun eine SELECT- und INSERT-Anweisung an Access. Kontaktiert wird die Datenbank ohne DSN (*Data Source Name*), da eine Authentifizierung hier nicht notwendig ist. In die Tabelle *Versandfirmen* fügen Sie eine neue Firma hinzu und geben anschließend alle eingetragenen Versandfirmen aus.

Listing 8.1: *phpadodb-msaccess.php*

```
<?
include('adodb/adodb-errorpear.inc.php');
include('adodb/adodb.inc.php');
include('adodb/tohtml.inc.php');

// Ihre Datenbank ist welche?
$accessdb="C:\\Programme\\Microsoft Office\\Office10\\1031\\fnpwind.mdb";
// CONNECT
$conn =& ADONewConnection('access');
$dsn = "Driver={Microsoft Access Driver (*.mdb)};Dbq=$accessdb;Uid=Admin;Pwd='';";
$conn->Connect($dsn);
// INSERT
$sql = "insert into Versandfirmen (Firma,Telefon) values ('Deutsche ↵
Post','018023333')";
```

8.2 PHP-Zugriff auf COM-Komponenten über XAMPP für Windows

```

if ($conn->Execute($sql) === false) {
    print 'error inserting: '.$conn->ErrorMsg().'<BR>'; }

// SELECT
$recordSet = $conn->Execute('select * from Versandfirmen');
if (!$recordSet)
    print $conn->ErrorMsg();
else
    while (!$recordSet->EOF) {
        print $recordSet->fields[0].' '.$recordSet->fields[1].'<BR>';
        $recordSet->MoveNext();
    }
$recordSet->Close();

// DISCONNECT
$conn->Close();
?>

```

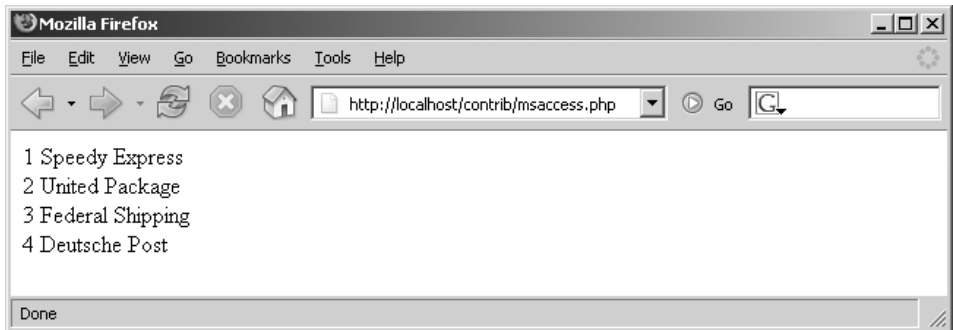


Abbildung 8.3: Die Ausgabe aller Versandfirmen aus der *mwind.mdb* in PHP mit ADOdb

8.2.2 MS EXCEL

Grundlage für das zweite Beispiel einer COM-Anwendung ist das Lesen in einem Excel-Sheet. Hierfür benutzen Sie die PHP-Funktionsbibliothek *Spreadsheet Excel Reader*. Der Reader ist zwar schon älteren Datums (Oktober 2004), machte aber bei unseren Tests einen soliden Eindruck.

PHP Excelreader:

<http://prdownloads.sourceforge.net/phpexcelreader/>

Im XAMPP ab Version 1.5.2 finden Sie diesen Reader standardmäßig im *pear*-Verzeichnis.

XAMPP für Windows:

```
<xampp>\php\pear\spreadsheet\Excel\Reader.php
```

Für das folgende Beispiel haben wir die *MySQL*-Tabelle *cds* über *phpMyAdmin* in eine *Microsoft Excel 2000*-Datei exportiert. Sollte es danach Schwierigkeiten mit dem Format geben, öffnen Sie bitte die *cds.xls* und speichern die Datei erneut als *cds.xls* ab. Damit sollte Ihnen ein »waschechtes« *Excel*-Format zur Verfügung stehen.

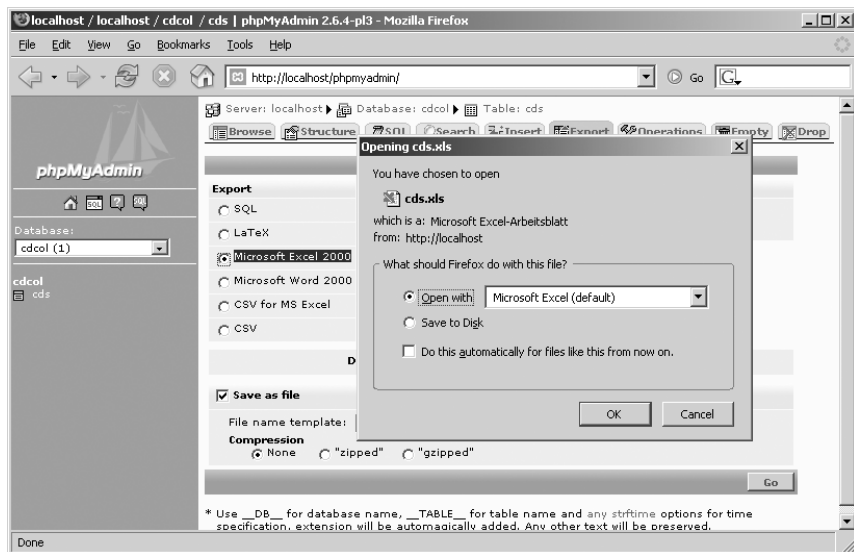


Abbildung 8.4: Die Tabelle *cds* in eine *Excel*-Datei über *phpMyAdmin* exportieren

Listing 8.2: *select-msexcel.php*

```
<html>
<head>
<title>PHP und Excel Reader</title>
</head>
<body>
<h1>PHP und Excel Reader</h1>
<table border="1">
<tr>
<th>Titel</th>
<th>Interpret</th>
<th>Jahr</th>
</tr>

<?php
require_once 'spreadsheet/Excel/reader.php';
$data = new Spreadsheet_Excel_Reader();
$data->setOutputEncoding('CP1251');
$data->read('cds.xls');
error_reporting(E_ALL ^ E_NOTICE);
```

```

for ($i = 1; $i <= $data->sheets[0]['numRows']; $i++) {
    for ($j = 1; $j <= $data->sheets[0]['numCols']; $j++) {
        echo "<tr>";
        echo "<td>".$data->sheets[0]['cells'][$i][$j++]."</td>";
        echo "<td>".$data->sheets[0]['cells'][$i][$j++]."</td>";
        echo "<td>".$data->sheets[0]['cells'][$i][$j++]."</td>";
        echo "</tr>";
    }
}
?>

</table>
</body>
</html>

```

Wesentliche Merkmale sind hier das *Include* für den Excel Reader, das Erstellen einer neuen Instanz sowie das Öffnen der Excel-Datei.

```

require_once 'spreadsheet/Excel/reader.php';
$data = new Spreadsheet_Excel_Reader();
$data->setOutputEncoding('CP1251');
$data->read('cds.xls');

```

Die Abfrage erfolgt nun über das *\$data*-Objekt.

Schemata zum Ansprechen eines Arbeitsblattes:

```
sheets[$a]
```

Beispiel für das erste Arbeitsblatt:

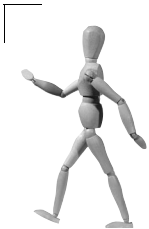
```
sheets[0]
```

Schemata zum Ansprechen einer Zelle eines Arbeitsblattes:

```
['cells'][$b][$c]
```

Beispiel für das zweite Arbeitsblatt, Zelle 1 C:

```
sheets[1]['cells'][1][3]
```



Und wie funktioniert das Schreiben in eine Excel-Datei?

Um es kurz zu machen: Wir haben keinen Weg gefunden, in eine vorhandene Excel-Datei zu schreiben. Zwar gibt es den schönen *Spreadsheet Excel Writer*. Ein Beispiel für die Funktionsweise ist im XAMPP bereits integriert. Dieser Writer erstellt bei jeder Anfrage ein neues Excel-Format und sendet schließlich den passenden HTTP-Header. In eine vorhandene Excel-Datei schreiben konnten wir damit jedoch nicht. Aber vielleicht haben wir ja nur den Wald vor lauter Bäumen nicht gesehen.

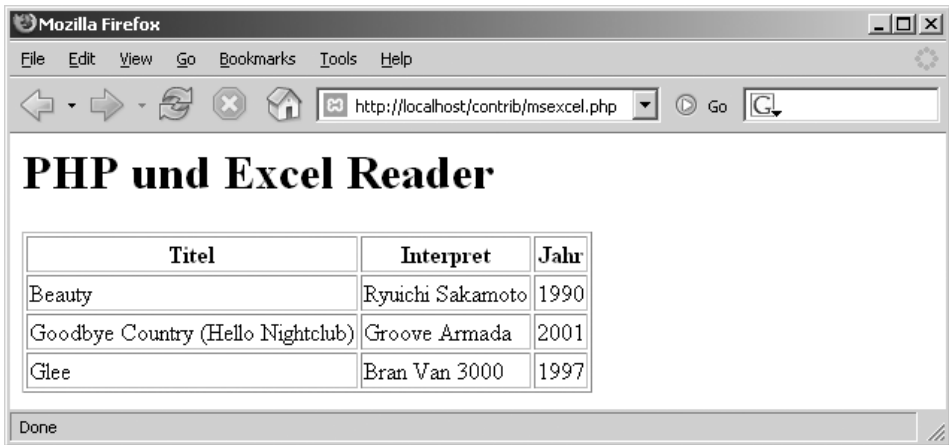


Abbildung 8.5: Der PHP Excel Reader im Einsatz

8.3 Umleiten mit RewriteRule und ReverseProxy

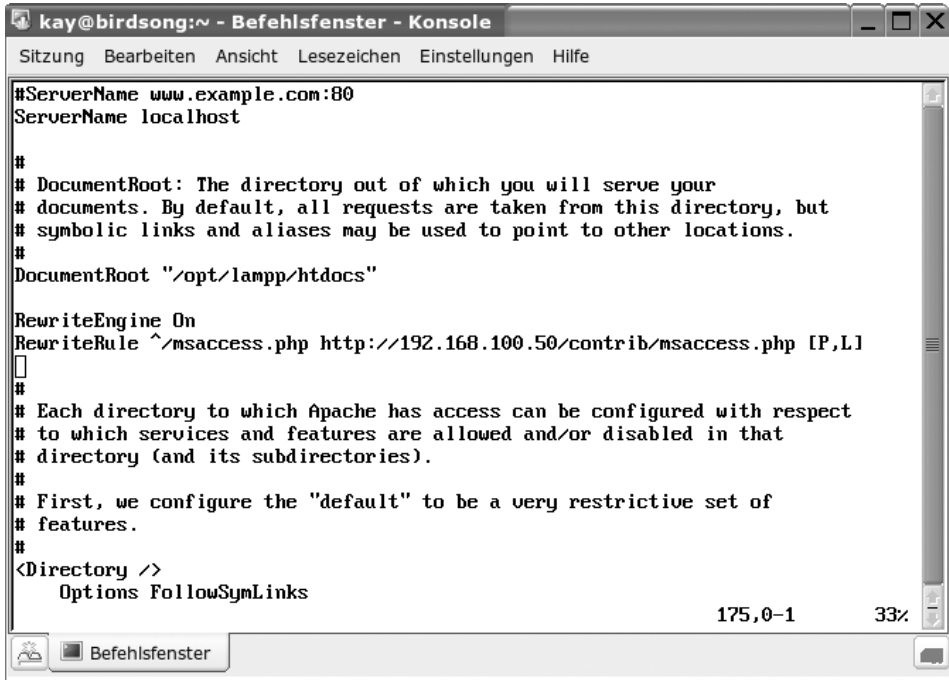
Jetzt haben Sie einen exklusiven Windows-Zugriff realisiert und wollen diesen über Ihren Linux-Server anbieten. Natürlich soll der Windows-Server selbst nicht direkt ans Netz. Und der Windows-XAMPP soll darüber hinaus als solcher nicht sichtbar werden. Das erreichen Sie im Apache mit einer URL-Weiterleitung über die Direktive *RewriteRule* und einem *Reverse Proxy*, der als vermittelnde Instanz zwischen der Anfrage (Browser) und dem Apache auf Ihrer Windows-Plattform agiert. Der Vorteil hierbei: Der anfragende Browser bekommt Ihren Windows-Server nicht direkt zu Gesicht. Folgende Module müssen Sie in Ihrem primären Apache-Daemon geladen haben: das *Rewrite-Modul*, das *Proxy-Modul* und das *Proxy-HTTP-Modul*.

Konfigurationsdatei: `/opt/lampp/etc/httpd.conf`

```
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
```

Nun soll jede Anfrage auf die Datei `msaccess.php` auf den dahinter gelagerten Windows-XAMPP (192.168.100.50) weitergeleitet werden. Das Flag [P] definiert dabei die Proxy-Funktion, das Flag [L] bestimmt das *RewriteRule*. Aus der Anfrage nach `http://www.domain.org/msaccess.php` wird damit real eine Anfrage auf Fehler! Hyperlink-Referenz ungültig., auch wenn das der Besucher der Site dank dem Proxy nicht sieht.

```
RewriteEngine On
RewriteRule ^/msaccess.php http://192.168.100.50/contrib/msaccess.php [P,L]
```



```
#ServerName www.example.com:80
ServerName localhost

#
# 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 "/opt/lampp/htdocs"

RewriteEngine On
RewriteRule ^/msaccess.php http://192.168.100.50/contrib/msaccess.php [P,L]

#
# 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
# features.
#
<Directory >>
    Options FollowSymLinks
```

Abbildung 8.6: Die RewriteRule-Anweisung in der httpd.conf des Linux-XAMPP

Alternativ können Sie auch ganze Verzeichnisse weiterleiten. Die folgende Regel leitet pauschal alle Anfragen auf das Verzeichnis `http://www.domain.org/extern` auf den Server mit der IP-Nummer 192.168.100.51 um.

```
RewriteRule ^/extern(.*) http://192.168.100.51 [P,L]
```



```
kay@birdsong:~> su
Password:
birdsong:/home/kay # vi /opt/lampp/etc/httpd.conf
birdsong:/home/kay # /opt/lampp/bin/apachectl restart
```

Abbildung 8.7: Den Apache-Restart nach jeder Änderung in der httpd.conf nicht vergessen!

**Tipp**

Selbstverständlich brauchen Sie nicht in jedem Fall eine *Standleitung*. Um einen internen Rechner anzusprechen, können Sie alternativ *dynamisches DNS* einsetzen. Bitte lesen Sie zu diesem Thema Kapitel 11, »Online mit DynDNS«.



9 Performance Tuning

Die Performance nicht aus den Augen zu verlieren ist eine sehr wichtige Aufgabe des Systemadministrators und auch des Programmierers. Dabei stehen den beiden unterschiedliche Mittel zur Verfügung:

Der Programmierer muss bereits während der Planung und Entwicklung darauf achten, dass sein Programm später möglichst schnell laufen kann. Und gerade hier können ihm sehr viele Fehler unterlaufen.

Verglichen damit sind die Möglichkeiten eines Administrators sehr gering. Ein Geschwindigkeitsgewinn von insgesamt 20 % ist in vielen Fällen schon das Maximum, das ein Admin durch geschickte Konfiguration herausholen kann. Allerdings ist es auch oft so, dass nur noch der Administrator ein langsames System retten kann. Wer ändert schon gerne Programmcode, der zwar langsam ist, aber funktioniert?

9.1 Gefühlter oder realer Performance-Gewinn?

Viele Tipps zum Performance Tuning gehen in Richtung Scharlatanerie und Aberglauben und wirken sich überhaupt nicht oder nur geringfügig aus. Das kommt zum einen daher, dass Administratoren von Natur aus sehr gerne an ihren Systemen herumbasteln und dann auch oft gefühlsmäßige Verbesserungen erzielen. Auch hier scheint es so etwas wie einen Placeboeffekt zu geben.

Zum anderen ist der Erfolg von Performance Tuning sehr schwer messbar. Ein einfacher Benchmark reicht meist nicht aus und wenn, dann liefern solche Tests so unterschiedliche Ergebnisreihen, dass sie kaum miteinander vergleichbar sind.

9.2 ApacheBench

Da alle Tuning-Tipps auf den folgenden Seiten direkt oder indirekt einen Webdienst betreffen, haben wir uns bei der Wahl des Benchmark-Tools für den *ApacheBench* entschieden. Er ist einfach zu handhaben und da er zum Apache-Paket gehört, ist er auch in XAMPP enthalten. Aber man findet ihn auch auf jedem anderen System, auf dem Apache installiert ist.



Mehr Web-Benchmarks

Andere Web-Benchmarks für Linux sind zum Beispiel *httpperf* (<http://www.hpl.hp.com/research/linux/httpperf/>) und *http_load* (http://www.acme.com/software/http_load/). Genau genommen machen sie aber auch nichts anderes als der ApacheBench.

Für Windows kann man das kostenlos erhältliche *Web Application Stress Tool* (<http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webstres.msp>) von Microsoft empfehlen. Damit ist es nicht nur möglich, einzelne URLs zu messen, sondern auch, eine klickende Benutzerbewegung über eine Website zu simulieren.

Web Application Stress - C:\Programme\Microsoft Web Application Stress Tool\WAS.mdb [Reports]

Page Summary

Page	Hits	TTFB Avg	TTLB Avg	Auth	Query
GET /	2	56.50	604.50	No	No
GET /de/	2	230.00	279.50	No	No
GET /af.css	2	235.50	236.00	No	No
GET /af.js	2	69.50	69.50	No	No
GET /search/client+navclient-a	2	113.00	445.50	No	No
GET /img/spacer.gif	2	123.00	123.00	No	No
GET /img/projects_on-de.jpg	2	68.50	69.00	No	No
GET /img/logo_on-de.jpg	2	72.00	72.00	No	No
GET /img/afstat_on-de-03.jpg	2	121.00	121.50	No	No
GET /img/teaa_on-de.jpg	2	112.00	112.00	No	No
GET /img/teaa_on-de.jpg	2	120.50	121.00	No	No
GET /img/extra_on-de.jpg	2	63.50	63.50	No	No
GET /img/press_on-de.jpg	2	98.50	98.50	No	No
GET /img/partner_on-de.jpg	2	67.00	67.00	No	No
GET /img/contact_on-de.jpg	2	73.00	74.00	No	No
GET /img/language-de.jpg	2	62.50	62.50	No	No
GET /img/link.gif	2	102.00	102.00	No	No
GET /cache/37fa879c0dbd844835c	2	62.50	63.50	No	No
GET /images/120.jpg	2	242.50	304.50	No	No
GET /cache/d45f1353744e204d59e	2	66.00	67.00	No	No
GET /img/afbullet.gif	2	64.00	64.00	No	No
GET /images/123.jpg	2	167.50	167.50	No	No
GET /images/123.jpg	2	74.00	74.00	No	No
GET /images/075.jpg	2	73.00	74.00	No	No
GET /cache/4cd7a36b3ae09e4d32	2	107.00	107.00	No	No
GET /img/linkblack.gif	2	57.50	58.50	No	No
GET /img/de_ie-x-click-but04.g	2	120.00	121.00	No	No
GET /img/aflogo.png	2	65.50	66.00	No	No
GET /img/news_on-de-05.jpg	2	114.50	114.50	No	No
GET /img/teaa_on-de-07.jpg	2	63.50	65.00	No	No
GET /img/projects_on-de-09.jp	2	64.50	64.50	No	No
GET /img/texts_on-de-11.jpg	2	75.50	76.50	No	No
GET /img/extra_on-de-13.jpg	2	66.00	67.00	No	No
GET /img/press_on-de-15.jpg	2	63.50	64.50	No	No

Abbildung 9.1: Das Web Application Stress Tool von Microsoft kann einen Klickverlauf über den IE aufzeichnen und hinterher im Benchmark als Testlauf verwenden.

Im Grunde misst der ApacheBench, wie lange eine Anfrage an den Webserver gedauert hat und wie viele Anfragen pro Sekunde vom Webserver beantwortet wurden. Im Gegensatz zu anderen Web-Benchmarks ist der ApacheBench nicht in der Lage, mehrere URLs gleichzeitig zu testen. Der ApacheBench kann auch nicht den Klickverlauf eines Benutzers simulieren. Wer diese Fähigkeiten benötigt, findet im Hinweis-Kasten »Mehr Web-Benchmarks« eine kleine Liste weiterer Benchmark-Tools.

Ein normaler Webzugriff eines Browsers auf eine Seite lädt ja nicht nur die Seite, sondern auch noch viele Bilder dieser Seite mit herunter. So gesehen sind die Ausgaben des ApacheBench nicht unbedingt realistisch.

Aber das ist in unserem Fall auch gar nicht wirklich notwendig, denn es geht uns ja nicht um die absoluten Werte, sondern um den Vergleich von Werten. Anders ausgedrückt lautet die Frage: Ist der Zugriff nach dem Tuning schneller als vorher?

Aufgerufen wird ApacheBench in der Kommandozeile mit dem Befehl `/opt/lampp/bin/ab`. Auf Windows-Plattformen benutzen Sie hierfür `<xampp>\apache\bin\ab.exe`.

Die beiden wichtigsten Optionen sind `-n` und `-c`:

- Mit `-n` wird die Anzahl der Anfragen, die an den Webserver gerichtet werden, festgesetzt. Ein typischer Wert ist 100.
- Mit `-c` wird bestimmt, wie viele von den mit `-n` festgesetzten Anfragen gleichzeitig gestellt werden sollen. Je mehr gleichzeitige Anfragen es gibt, desto höher ist natürlich die Last auf dem Server und umso langsamer kann eine einzelne Anfrage bearbeitet werden. Ein typischer Wert ist hier 10.

In der Praxis sollte der Wert aber in dem Rahmen liegen, der auch bei der realen Website zu erwarten ist.

Im folgenden Screenshot sehen Sie die Ausgabe einen ApacheBench-Aufrufs mit den oben genannten Einstellungen und als Messobjekt die Website von Apache Friends.

Achten Sie beim Vergleich von Werten immer darauf, dass die *Document Length* gleich bleibt. Es macht keinen Sinn, die Zeiten einer großen Seite mit den Zeiten einer kleinen Seite zu vergleichen. Die kleine Seite ist natürlich zwangsläufig schneller.

```
root@ubuntu:~ # /opt/lampp/bin/ab -n 100 -c 10 http://www.apachefriends.org/
This is ApacheBench, Version 2.0.41-dev <$Revision: 1.121.2.12 $> apache-2.0
Copyright (c) 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Copyright (c) 1998-2002 The Apache Software Foundation, http://www.apache.org/

Benchmarking www.apachefriends.org (be patient)...
```

Server Software:	Apache/1.3.33
Server Hostname:	www.apachefriends.org
Server Port:	80
Document Path:	/
Document Length:	290 bytes
Concurrency Level:	10
Time taken for tests:	3.119671 seconds
Complete requests:	100
Failed requests:	0
Write errors:	0
Non-2xx responses:	100
Total transferred:	60600 bytes
HTML transferred:	29000 bytes
Requests per second:	32.05 [#/sec] (mean)
Time per request:	311.967 [ms] (mean)

Abbildung 9.2: Die wichtigste Ausgabe in diesem Report ist der Wert der *Requests per second*: Also wie viele Anfragen konnten im Durchschnitt pro Sekunde beantwortet werden.

9.3 Testlabor

Alle in diesem Kapitel vorgestellten Tipps haben wir – sofern technisch möglich – in einem kleinen Testlabor vor der Aufnahme in dieses Buch getestet. Unser Testlabor bestand aus zwei Linux-Systemen (jeweils Pentium 4, 2.6 GHz, 256 Mbyte RAM), wobei auf dem einen Rechner der ApacheBench und auf dem anderen Rechner der Apache mit der jeweiligen Testkonfiguration lief.

Auf beiden Rechnern liefen – bis auf die üblichen System-Daemons wie *cron*, *syslog* etc. – keine weiteren Prozesse, die das Testergebnis irgendwie beeinflussen könnten. Beide Rechner waren über einen dedizierten Switch miteinander verbunden.

Allerdings ist die Beschaffenheit unseres Testszenarios unbedeutend, da wir ja keine absoluten Ergebnisse messen, sondern nur die Ergebnisse der einzelnen Testläufe untereinander vergleichen wollten.

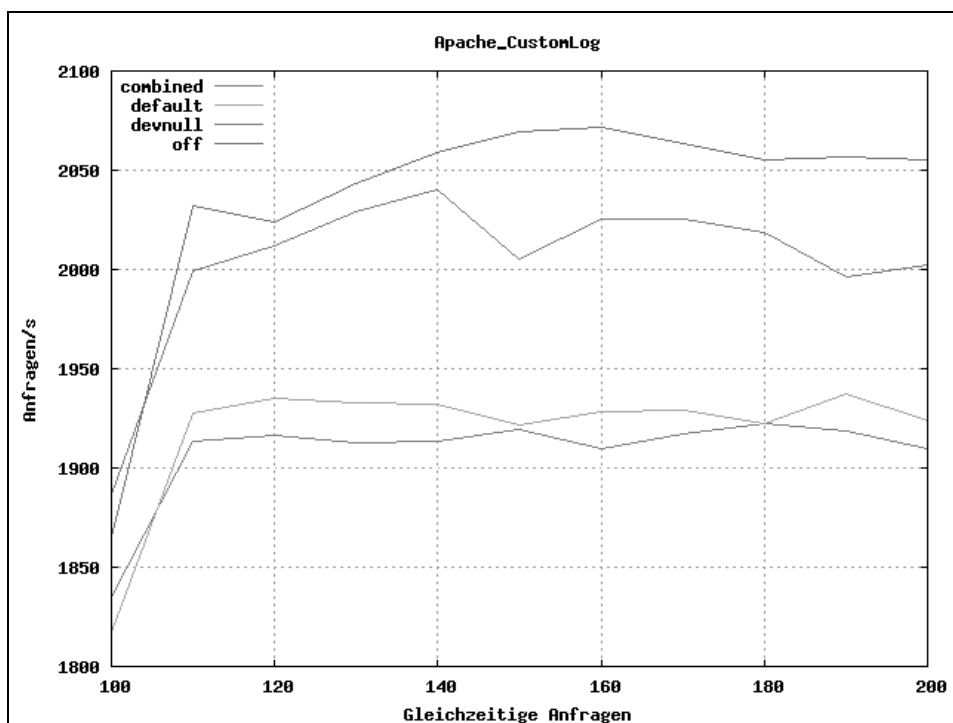


Abbildung 9.3: Ein schönes Positiv-Beispiel: Die unterschiedlichen Messreihen liegen klar auseinander und man erkennt deutlich, dass die eine Konfiguration der anderen überlegen bzw. unterlegen ist.

Das System war so weit automatisiert, dass der Messrechner den Testrechner fernsteuern und die verschiedenen Konfigurationen automatisch austauschen konnte. Die Messungen konnten wir so über Nacht oder übers Wochenende unbeaufsichtigt laufen lassen.

Eine einzelne Messreihe bestand aus 1 100 000 Messungen, wobei jeweils 100 000 Messungen für 100, 110, 120 usw. bis 200 gleichzeitige Anfragen durchgeführt wurden. Aus diesen Messungen wurde dann mittels Gnu-Plot ein Graph generiert, den wir nach Abschluss der Messungen sichtigten. Damit konnten wir dann entscheiden, ob eine bestimmte Konfiguration tatsächlich einen Performance-Gewinn brachte oder nicht.

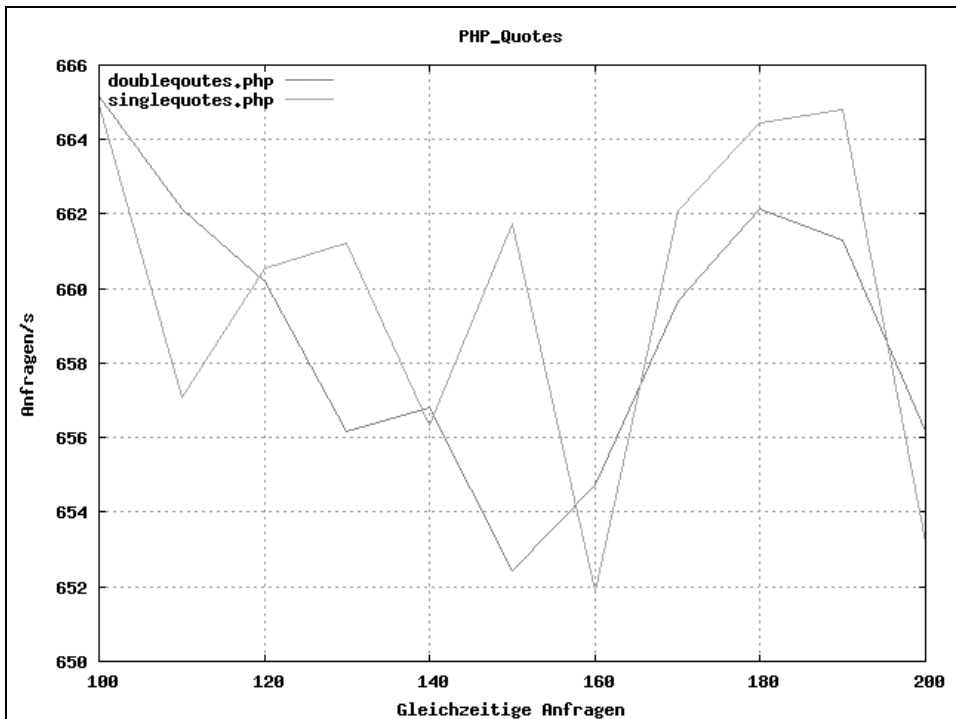


Abbildung 9.4: Hier sieht man ein typisches Negativ-Beispiel: Man kann sehen, dass mal die eine, mal die andere Konfiguration schneller ist. Es ist kein Vorteil der einen gegenüber der anderen Konfiguration festzustellen.

9.4 Apache

Der Apache ist das zentrale Organ eines XAMPP-Systems. Sämtliche Kommunikation zum Benutzer nach draußen geht ihren Weg über den Apache. Ist er zu langsam, wird er die gesamte Website ausbremsen.

Allerdings muss man auch sagen, dass in der Regel nicht der Apache dafür verantwortlich ist, dass ein System zu langsam läuft. Die großen Bremsen sind und bleiben einfach dynamisch generierte Seiten. Sie sind mindestens 10- bis 20-mal langsamer als statische Seiten.

Die Möglichkeiten, den Apache zu beschleunigen, liegen eher im Bereich zwischen 5 % und 20 % – also im Verhältnis zum Geschwindigkeitsverlust durch dynamisch generierte Webseiten ein relativ unbedeutender Wert. Da man nun aber auch nicht auf PHP oder Perl verzichten und auch keinen großen Eingriff in seine Architektur vornehmen möchte, bleiben oft nur diese Möglichkeiten übrig.

9.4.1 CustomLog

Auch wenn es im ersten Moment vielleicht ein bisschen absurd klingt, ist es trotzdem eine gute Möglichkeit, noch etwas Geschwindigkeit aus dem Apache herauszuholen: das Abschalten der Logfiles. Es ist sowohl für den Apache als auch für das System eine nicht zu unterschätzende Aufgabe, die Zugriffe zu protokollieren. Ein Abschalten dieser Funktion kann daher den Apache und auch das System entlasten.

Natürlich macht diese Konfiguration nur dann Sinn, wenn die Logdateien auch wirklich nicht benötigt werden. Das kann zum Beispiel in folgenden Szenarien der Fall sein:

- Die Webapplikation protokolliert selbst die Zugriffe.
- Es wird ein externer Webstatistikdienst verwendet (zum Beispiel Google Analytics).
- Der Apache befindet sich hinter einem Proxy (zur Lastverteilung oder zum Cachen von Inhalten) und würde sowieso nur die IP-Adresse des Proxys protokollieren.
- Über den Apache werden nur statische Daten wie zum Beispiel die Bilder einer Website angeboten und die Zugriffsdaten spielen keine Rolle.

Allerdings darf man auch nicht zu viel erwarten: Durch diese Konfiguration kann man lediglich einen Geschwindigkeitsvorteil von ca. 5 % erreichen. Allerdings dürfte dieser Wert bei Systemen mit besonders vielen Zugriffen auch höher liegen, da dort auch viel mehr in den Logfiles protokolliert wird als auf Systemen mit wenigen Zugriffen. Bei Letzteren wird diese Einstellung sicherlich kaum Vorteile bringen.

Die Konfiguration

In der `httpd.conf` werden einfach die Zeilen mit der `CustomLog`-Direktive bei den Virtual Hosts, bei denen die Logfiles nicht mehr benötigt werden, auskommentiert.

Aus

```
CustomLog logs/access_log common
```

wird

```
#CustomLog logs/access_log common
```

Wird schon die `access_log` nicht benötigt, dann gilt das auch für die entsprechenden `error_logs` und die `ErrorLog`-Direktiven können auch auskommentiert werden. Dies geschieht nur zur Abrundung der Konfiguration, einen Geschwindigkeitsvorteil kann man hier nicht erwarten. Dazu wird die `error_log` zu selten geschrieben.

9.4.2 HostnameLookups

Bei der Konfiguration *HostnameLookups on* ermittelt der Apache selbständig IP-Adressen in FQDN, um diese anstelle von IP-Adressen im Logfile zu protokollieren. Zwar ist bei XAMPP `HostnameLookups` standardmäßig ausgeschaltet, aber falls Sie diese Konfiguration selbst eingeschaltet haben, dann sollten Sie sich das vielleicht noch einmal überlegen.

Dieser Vorgang ist extrem aufwendig und bremst den Apache wie sonst keine andere Konfiguration. Ein Blick auf Abbildung 9.5 zeigt den fatalen Nebeneffekt dieser Konfiguration: Kann der Apache beim ausgeschalteten `HostnameLookups` knapp 2 000 Anfragen pro Sekunde bearbeiten, sind es beim aktivierten `HostnameLookups` nur noch um die 600 Anfragen pro Sekunde.

Wenn überhaupt, dann sollte diese Aufgabe vom Webstatistik-Tool übernommen werden.

Die Konfiguration

Gehen Sie einfach in die Konfigurationsdatei des Apache und ersetzen Sie die folgende Zeile

```
HostnameLookups On
```

durch

```
HostnameLookups Off
```

Aber wie bereits erwähnt: Standardmäßig ist `HostnameLookups` in XAMPP schon deaktiviert.

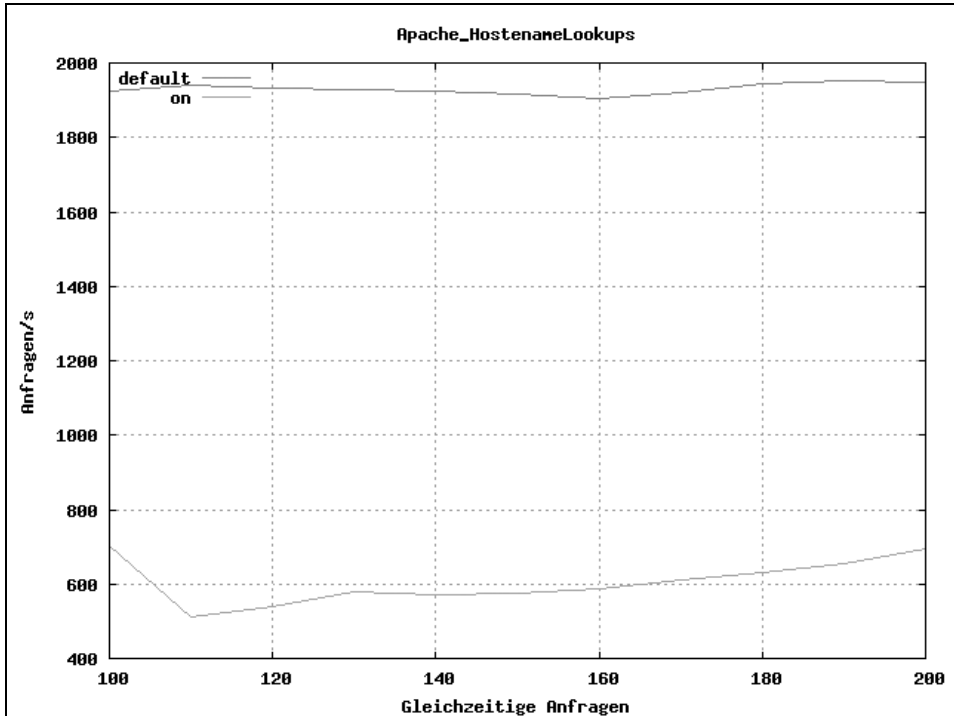


Abbildung 9.5: Ob `HostnameLookups` aktiviert oder deaktiviert ist, macht einen erheblichen Unterschied.

9.4.3 EnableMMap und EnableSendfile

Die Direktive *EnableMMap* aktiviert das sog. *memory-mapping* für Dateien, die vom Apache geöffnet werden. D. h., geöffnete Dateien werden in den Arbeitsspeicher abgebildet und dort vom Apache verarbeitet. Diese Methode kann schneller sein, muss es aber nicht in jedem Fall.

Mit der Direktive *EnableSendfile* kann der Apache veranlasst werden, die Inhalte von Dateien mittels der Kernel-Funktion *sendfile(2)* an den Browser durchzureichen. D. h., der Apache muss nicht selbst die Datei lesen, um sie weiterzureichen, sondern überlässt dies dem Kernel. Wie schon bei *EnableMMap* kann auch diese Methode schneller sein, was aber nicht zwangsläufig der Fall sein muss.

Die typischen Apache-Tuning-Tipps sind nicht so differenziert und versprechen durch beide Direktiven einen Geschwindigkeitsvorteil. In unseren Tests konnten wir aber keinen solchen messen. Eher im Gegenteil: Im Falle von *EnableSendfile* war der Apache sogar langsamer.

Erschwerend kommt noch hinzu, dass beide Direktiven die Stabilität des Apache gefährden können. Ein typisches Symptom ist das Fehlen von Grafiken in einer Webseite.

Aus diesen Gründen haben wir im XAMPP beide Features standardmäßig ausgeschaltet und empfehlen auch jedem, es dabei zu belassen.

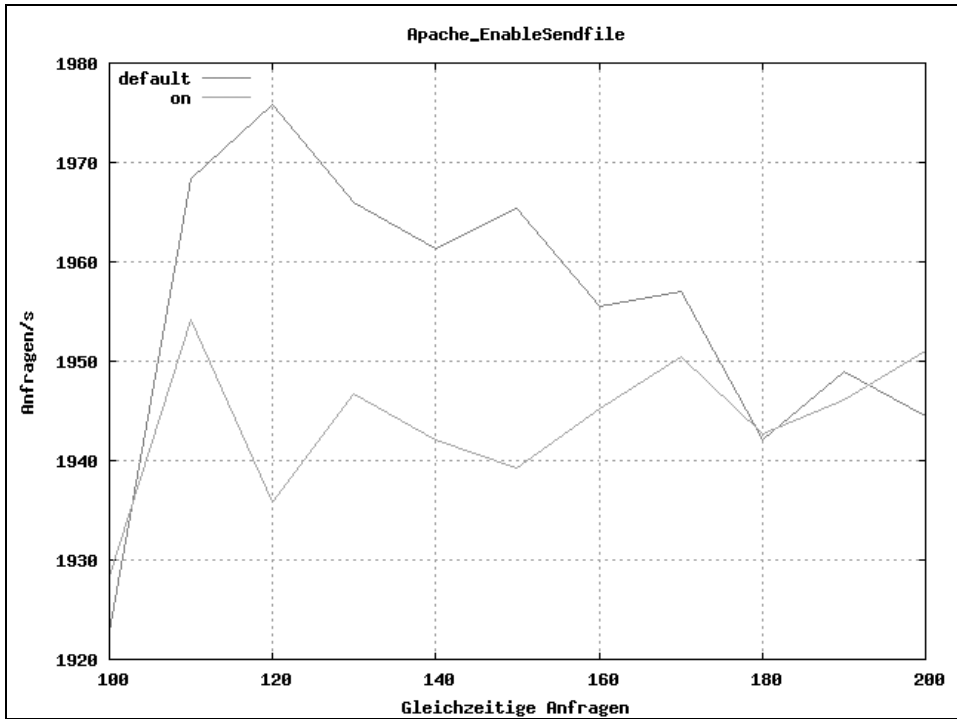


Abbildung 9.6: Bei eingeschaltetem `EnableSendfile` war unser Apache sogar langsamer.

9.4.4 DirectoryIndex

Mit der `DirectoryIndex`-Direktive wird im Apache eingestellt, welche Datei der Apache dem Surfer präsentieren soll, wenn dieser auf ein Verzeichnis des Webserver zugreift. Wenn Sie zum Beispiel auf `http://localhost/` zugreifen, dann wird Ihnen die `index.php` oder `index.html` aus dem `htdocs`-Verzeichnis gezeigt.

Die `DirectoryIndex`-Direktive bestimmt nun, welche Dateien das sind und in welcher Reihenfolge sie abgefragt werden sollen. Zum Beispiel könnte so definiert werden, ob zuerst nach der `index.html` oder zuerst nach der `index.php` gesucht werden soll.

In der Standardkonfiguration von XAMPP enthält die `DirectoryIndex`-Einstellung folgende Dateinamen:

```
index.html index.html.var index.php index.php3 index.php4
```

Das bedeutet, dass der Apache zunächst nachschaut, ob eine `index.html` in dem Verzeichnis ist. Wenn nicht, dann sucht der Apache nach einer `index.html.var`. Ist auch

diese nicht vorhanden, sucht er nach einer `index.php`. und so weiter. Jeder dieser Versuche, eine Datei zu finden, kostet den Apache Zeit und macht ihn so langsamer.

Die Dateinamen, die man niemals für die Index-Datei verwendet, brauchen in dieser Liste also gar nicht erst aufzutauchen. Und die Dateinamen, die man am häufigsten verwendet, sollten ganz am Anfang dieser Liste stehen.

Sollte die Standardkonfiguration für Sie ungünstig sein, dann können Sie mit einer optimierten Einstellung einen Geschwindigkeitsvorteil von ca. 9 % erzielen.

Die Konfiguration

Suchen Sie in der Apache-Konfigurationsdatei nach der *DirectoryIndex*-Direktive und ersetzen Sie diese durch eine auf Ihr System optimierte Dateireihenfolge.

Verwenden Sie zum Beispiel die `index.php` häufiger als die `index.html`, dann wird die folgende Konfiguration wahrscheinlich optimal für Sie sein:

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

Im umgekehrten Fall – Sie verwenden häufiger die `index.html` als die `index.php` – drehen Sie die Reihenfolge einfach um.

9.4.5 ServerTokens

Über die *ServerTokens*-Direktive wird eingestellt, wie der Apache sich im HTTP-Header zu erkennen geben soll. Die Standardeinstellung beim Apache und so auch beim XAMPP ist *Full* und damit wird der Apache angewiesen, möglichst viel Informationen über sich im Header unterzubringen.

Zum Beispiel sieht die Server-Zeile im HTTP-Header von XAMPP für Linux etwa so aus:

```
Server: Apache/2.2.0 (Unix) mod_ssl/2.2.0 OpenSSL/0.9.8a DAV/2  
        PHP/5.1.1 mod_apreq2-20050712/2.1.3-dev mod_perl/2.0.2  
        Perl/v5.8.7
```

Mit der Einstellung *ServerTokens Prod* wird diese Zeile nur noch reduziert dargestellt:

```
Server: Apache
```

Das Sammeln all dieser Informationen dauert natürlich auch etwas, und durch die Aktivierung der Einstellung *ServerTokens Prod* erzielen Sie einen leichten Geschwindigkeitsvorteil.

Die Konfiguration

Die *ServerTokens*-Direktive wird in der Standardeinstellung von XAMPP noch nicht verwendet. Fügen Sie daher einfach die folgende Zeile am Ende der Apache-Konfigurationsdatei hinzu:

ServerTokens Prod

Aber machen Sie sich nicht zu viel Hoffnungen: Etwa 1,5 % sind mit dieser Einstellung zu holen.

9.4.6 AllowOverride

Jeder, der schon mal ein wenig mit dem Apache gearbeitet hat, kennt die `.htaccess`-Datei. Über sie kann man bequem noch einige Änderungen an der Apache-Konfiguration vornehmen, ohne die zentrale Apache-Konfigurationsdatei ändern zu müssen – eine super bequeme Sache also.

Aber wie so oft verbirgt sich auch hier ein kleiner Geschwindigkeitsfresser: Der Apache muss nun bei jedem Zugriff nachschauen, ob in dem aktuellen Verzeichnis eine `.htaccess`-Datei liegt. Evtl. muss er auch in dem Verzeichnis darüber und dem Verzeichnis darüber nach einer `.htaccess` Ausschau halten.

Nehmen wir als Beispiel den Zugriff auf `http://localhost/a/b/c/d.html`:

1. Zunächst schaut der Apache im DocumentRoot-Verzeichnis nach, ob dort eine `.htaccess` liegt. Wenn ja, muss sie gelesen und verarbeitet werden.
2. Danach sucht er im Verzeichnis `a` die `.htaccess`. Existiert sie auch hier, dann wird sie eingelesen und verarbeitet.
3. Das Gleiche passiert nun mit dem Verzeichnis `b`.
4. Und zum Schluss wird noch im Verzeichnis `c` gesucht.

Der Apache muss also viermal die `.htaccess` suchen und sie evtl. auch viermal einlesen und verarbeiten. Dass dies ein nicht zu unterschätzender Aufwand ist, kann man sich ja denken.

In unserem Test waren wir großzügig und haben nur ein Szenario mit einem Verzeichnis und einer `.htaccess` getestet. Dabei haben wir einen Geschwindigkeitsgewinn von 13 % gemessen – und das nur bei *einem* Verzeichnis und *einer* `.htaccess`-Datei. Bei mehreren Verzeichnissen müsste entsprechend mehr herauszuholen sein.

Die Konfiguration

Suchen Sie in der Apache-Konfigurationsdatei nach `AllowOverride` und deaktivieren Sie diese Option mit `None`:

```
AllowOverride None
```

Aber Achtung: Dadurch verlieren alle `.htaccess`-Dateien ihre Bedeutung und Einstellungen, die man bisher in einer `.htaccess` vorgenommen hat, muss man nun in die zentrale Konfigurationsdatei des Apache übernehmen.

Benutzte man bisher keine `.htaccess`-Dateien, dann ist das Abschalten von `AllowOverride` natürlich noch problemloser. Allerdings wird man dann nur einen Geschwindigkeitsvorteil von ca. 2 % erreichen.

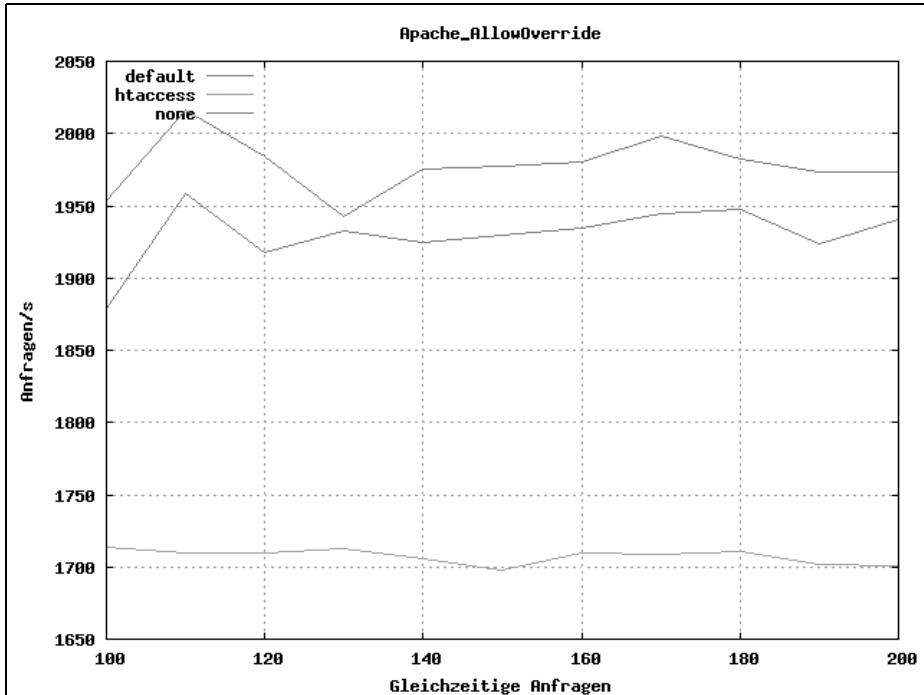


Abbildung 9.7: `AllowOverride None` ist eine sehr dankbare Einstellung.

9.4.7 LoadModule

Der Apache ist ein modularer Server, d. h., über dynamisch ladbare Module kann der Funktionsumfang des Webservers erweitert werden. Der Apache 2.2 hat etwa 70 solcher Module. In der Regel wird man nicht alle dieser Module wirklich brauchen, so dass diese quasi ungenutzt Hauptspeicher verbrauchen.

Das wäre ja auch nicht so schlimm, ein bisschen Hauptspeicher (und hier geht es um wenige Mbyte) kann man gerne verschwenden. Allerdings machen diese geladenen Module auch den Apache langsamer. Je nach Modul kann dies eine zusätzliche Last von 1 % bis 10 % im Apache bedeuten.

Beispiel SSL

Da der Apache die SSL-Funktionalität über ein Modul erhält, gehört diese zu den oben beschriebenen Fällen. Wegen der besonderen Bedeutung und Bekanntheit wollen wir diesen Punkt hier als Beispiel verwenden.

Unsere Messungen haben gezeigt, dass das SSL-Modul den Apache um etwa 7 % langsamer macht. Und eben auch dann, wenn HTTPS überhaupt nicht benutzt wird. Wenn man HTTPS also nicht benötigt, dann braucht man das SSL-Modul auch gar nicht erst zu laden.

Die Konfiguration

Suchen Sie in der Konfigurationsdatei des Apache die folgende Zeile:

```
LoadModule ssl_module modules/mod_ssl.so
```

Kommentieren Sie diese Zeile aus, indem Sie ein Doppelkreuz (#) davor setzen:

```
# LoadModule ssl_module modules/mod_ssl.so
```

Starten Sie nun den Apache neu, um die Änderung wirksam zu machen. Ihr Apache wird nun kein HTTPS mehr unterstützen, dafür aber schneller sein.

Genauso gehen Sie bei allen anderen Modulen vor, von denen Sie meinen, dass Sie sie nicht benötigen würden. Starten Sie am besten nach jeder Auskommentierung einer LoadModule-Zeile den Apache neu, um zu sehen, ob der Apache auch wirklich ohne dieses Modul auskommt.

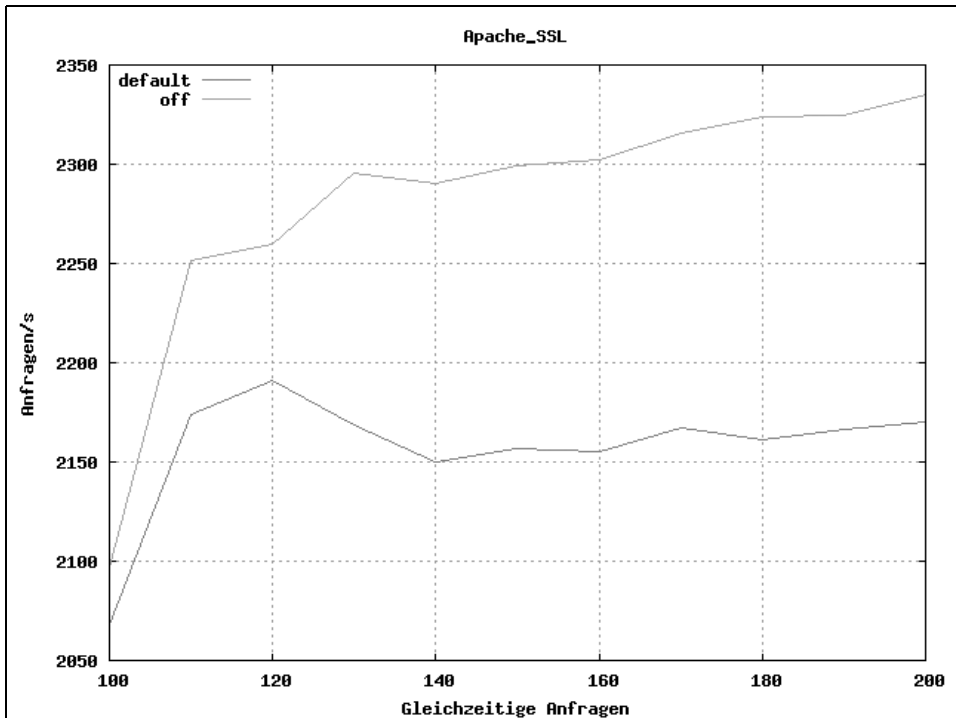


Abbildung 9.8: Auch wenn das SSL-Modul vom Apache überhaupt nicht benötigt wird, macht es diesen langsamer.

**Tipp****PHP deaktivieren**

Auch wenn es auf den ersten Blick absurd scheint, aber wenn Sie PHP nicht brauchen, dann deaktivieren Sie das PHP-Modul. Wenn Sie zum Beispiel einen Server haben, der nur für statische Daten verantwortlich ist, dann benötigt dieser Server PHP nicht. Auch so können Sie ca. 8 % an Geschwindigkeit gewinnen.

9.4.8 Optionen `FollowSymLinks` und `SymLinksIfOwnerMatch`

Mit diesen beiden Optionen wird im Apache eingestellt, wie dieser mit symbolischen Links umgehen soll. `FollowSymLinks` stellt ein, dass der Apache generell symbolischen Links folgen soll. `SymLinksIfOwnerMatch` verfügt, dass er nur dann einem Link folgt, wenn der Eigentümer der Link-Datei identisch ist mit dem Eigentümer der Datei, auf den der Link zeigt.

Beide Optionen haben die Eigenschaft, das System sicherer zu machen. Gerade wenn Sie andere Benutzer auf Ihrem System haben, die Inhalte in einem DocumentRoot-Verzeichnis (`htdocs`) ändern dürfen, dann sollten Sie diese Optionen erwägen.

**Hinweis****Um Missverständnisse zu vermeiden**

Die sicherste Konfiguration erreicht man, wenn `FollowSymLinks` ausgeschaltet ist. Bei der zweitsichersten Konfiguration ist `SymLinksIfOwnerMatch` eingeschaltet. Und die unsicherste Konfiguration erhält man, wenn `FollowSymLinks` eingeschaltet ist.

Lässt man die Sicherheitsaspekte außen vor, dann sind beide Optionen (in ihrer sicheren Einstellung, siehe Kasten) auch extreme Performance-Fresser. Im Grunde können sie nicht einmal etwas dafür – es liegt einfach in ihrer Natur.

Stellen Sie sich folgenden Zugriff auf Ihre Website vor: `http://localhost/a/b/c/d.html`, und gehen wir davon aus, dass Ihr DocumentRoot-Verzeichnis unter `/opt/lampp/htdocs` liegt. Der Apache muss nun folgende Überprüfungen vornehmen:

1. Ist `/opt` ein symbolischer Link? Wenn ja, dann Zugriff verweigern. Ansonsten weitermachen.
2. Ist `/opt/lampp` ein symbolischer Link? Wenn ja, dann Zugriff verweigern. Ansonsten weitermachen.
3. Ist `/opt/lampp/htdocs` ein symbolischer Link? Wenn ja, dann Zugriff verweigern. Ansonsten weitermachen.

4. Ist `/opt/lampp/htdocs/a` ein symbolischer Link? Wenn ja, dann Zugriff verweigern. Ansonsten weitermachen.
5. Und so weiter ...

Und das muss der Apache bei jedem Zugriff machen. Wir haben hier auch wieder einen sehr vorsichtigen Lasttest durchgeführt, der von einem direkten Zugriff auf das `htdocs`-Verzeichnis ausging. Der Apache musste also »nur« dreimal die Linküberprüfung durchführen. In diesem Szenario haben wir einen Geschwindigkeitsvorteil von 11 % gemessen.



Hinweis

Das Äquivalent zu *symbolischen Links* unter Linux sind *Verknüpfungen* von Verzeichnissen unter MS Windows.

Die Konfiguration

Öffnen Sie die Apache-Konfigurationsdatei und suchen Sie nach der *Options*-Direktive. Setzen Sie dort überall *FollowSymLinks* und vermeiden Sie *SymLinksIfOwnerMatch*.

Zum Beispiel:

Anstelle von:

```
Options -FollowSymLinks
Options SymLinksIfOwnerMatch
```

schreiben Sie:

```
Options +FollowSymLinks
Options -SymLinksIfOwnerMatch
```

Bei XAMPP ist `+FollowSymLinks/-SymLinksIfOwnerMatch` die Standardkonfiguration. Sofern Sie selbst nichts geändert haben, brauchen Sie also keine Änderung an der `httpd.conf` vorzunehmen.

9.4.9 Und sonst?

Neben all diesen Einstellungen, die man noch am Apache vornehmen kann, gibt es zwei Techniken, die sich in der Praxis immer bestens bewährt haben. Allerdings sind beide nicht ganz so einfach umzusetzen und der Erfolg ihres Einsatzes hängt auch ganz spezifisch von Ihrem eigenen System ab.

Caching von dynamischen Inhalten

Statische Inhalte sind ca. 10- bis 20-mal schneller als dynamisch generierte Inhalte. Bauen Sie daher Ihr System so um, dass es nach Möglichkeit alle dynamischen Inhalte in einem Cache legt und dieser dann über den Apache dem Surfer angeboten wird.

Wir verwenden diese Technik auch bei unserer Apache Friends-Website und konnten die Leistungsfähigkeit unseres Servers von ca. 25 Zugriffen/s auf 350 Zugriffe/s erhöhen.

Zweiter Server für statische Inhalte

Benutzen Sie einen zweiten Webserver für die statischen Inhalte Ihrer Website. Ganz besonders wenn es sich um Bilder handelt, die in Ihren Webseiten eingebunden sind. Durch diese Aufteilung erreichen Sie eine Lastverteilung auf zwei Systeme. In der Regel wird dadurch Ihr Webserver entlastet. Auf alle Fälle werden die Besucher Ihrer Website die Seiten nun schneller angezeigt bekommen.

Wie schon erwähnt, sind diese beiden Möglichkeiten nicht unbedingt einfach zu erreichen. Wer jetzt an die finanziellen Folgen eines zweiten Servers denkt, der sollte auch daran denken, dass die erste Möglichkeit auch Geld kostet. Zumindest Geld in Form von Zeit, die investiert werden muss, um das System entsprechend zu ändern.

9.5 PHP

Wenn es Ihnen darum geht, PHP zu beschleunigen, stehen Ihnen im Wesentlichen zwei Möglichkeiten zur Verfügung:

1. Komprimieren Sie die Ausgaben des PHP-Skripts und lassen Sie diese den Browser des Benutzers wieder dekomprimieren.
2. Benutzen Sie einen so genannten PHP-Cache, wie zum Beispiel den *eAccelerator*, der auch in XAMPP enthalten ist.

Wenn Sie selbst in PHP entwickeln, dann sollten Sie natürlich auch bereits beim Programmieren darauf achten, dass das Skript später besonders effektiv ablaufen kann.

9.5.1 Komprimieren mit ob_gzhandler

Eine gute Methode, Webseiten mit PHP zu beschleunigen, besteht darin, die Ausgaben – also die HTML-Seite, die mit PHP generiert wurde – komprimiert zum Browser des Benutzers zu übertragen. Auf den ersten Blick scheint dieser Tipp ein wenig fragwürdig zu sein, denn die Erfahrung zeigt, dass das Komprimieren von etwas nicht einfach so passiert, sondern ein Vorgang ist, der zusätzliche Last auf dem Server erzeugt.

Das ist richtig und auch in diesem Fall trifft es tatsächlich zu. Das Komprimieren erzeugt mehr Last auf dem Server, allerdings sind die Webseiten nun vom Umfang

her kleiner und gelangen so schneller zum Rechner des Surfers. D. h., die Website läuft intern langsamer, wirkt aber für den Surfer schneller. Und genau das möchte man ja erreichen.

Eine kleine Schwierigkeit ergibt sich aber noch beim Komprimieren der HTML-Inhalte: Nicht jeder Browser unterstützt komprimierte Inhalte und man muss darauf achten, diese Optimierung nur bei Browsern, die diese Funktion auch unterstützen, anzuwenden.

Zum Glück hilft uns hier PHP und man muss sich nicht selbst um das Komprimieren und die Browserfähigkeiten kümmern, sondern kann auf einen vorgefertigten Mechanismus von PHP zurückgreifen: den sog. *ob_gzhandler*.

Der *ob_gzhandler* macht genau das, was wir brauchen: Er sieht nach, ob der Browser des Benutzers das Komprimieren von Inhalten unterstützt, und wenn ja, dann sendet er an diesen Benutzer komprimierte Inhalte. Alle anderen Benutzer erhalten weiterhin die unkomprimierten Inhalte.

Global aktivieren

Um diese Funktion global zu aktivieren, also für wirklich alle PHP-Skripte auf Ihrem System, öffnen Sie die Konfigurationsdatei von PHP, die *php.ini*, und suchen folgende Zeile:

```
output_handler =
```

Ändern Sie die Zeile folgendermaßen:

```
output_handler = ob_gzhandler
```

Nun starten Sie den Apache neu, um die Änderung wirksam zu machen. Testen Sie anschließend, ob alle Ihre Skripte noch funktionieren. Sollte es Probleme geben, machen Sie die Änderung wieder rückgängig, starten den Apache wieder neu und aktivieren die Komprimierung nur bei den PHP-Skripten, bei denen sie auch tatsächlich funktioniert.

Im Einzelfall aktivieren

Es kann gut sein, dass die globale Einstellung, also einfach alles, was PHP generiert, zu komprimieren, nicht in allen Fällen funktioniert. In diesem Fall können Sie dann allerdings die Komprimierung individuell in den PHP-Skripten, bei denen Sie keine Probleme festgestellt haben, aktivieren.

Dazu schreiben Sie einfach den folgenden Befehl ganz an den Anfang eines PHP-Skripts:

```
ob_start("ob_gzhandler");
```

Wichtig ist, dass dieser Befehl aufgerufen wird, bevor PHP die erste Ausgabe gemacht hat.

**Tipp****Kombination**

Besonders gut eignet sich diese Methode auch in der Kombination mit dem Caching-Vorschlag aus dem Apache-Abschnitt. Cacht man die komprimierten Daten, dann müssen die Daten nicht bei jedem Zugriff neu komprimiert werden und der Benutzer erhält trotzdem komprimierte Inhalte. D. h., man kann die Vorteile des Komprimierens nutzen, ohne die Nachteile (eine höhere Last auf dem Server) zu haben.

9.5.2 eAccelerator

PHP ist eine Skriptsprache. Das bedeutet, dass PHP einfach so ausgeführt werden kann. Ein PHP-Programm muss nicht – wie zum Beispiel eine Compilersprache wie Java oder C# – vorher kompiliert werden, um es ausführen zu können. Man ändert einfach etwas am PHP-Code und schon kann man die Änderung online sehen.

Aber der erste Blick täuscht. PHP verhält sich zwar wie eine Skript- oder Interpretersprache, intern ist PHP aber schon eine Compilersprache. Nur wird der Compiler automatisch aufgerufen und arbeitet so schnell, dass man ihn gar nicht bemerkt.

Auch wenn es schnell geht, findet dieser Kompilierungsvorgang bei jedem Aufruf des PHP-Skripts erneut statt und das kostet Systemressourcen. Das ist ein Aufwand, der eigentlich nicht bei jedem Aufruf passieren müsste, sondern nur dann, wenn sich auch das zu kompilierende PHP-Programm tatsächlich verändert hat.

Und an dieser Stelle setzt ein PHP-Cache an. Er speichert das kompilierte PHP-Programm zwischen und sorgt dafür, dass ein PHP-Programm nur dann neu kompiliert werden muss, wenn sich dieses geändert hat. PHP-Caches sind wohl die effektivste Methode, PHP-Programme zu beschleunigen.

Der bekannteste Open-Source-PHP-Cache ist der *eAccelerator*. Je nach Größe und Aufbau Ihrer PHP-Programme kann man mittels eAccelerator eine Beschleunigung um den Faktor 4 bis 5 erreichen.

eAccelerator aktivieren

Da der eAccelerator bereits Bestandteil von XAMPP ist, fällt die Aktivierung besonders leicht aus. Sie müssen lediglich Ihre `php.ini` öffnen und dort zunächst die folgenden Zeilen suchen:

Listing 9.1: Ausschnitt aus der php.ini

```
;extension="eaccelerator.so"  
;eaccelerator.shm_size="16"  
;eaccelerator.cache_dir="/opt/lampp/tmp/eaccelerator"
```

```
;eaccelerator.enable="1"
;eaccelerator.optimizer="1"
;eaccelerator.check_mtime="1"
;eaccelerator.debug="0"
;eaccelerator.filter=""
;eaccelerator.shm_max="0"
;eaccelerator.shm_ttl="0"
;eaccelerator.shm_prune_period="0"
;eaccelerator.shm_only="0"
;eaccelerator.compress="1"
;eaccelerator.compress_level="9"
```

Entfernen Sie das Semikolon am Anfang einer jeden Zeile. Speichern Sie die `php.ini` und starten Sie dann den Apache neu.

Testen Sie nun bitte Ihre PHP-Programme, ob diese auch problemlos mit dem eAccelerator funktionieren.

9.6 MySQL

Ein weiteres Nadelöhr in einem AMP-System kann der MySQL-Server sein. Im Vergleich zu den anderen Bestandteilen ist allerdings das Optimieren vom MySQL viel, viel schwieriger. Da eine MySQL-Datenbank nicht so zustandslos wie zum Beispiel der Apache oder PHP ist, kann man hier nur in ganz wenigen Fällen mit einem Benchmark arbeiten, um Performance-Messungen vorzunehmen.

Oft wird empfohlen, den einen Wert so und den anderen Wert so zu setzen und einen ganz anderen Wert möglichst groß. Solche Ratschläge können Sie in der Regel gleich vergessen. So einfach ist das Tuning vom MySQL leider nicht. Schön wär's. Aber es ist definitiv komplizierter.

In der Regel wird man einen MySQL-Server über einen längeren Zeitraum beobachten bzw. laufen lassen müssen, bevor man eine sinnvolle Änderung an der Konfiguration vornehmen kann. Und eigentlich muss man nach einer Änderung den MySQL-Server erneut über einen längeren Zeitraum laufen lassen, um den Erfolg oder Misserfolg seiner neuen Einstellung überprüfen zu können.

9.6.1 Die richtige Konfigurationsdatei

Da das Optimieren einer MySQL-Installation so komplex und schwierig ist, werden beim MySQL-Server mehrere Variationen der Konfigurationsdatei mitgeliefert, die Sie bei der Optimierung von MySQL auf Ihrem System ein wenig unterstützen.

Je nach Größe Ihres Systems können Sie zwischen verschiedenen Voreinstellungen wählen:

`my-small.cnf`

Diese Konfigurationsdatei ist für Systeme optimiert, die nur 64 Mbyte Hauptspeicher haben. Nur bei wirklich sehr kleinen Datenbanken dürfte diese Konfiguration ausreichend sein.

`my-medium.cnf`

Diese Konfigurationsdatei ist für Systeme mit 128 Mbyte Hauptspeicher. Bei XAMPP ist diese Datei die Standardeinstellung. Sicherlich werden die meisten Rechner, auf denen XAMPP installiert wird, mehr als 128 Mbyte Hauptspeicher haben, jedoch darf man hier auch nicht vergessen, dass es in der Regel wohl keine Serversysteme sind und auch ein großer Teil des Hauptspeichers für die grafische Oberfläche des Betriebssystems und die Applikationen verwendet wird.

`my-large.cnf`

Diese Konfigurationsdatei ist für Systeme mit 512 Mbyte Hauptspeicher optimiert, auf denen hauptsächlich nur ein MySQL-Server läuft.

`my-huge.cnf`

Diese Konfigurationsdatei ist für Systeme mit 1 bis 2 Gbyte Hauptspeicher optimiert, auf denen hauptsächlich nur ein MySQL-Server läuft.

`my-innodb-heavy-4G.cnf`

Diese Konfigurationsdatei ist für Systeme mit 4 Gbyte Hauptspeicher optimiert, auf denen hauptsächlich nur ein MySQL-Server läuft, der ausschließlich InnoDB-Tabellen verwendet.

Unter der Linux-Version von XAMPP finden Sie alle diese Konfigurationsdateien im Verzeichnis `/opt/lampp/share/mysql/`. Unter Windows befinden sich diese Dateien im Verzeichnis `<xampp>\mysql\bin\`.

9.6.2 table_cache

Die Servereinstellung `table_cache` gibt an, wie viele Tabellen der MySQL-Server maximal auf einmal cachen kann. Über die MySQL-Konfigurationsdatei `my.cnf` (bzw. `my.ini` unter Windows) können Sie diesen Wert für Ihren MySQL-Server einstellen. Optimalerweise sollte dieser Wert nicht zu klein und auch nicht zu groß für die Anforderungen an Ihre Datenbank gewählt sein.

Den optimalen Wert zu finden ist ein längerer Prozess. Alle Informationen, die Sie benötigen, können Sie über den MySQL-Monitor erhalten. Rufen Sie also zunächst diesen auf:

Linux: `/opt/lamp/bin/mysql -uroot -pgeheim`

Windows: `<xampp>\mysql\bin\mysql -u root -p geheim`

Stellen Sie im MySQL-Monitor fest, wie lange Ihr MySQL-Server bisher lief:

```
mysql> SHOW STATUS LIKE 'uptime';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Uptime        | 2246260 |
+-----+-----+
1 row in set (0.00 sec)
```

Der Wert 2246260 gibt die Laufzeit in Sekunden an. In diesem Fall läuft die MySQL-Datenbank also bereits seit 25 Tagen – ein Zeitraum, der für verlässliche Werte ausreichen sollte. Wie groß dieser Zeitraum genau sein muss, hängt von Ihrem System ab. Innerhalb dieses Zeitraums sollte Ihr System mindestens ein paar Mal stark unter Last geraten sein. Ist das nicht passiert, dann können Sie aus den folgenden Werten kaum eine verlässliche Aussage treffen.

Schauen Sie sich jetzt die bestehende Konfiguration an:

```
mysql> SHOW VARIABLES LIKE 'table_cache';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| table_cache   | 256   |
+-----+-----+
1 row in set (0.00 sec)
```

Der `table_cache` ist also auf eine Größe von 256 gesetzt.

Sehen Sie nun nach, wie viele Tabellen im Moment geöffnet sind. Schauen Sie sich diesen Wert am besten an, wenn Ihr System sich unter Last befindet.

```
mysql> SHOW STATUS LIKE 'Open_tables';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Open_tables   | 256   |
+-----+-----+
1 row in set (0.00 sec)
```

Zurzeit sind also 256 Tabellen geöffnet. Der Table Cache wird also voll ausgenutzt. Anhand dieses Werts kann schon einmal folgende Aussage getroffen werden: Die `table_cache`-Einstellung ist nicht zu groß gewählt.

Wäre der Wert von `open_tables` wesentlich kleiner als der Wert von `table_cache`, dann kann man stark vermuten, dass der `table_cache` zu groß gewählt wurde und der Wert reduziert werden sollte. Diese Entscheidung hängt aber auch noch von dem nächsten Wert ab.

Ermitteln Sie daher noch den `opened_tables`-Wert:

```
mysql> SHOW STATUS LIKE 'Opened_tables';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Opened_tables | 294   |
+-----+-----+
1 row in set (0.00 sec)
```

Dieser Wert gibt an, wie oft Tabellen in der bisherigen Laufzeit von MySQL geöffnet wurden. Je größer dieser Wert über den bisher gemessenen Werten liegt, desto schlechter. In diesem Fall ist allerdings 294 zwar größer als 256, da der Server aber schon 25 Tage läuft, ist dieser Wert durchaus noch normal.

Wäre er um ein Vielfaches größer als `open_tables` oder `table_cache`, dann wäre `table_cache` auf jeden Fall zu klein gewählt und Sie sollten den Wert erhöhen.

9.6.3 max_connections

Die Servereinstellung `max_connections` gibt an, wie viele Verbindungen der MySQL-Server maximal gleichzeitig bedienen kann. Auch dieser Wert sollte weder zu groß noch zu klein gewählt werden – wobei jedoch ein zu kleiner Wert viel fataler ist als ein zu großer Wert.

Im Gegensatz zur optimalen `table_cache`-Einstellung können Sie den optimalen Wert in diesem Fall leichter bestimmen.

Rufen Sie auch hierzu den MySQL-Monitor auf und ermitteln Sie den Wert der aktuellen Einstellung:

```
mysql> SHOW VARIABLES LIKE 'max_connections';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| max_connections | 300   |
+-----+-----+
1 row in set (0.00 sec)
```

Nun ermitteln Sie den Wert von `max_used_connections`:

```
mysql> SHOW STATUS LIKE 'max_used_connections';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Max_used_connections | 96    |
+-----+-----+
1 row in set (0.00 sec)
```

Dieser Wert zeigt Ihnen, wie viele Verbindungen bisher maximal gleichzeitig zum MySQL-Server bestanden. Der Wert von 96 liegt deutlich unter 300. Sie können diese Einstellung also beibehalten. Reicht der Wert von *max_used_connections* allerdings näher an die 300 heran, sollten Sie unbedingt die *max_connections*-Einstellung erhöhen.

9.6.4 table_locks_immediate vs. table_locks_waited

Um den Zugriff auf eine Tabelle zu organisieren, kennt MySQL drei verschiedene Locking-Mechanismen:

- *table-level locking* – im Falle von MyISAM-Tabellen
- *page-level locking* – im Falle von BDB-Tabellen
- *row-level locking* – im Falle von InnoDB-Tabellen

Knapp zusammengefasst: MySQL sperrt im Zweifelsfall entweder ganze Tabellen oder nur einzelne Datensätze in einer Tabelle. Jede dieser Locking-Methoden hat ihre Vor- und Nachteile und deren Erläuterung sollte Inhalt eines MySQL-Buches sein und nicht eines XAMPP-Buches.

Für Sie sollten folgende Informationen bzw. Argumentation interessant sein:

- MyISAM ist das Standardtabellenformat von MySQL.
- MyISAM-Tabellen verwenden *Table-level locking*-Verfahren.
- *Table-level locking* ist schnell, solange es keine Locking-Konflikte gibt.
- Gibt es Konflikte, sollten Sie auf InnoDB umsteigen, da hier ein *row-level locking* verwendet wird und die Wahrscheinlichkeit von Locking-Problemen geringer ist.
- Steigen Sie aber nur bei bestehenden Locking-Problemen auf InnoDB um. Generell wird InnoDB langsamer bzw. ressourcenintensiver als MyISAM sein.

Ob Konflikte vorliegen, können Sie im MySQL-Monitor mit der folgenden Anfrage feststellen:

```
mysql> SHOW STATUS LIKE 'table%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Table_locks_immediate | 10925090 |
| Table_locks_waited    | 2139   |
+-----+-----+
2 rows in set (0.00 sec)
```

Ist der Wert von *table_locks_waited* viel größer als der Wert von *table_locks_immediate*, dann hat Ihre Datenbank oft Locking-Konflikte beim Zugriff auf ihre Tabellen. In diesem Fall sollten Sie dringend überlegen, auf InnoDB umzusteigen.

9.6.5 key_buffer_size

Der Key Buffer (oft auch Key Cache genannt) beinhaltet die Indexinformationen der verschiedenen Tabellen Ihrer Datenbanken. Im Prinzip kann man sagen, dass dieser Buffer so groß wie möglich sein sollte. Allerdings sollte er auf einem System, auf dem nur der MySQL-Server läuft, nicht mehr als 25 % des Hauptspeichers belegen. Bei Systemen, auf denen auch noch andere Dienste laufen, sollte dieser Wert entsprechend kleiner gehalten werden. Sonst bekommen die anderen Dienste nicht genug Speicher zum Atmen.

Überprüfen Sie zunächst Ihre bestehende Konfiguration:

```
mysql> SHOW VARIABLES LIKE 'key_buffer_size';
+-----+-----+
| Variable_name | Value      |
+-----+-----+
| key_buffer_size | 268435456 |
+-----+-----+
1 row in set (0.00 sec)
```

In diesem Fall ist der Key Buffer 256 Mbyte groß. Da der Server, von dem die obige Ausgabe kommt, 2 Gbyte Hauptspeicher hat, könnte der Key Buffer also größer dimensioniert werden.

Bevor Sie den Key Buffer Ihrer Datenbank aber vergrößern, sollten Sie überprüfen, ob der Key Buffer Ihres Systems überhaupt vergrößert werden muss.

Gehen Sie dabei folgendermaßen vor:

Ermitteln Sie im MySQL-Monitor die Werte für *key_read_requests* und *key_reads*:

```
mysql> SHOW STATUS LIKE 'key_read%';
+-----+-----+
| Variable_name | Value      |
+-----+-----+
| Key_read_requests | 1034631299 |
| Key_reads        | 29267      |
+-----+-----+
2 rows in set (0.00 sec)
```

Nun teilen Sie den Wert von *key_reads* durch den Wert von *key_read_requests*:

$29267 / 1034631299 = 0.00002828737157699305$

Kommt dabei ein Wert heraus, der kleiner als 0.01 ist, dann ist die *key_buffer_size* groß genug gesetzt. Ist der Wert größer als 0.01, sollten Sie überlegen, den Key Buffer zu vergrößern.

Auch wenn Sie eben einen Wert errechnet haben, der kleiner als 0.01 ist, sollten Sie noch folgende Werte überprüfen:


```
mysql> SHOW STATUS LIKE 'key_write%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Key_write_requests | 778787 |
| Key_writes        | 307558 |
+-----+-----+
2 rows in set (0.00 sec)
```

Teilen Sie den Wert von *key_writes* durch den Wert von *key_write_requests*:

$307558 / 778787 = 0.39491927831358253283$

Hier sollte ein Wert um 1 oder kleiner herauskommen. Ein größerer Wert ist jedoch schlecht und zeigt an, dass Sie den Key Buffer vergrößern sollten.

9.6.6 log_slow_queries

Jetzt kommt noch ein letzter Tipp. Mit den bisherigen Einstellungen konnten Sie MySQL aus rein administrativer Sicht optimieren. Sind Sie damit an die Grenzen gestoßen, werden Sie als Nächstes versuchen müssen, die Effizienz der SQL-Anfragen Ihrer Applikation zu überprüfen.

Der MySQL-Server bietet Ihnen hier eine sehr praktische Funktion an: das so genannte *Slow Query Log*. Über dieses Logfile können Sie besonders langsame SQL-Anfragen protokollieren lassen.

Nehmen Sie dazu folgende Einstellung in der Konfigurationsdatei Ihres MySQL-Servers vor:

```
[mysqld]
log_slow_queries = /tmp/mysql-slow.log
long_query_time = 1
```

Mit `log_slow_queries = /tmp/mysql-slow.log` stellen Sie ein, in welche Datei das Slow Query Log geschrieben werden soll. Und mit `long_query_time = 1` stellen Sie ein, dass Anfragen, die länger als 1 Sekunde dauern, mitgeloggt werden sollen.

Starten Sie nun den MySQL-Server neu und lassen Sie ihn ein paar Tage mit der zu testenden Applikation laufen.

Die Slow Query Log-Datei können Sie wunderbar mit einem Texteditor lesen. Sie können aber auch den `mysqldumpslow`-Befehl verwenden. Er hat den Vorteil, dass Sie mit ihm gleich nur die häufigsten, langsamen SQL-Anfragen anzeigen lassen können.

Zum Beispiel:

```
/opt/lampp/bin/mysqldumpslow -t 2 -s=c /tmp/mysql-slow.log
```

Mit diesem Befehl lassen Sie sich die Top 2 (-t 2) der am häufigsten vorkommenden (-s=c) SQL-Anfragen anzeigen. Das könnte zum Beispiel so aussehen:

Listing 9.2: Auszug aus der mysql-slow.log

```
Reading mysql slow query log from /tmp/mysql-slow.log
```

```
Count: 15  Time=3.33s (50s)  Lock=0.00s (0s)  Rows=0.0 (0)
  SELECT word_id
  FROM phpbb_search_wordmatch
  GROUP BY word_id
  HAVING COUNT(word_id) > N
```

```
Count: 6  Time=5.50s (33s)  Lock=0.00s (0s)  Rows=1959.0 (11754)
  SELECT m.post_id
  FROM phpbb_search_wordlist w, phpbb_search_wordmatch m
  WHERE w.word_text LIKE 'S'
  AND m.word_id = w.word_id
  AND w.word_common <> N
```

Überprüfen Sie nun, ob Sie die angegebenen SELECTs effizienter gestalten könnten. Hilfreich dürfte dabei das EXPLAIN-Kommando von MySQL sein.

Achtung



Liebe Windows-Benutzer: Das Programm `mysqldumpslow` existiert für Windows nicht!

9.7 Perl

Perl steht sicherlich nicht mehr in dem Rampenlicht, in dem es noch vor zehn Jahren stand. Die Popularität und Einfachheit von PHP hat es schon lange ins Abseits gedrängt. Aber trotzdem sollte man Perl nicht unterschätzen und ganz links liegen lassen.

Es gibt immer noch eine Vielzahl webbasierter Perl-Anwendungen und auch der Umfang der Perl-Community ist nach wie vor sehr groß. Aus diesem Grund ist auch Perl immer ein fester Bestandteil von XAMPP geblieben.

Einer der großen Nachteile von Perl gegenüber PHP ist, dass es eine klassische CGI-Anwendung ist, also ein vom Apache-Webserver völlig getrenntes Programm, das bei jedem Aufruf komplett neu gestartet und ausgeführt werden muss. Dafür kann Perl natürlich nichts, da trägt CGI ganz allein die Schuld, aber das macht webbasierte Perl-Anwendungen einfach unerträglich langsam.

Aber eigentlich ist die Verbindung Perl/CGI schon seit Jahren Geschichte. Seit es *mod_perl* gibt, muss Perl nicht als CGI ausgeführt werden, sondern kann – vergleichbar zu PHP – im Apache ausgeführt werden.

9.7.1 mod_perl

Der Geschwindigkeitsvorteil, der mit *mod_perl* erreicht werden kann, ist sicherlich nur ein Aspekt von *mod_perl*, und die Möglichkeiten, die es bietet, gehen noch weit darüber hinaus. Von *mod_perl* handeln ganze Bücher – und das nicht ohne Grund. In diesem Kapitel geht es aber nur um Performance Tuning und so wird auch *mod_perl* nur aus diesem Blickwinkel betrachtet.

In diesem Beispiel gehen wir davon aus, dass Ihre Perl-Programme mit der Endung *.pl* enden. Sollte das bei Ihnen anders sein, dann passen Sie einfach das *pl* in den folgenden Zeilen an Ihre Gegebenheiten an.

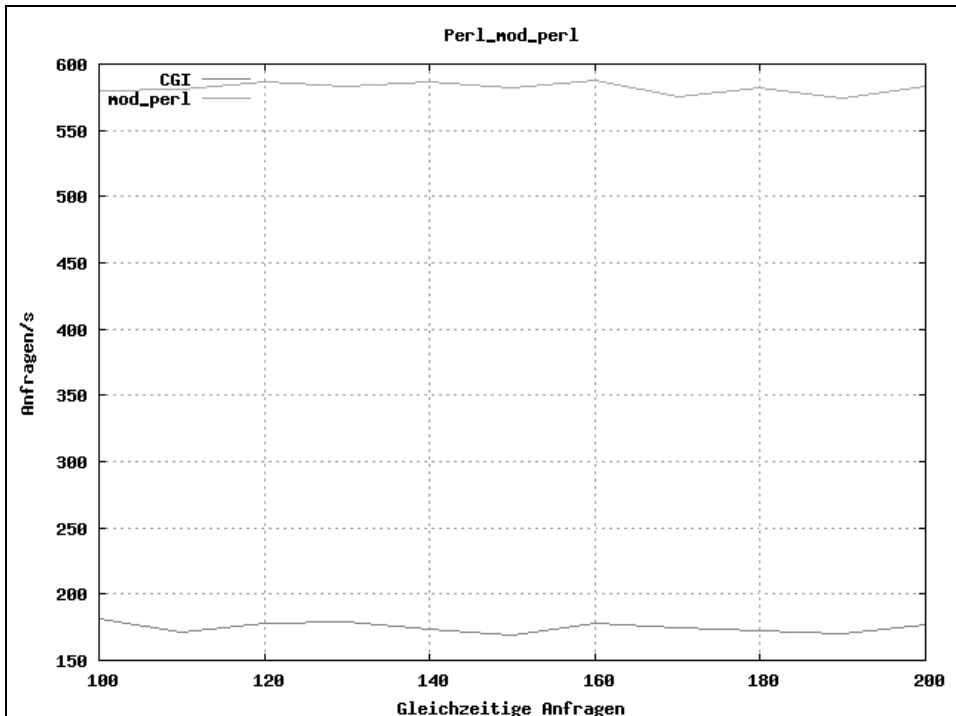


Abbildung 9.9: In unserem Testlabor war ein Hello-World-Perl-Programm als *mod_perl* gut dreimal so schnell wie seine reine CGI-Variante.

Fügen Sie folgende Zeilen an das Ende der *extra/httpd-xampp.conf*-Datei an. Unter MS Windows benötigen Sie das XAMPP Perl-Add-On. Dort finden Sie die Konfiguration unter *extra\perl.conf* oder bei neueren Versionen unter *extra\xampp-perl.conf*.

Listing 9.3: Einstellung in der Apache-Konfigurationsdatei

```
PerlModule ModPerl::PerlRun
<Files ~ "\.pl$">
    SetHandler perl-script
    PerlResponseHandler ModPerl::PerlRun
    PerlOptions +ParseHeaders
    Options +ExecCGI
</Files>
```

Starten Sie danach den Apache neu. Ihr Apache ist nun so konfiguriert, dass Perl-Programme nicht mehr über das CGI ausgeführt werden, sondern direkt im Apache über das Perl-Modul.



10 Sicherheit

Ein leider weit verbreiteter Trugschluss hört sich etwa so an: Ich habe doch keine wichtigen Daten, mein Server ist eh so unbedeutend, also muss ich mir doch keine Gedanken um die Sicherheit machen. Mal ernsthaft: Auch wenn Sie keine Wertgegenstände im Auto liegen hätten, würden Sie es doch auch nicht unverschlossen in der Innenstadt in Bahnhofsnähe stehen lassen.

Eine andere schöne Metapher beschreibt einen weiteren Sicherheitsaspekt, der ebenso oft vergessen wird: Sie lassen Ihr Auto regelmäßig warten und beim Fahren verwenden Sie einen Gurt. Beim Computer gibt es ähnliche Servicewartungen (zum Beispiel das Rotieren von Logdateien oder das Aktualisieren von Softwarepaketen) und auch einen Gurt gibt es (das Backup, das einem nach einem Crash rettet). Aber im Falle des Computers wird auch hier wieder oft geschlampt, was das Zeug hält.

In diesem Kapitel werden wir daher die wichtigsten sicherheitsrelevanten Informationen vorstellen und Ihnen zeigen, wie Sie sie auf Ihr XAMPP-System anwenden können.

10.1 Begrifflichkeiten: Server, Ports und Protokolle

Die größte Gefahr für Ihr System kommt von außen. Außen heißt im Fall von XAMPP: über das Netzwerk und ganz besonders über das Internet. Egal ob Sie Linux oder Windows verwenden: Die Gefahren sind dieselben.

Um Missverständnissen vorzubeugen, sollen zunächst die wichtigsten Begriffe definiert werden:

- *Server*
- *Service*
- *Dienst*
- *Daemon*

Auch wenn diese Begriffe vielleicht noch kleine Differenzierungen ausmachen, bedeuten sie doch eigentlich alle das Gleiche und sind durchaus Synonyme. *Dienst* ist lediglich die deutsche Übersetzung von *Service* und *Daemon* ist eigentlich nur das Unix-Pendant zu *Service*. Der Begriff *Server* kann noch ein bisschen missverständlich sein, da er sowohl die Hardware als auch die Serversoftware meint.

■ Port

Ein bestimmter Dienst/Server auf Ihrem System wird über einen bestimmten (in der Regel nur für ihn reservierten) Port übers Netzwerk erreicht. Der Port wird über eine Portnummer identifiziert und liegt im Wertebereich von 1 bis 65535. Sperren Sie einen Port, dann ist dessen Dienst nicht mehr übers Netzwerk erreichbar.

■ Protokoll

Ein Protokoll beschreibt quasi die Sprache, die der Server spricht. Dabei gibt es ganz grundlegende Protokolle (wie zum Beispiel die IP-Protokolle UDP und TCP), die dafür sorgen, dass die Daten überhaupt von dem einen Rechner zum anderen Rechner kommen. Und es gibt Protokolle, die auf der sog. Applikationsebene liegen: also Protokolle, die von den unterschiedlichen Diensten gesprochen werden. Tabelle 10.1 gibt einen Überblick über die wichtigsten Ports und Protokolle im XAMPP-Kontext.

Portnummer	IP-Protokoll	Protokoll	Server, Service, Dienst, Daemon
20	TCP	FTP	ProFTPD / FileZilla Server
21	TCP	FTP	ProFTPD / FileZilla Server
25	TCP	SMTP	Mercury
53	TCP/UDP	DNS	BIND
80	TCP	HTTP	Apache
110	TCP	POP3	Mercury
143	TCP	IMAP	Mercury
443	TCP/UDP	HTTPS	Apache
1433	TCP	proprietär	MSSQL (Standard)
1434	UDP	proprietär	MSSQL
1521	TCP/UDP	proprietär	Oracle (Standard)
1810	TCP/UDP	proprietär	Oracle
2481	TCP/UDP	proprietär	Oracle
3306	TCP	proprietär	MySQL
3389	TCP	proprietär	Remote Desktop unter Windows
5432	TCP/UDP	proprietär	PostgreSQL
7778	TCP/UDP	proprietär	Oracle
8007	TCP	AJP12	Tomcat
8009	TCP	AJP13	Tomcat
8080	TCP	HTTP	Tomcat

Tabelle 10.1: Die wichtigsten Ports und Protokolle im XAMPP-Umfeld

10.2 Linux

Dieses Kapitel behandelt die charakteristischen Sicherheitsfragen für XAMPP unter Linux. Allerdings können wir nicht die Frage beantworten, wie Sie Ihr Linux-System generell vor Angriffen schützen. Die Beantwortung dieser Frage ist allzu umfangreich und daher nicht Bestandteil dieses Buches. Doch möchten wir Ihnen Tipps geben und vielleicht so manche Anregung rund um das Thema Sicherheit vermitteln.

10.2.1 Eine generelle Absicherung

Wie schon an anderer Stelle erwähnt, ist XAMPP nicht für den Produktionseinsatz gedacht, sondern nur für Entwickler in Entwicklungsumgebungen. Das hat zur Folge, dass XAMPP absichtlich nicht restriktiv, sondern im Gegenteil sehr offen vorkonfiguriert ist. Für einen Entwickler ist das ideal, da ihm so keine Grenzen vom System vorgeschrieben werden. Für einen Produktionseinsatz ist diese Konfiguration allerdings überhaupt nicht geeignet.

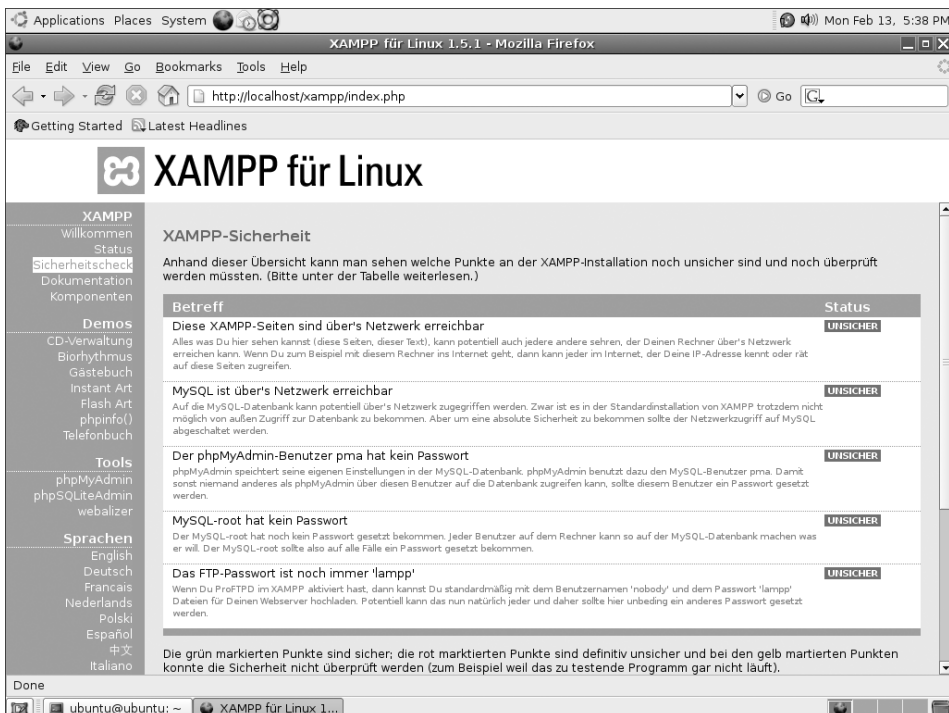
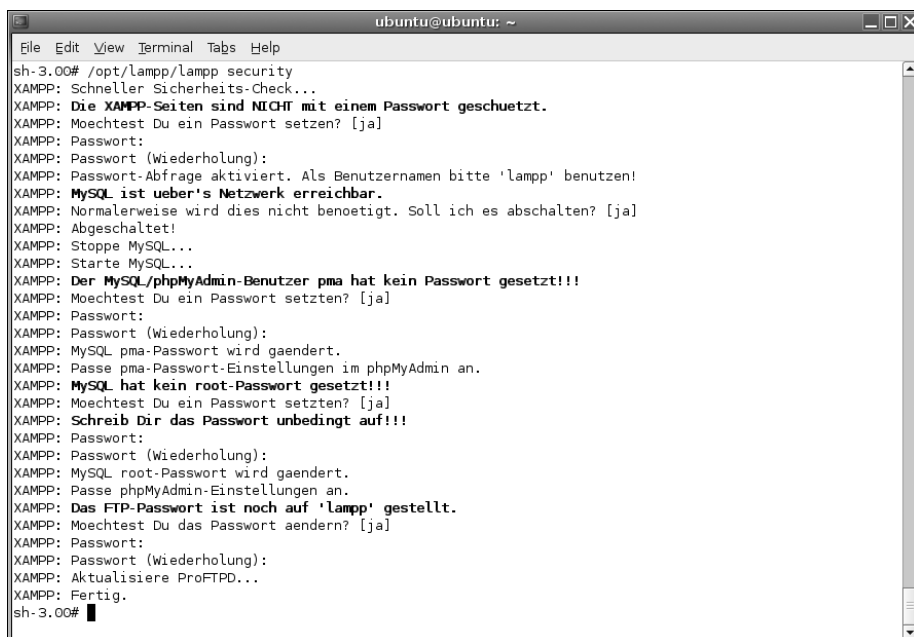


Abbildung 10.1: Die Konfiguration von XAMPP ist auf Entwickler zugeschnitten und standardmäßig unsicher.

Hier eine Liste der Dinge, die in XAMPP absichtlich (!) unsicher sind:

- Der MySQL-Administrator (*root*) hat kein Passwort.
- Der MySQL-Daemon ist übers Netzwerk erreichbar.
- Der ProFTPD-Daemon benutzt das Passwort »lamp«.
- phpMyAdmin ist übers Netzwerk erreichbar.
- MySQL und Apache laufen unter der gleichen Benutzerkennung (*nobody*).

Auf den XAMPP-Demoseiten kann man sich unter dem Punkt SICHERHEITSCHECK den aktuellen Sicherheitszustand seiner XAMPP-Installation anzeigen lassen.



```

ubuntu@ubuntu: ~
File Edit View Terminal Tabs Help
sh-3.00# /opt/lampp/lamp security
XAMPP: Schneller Sicherheits-Check...
XAMPP: Die XAMPP-Seiten sind NICHT mit einem Passwort geschützt.
XAMPP: Möchtest Du ein Passwort setzen? [ja]
XAMPP: Passwort:
XAMPP: Passwort (Wiederholung):
XAMPP: Passwort-Abfrage aktiviert. Als Benutzernamen bitte 'lamp' benutzen!
XAMPP: MySQL ist ueber's Netzwerk erreichbar.
XAMPP: Normalerweise wird dies nicht benoetigt. Soll ich es abschalten? [ja]
XAMPP: Abgeschaltet!
XAMPP: Stoppe MySQL...
XAMPP: Starte MySQL...
XAMPP: Der MySQL/phpMyAdmin-Benutzer pma hat kein Passwort gesetzt!!!
XAMPP: Möchtest Du ein Passwort setzen? [ja]
XAMPP: Passwort:
XAMPP: Passwort (Wiederholung):
XAMPP: MySQL pma-Passwort wird geändert.
XAMPP: Passe pma-Passwort-Einstellungen im phpMyAdmin an.
XAMPP: MySQL hat kein root-Passwort gesetzt!!!
XAMPP: Möchtest Du ein Passwort setzen? [ja]
XAMPP: Schreib Dir das Passwort unbedingt auf!!!
XAMPP: Passwort:
XAMPP: Passwort (Wiederholung):
XAMPP: MySQL root-Passwort wird geändert.
XAMPP: Passe phpMyAdmin-Einstellungen an.
XAMPP: Das FTP-Passwort ist noch auf 'lamp' gestellt.
XAMPP: Möchtest Du das Passwort ändern? [ja]
XAMPP: Passwort:
XAMPP: Passwort (Wiederholung):
XAMPP: Aktualisiere ProFTPD...
XAMPP: Fertig.
sh-3.00#
  
```

Abbildung 10.2: Die XAMPP für Linux-Installation sollte mittels der Sicherheitsfunktion abgesichert werden.

Wollen Sie XAMPP in einem Netzwerk betreiben, so dass der XAMPP-Server auch über dieses erreichbar ist, sollten Sie unbedingt den folgenden Befehl aufrufen, um diese Unsicherheiten zu schließen:

```
/opt/lampp/lamp security
```

Nun werden Sie durch einen kleinen Dialog geführt, in dessen Verlauf Sie Schritt für Schritt Ihre XAMPP-Installation absichern können:

```

XAMPP: Schneller Sicherheits-Check...
XAMPP: Die XAMPP-Seiten sind NICHT mit einem Paßwort geschützt.
XAMPP: Möchtest Du ein Paßwort setzen? [ja]
  
```


Hier sollten Sie auf alle Fälle mit »ja« antworten. Die Demoseiten und besonders der Sicherheitscheck selbst sind nur für Sie bestimmt.



Abbildung 10.3: Die XAMPP-Demoseiten sind nun nur noch mit einem Passwort erreichbar.

XAMPP: MySQL ist über's Netzwerk erreichbar.

XAMPP: Normalerweise wird dies nicht benötigt. Soll ich es abschalten? [ja]

Wenn Sie kein Tool verwenden, um auf MySQL direkt über das Netzwerk zuzugreifen, dann sollten Sie auch hier mit »ja« antworten. Zum normalen Betrieb von XAMPP muss MySQL in keiner Weise über das Netzwerk erreichbar sein.

Besonders wenn Sie MySQL auf einem Rechner laufen haben, der direkt übers Internet erreichbar ist, sollten Sie unbedingt die Netzwerkerreichbarkeit von MySQL abschalten oder alternativ wenigstens den Zugang zum MySQL-Port (3306) über eine Software-Firewall einschränken.

XAMPP: Der MySQL/phpMyAdmin-Benutzer pma hat kein Passwort gesetzt!!!

XAMPP: Möchtest Du ein Paßwort setzten? [ja]

Der Benutzer *pma* ist lediglich für die interne Kommunikation von phpMyAdmin und MySQL notwendig. phpMyAdmin speichert ein paar Einstellungen in der MySQL-Datenbank und verwendet dazu diesen Benutzer.

XAMPP: MySQL hat kein root-Paßwort gesetzt!!!

XAMPP: Möchtest Du ein Paßwort setzten? [ja]

Das ist vielleicht die wichtigste Einstellung überhaupt: das *root*-Passwort von MySQL. Setzen Sie hier unbedingt ein Passwort. Und merken Sie es sich. Schreiben Sie es sich auf und verlieren Sie den Zettel nicht.



Hinweis

Wenn Sie hier ein Passwort gesetzt haben, kann die XAMPP-CD-Verwaltungsdemo nicht mehr funktionieren, da sie nun nicht mehr auf die MySQL-Datenbank zugreifen kann. Nicht nur menschliche Benutzer der Datenbank müssen sich authentifizieren, sondern auch Programme, wenn sie auf die Datenbank zugreifen wollen.

Sie haben nun zwei Möglichkeiten:

1. Sie fügen das neue *root*-Passwort in die `mysql_connect()`-Funktion der Datei `htdocs/xampp/cds.php` ein. Allerdings ist diese Methode nicht wirklich zu empfehlen, da damit Ihr *root*-Passwort unverschlüsselt auf der Festplatte zu finden ist. Zwar sollte von »außen« niemand darankommen, aber trotzdem ist es immer eine schlechte Idee, wichtige Passwörter unverschlüsselt irgendwo gespeichert zu haben.
2. Richten Sie für die CD-Verwaltung einen neuen Benutzer mit phpMyAdmin ein und geben Sie diesem Benutzer Zugriff auf die `cdcol`-Datenbank. Und fügen Sie dann dessen Namen und Passwort der `mysql_connect()`-Funktion in der `cds.php` hinzu.

Die zweite Möglichkeit ist zwar viel komplizierter, aber der ersten auf jeden Fall vorzuziehen. Im Zweifelsfall sollten Sie einfach auf die CD-Verwaltungsdemo verzichten. So wichtig ist sie ja auch nicht.

XAMPP: Das FTP-Passwort ist noch auf 'lamp' gestellt.

XAMPP: Möchtest Du das Paßwort ändern? [ja] ja

Neben dem *root*-Passwort für MySQL dürfte das Passwort für FTP die wichtigste Sicherheitseinstellung sein. Ist hier kein Passwort gesetzt, kann jedermann von außen auf die Dateien des `htdocs`-Verzeichnis zugreifen und dort auch eigene Dateien ablegen. Diese Sicherheitslücke wird sehr gerne von Phishern verwendet, um gefälschte Webseiten ins Netz zu stellen. Es ist also eine wirklich wichtige Einstellung.

XAMPP: Fertig.

Und das war's. Wenn Sie überall mit »ja« geantwortet und ein Passwort gesetzt haben, dann können Sie nun davon ausgehen, dass Ihre XAMPP-Installation sicher ist.

Die einzigen Unsicherheiten, die nun noch vorhanden sein könnten, sind entweder in Ihrem Betriebssystem, in einer bisher unbekannten Sicherheitslücke in einer der XAMPP-Komponenten oder in Ihrer Webanwendung zu suchen.

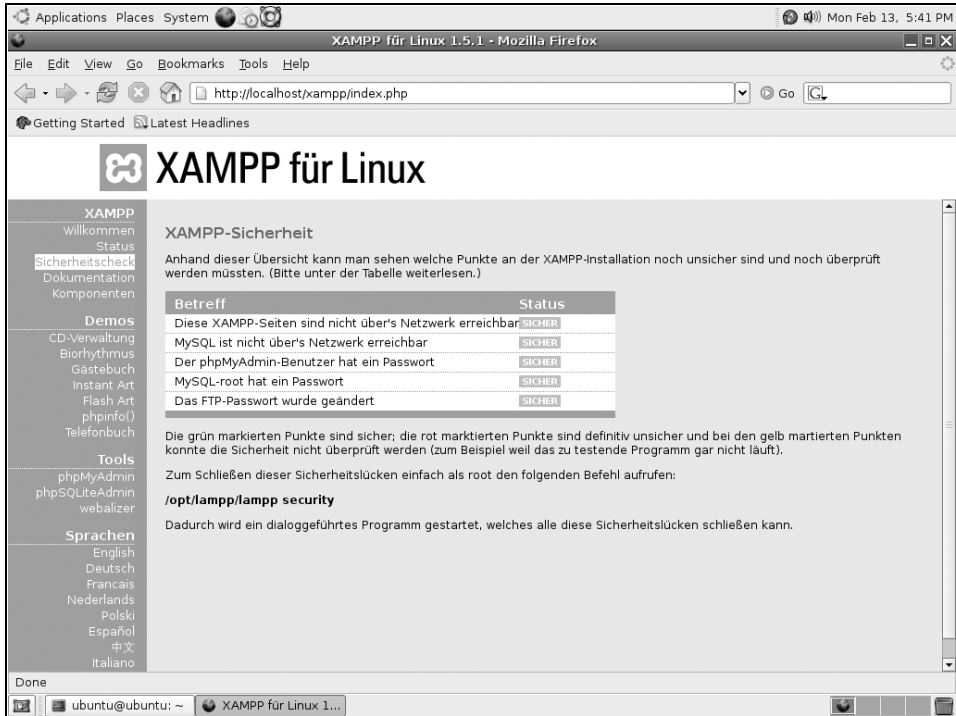


Abbildung 10.4: Alles im grünen Bereich: Nach Ausführung der Sicherheitsfunktion sollte die Konfiguration von XAMPP nun sicher sein.

10.2.2 Zugriffsrechte

Unsere Erfahrung hat gezeigt, dass viele XAMPP-Nutzer zu Linux wechseln, nachdem sie mit XAMPP für Windows erste Schritte gemacht haben. Nun ist Linux aufgrund der Entwicklungen der letzten Jahre im Bereich der grafischen Oberflächen immer benutzerfreundlicher geworden, doch ein paar kleine Details bereiten dem Windows-Benutzer noch immer Kopfzerbrechen.

Die größte dieser Kleinigkeiten dürften die Dateizugriffsrechte von Linux sein. Zum einen sind sie gänzlich anders aufgebaut als ihr Gegenstück aus der Windows-Welt und zum anderen sind Dateizugriffsrechte unter Windows bis heute für den Heim-anwender kaum gebräuchlich.

Grundlagen

Am einfachsten dürften die Zugriffsrechte anhand eines Beispiels zu erklären sein. Gehen Sie in eine Terminal-Shell und rufen Sie den `ls`-Befehl, im folgenden Beispiel fett hervorgehoben, auf:

```
# ls -l /opt/lampp/htdocs/
total 6
-rw-r--r--  1 nobody  root           163 Oct 31  2003 index.html
drwxr-xr-x  2 nobody  root            48 Dec 26  2004 webalizer
drwxr-xr-x  6 root     root          1624 Feb  6  17:50 xampp
```

Mit diesem Befehl werden die Inhalte des `htdocs`-Verzeichnisses angezeigt. Schauen Sie sich diese Zeilen etwas genauer an. Für die Zugriffsrechte relevant sind die erste, die dritte und die vierte Spalte.

```
drwxr-xr-x
```

Dies sind die eigentlichen Zugriffsrechte, wobei das erste Zeichen – in diesem Fall ein `d` – nicht wirklich zu den Zugriffsrechten gehört, sondern den Dateityp definiert. »d« steht dabei für *directory*, also Verzeichnis. Ein »-« steht für *plain file*: eine ganz normale Datei.

Die folgenden 9 Zeichen können Sie in drei Bereiche aufteilen. Etwa so:

```
rwx  r-x  r-x
```

Der erste Bereich definiert die Zugriffsrechte für den Eigentümer dieser Datei. Der zweite Bereich definiert die Zugriffsrechte für die Gruppe, zu der diese Datei gehört, und der dritte und letzte Bereich definiert die Rechte für alle anderen Benutzer auf Ihrem Linux-System.

Innerhalb eines Bereichs werden die drei Rechte definiert: Leserecht (das Zeichen `r` befindet sich an erster Stelle), Schreibrecht (das Zeichen `w` steht an zweiter Stelle) und das Ausführbarkeitsrecht (ein `x` an der dritten Stelle). Dabei bedeutet das Ausführbarkeitsrecht zweierlei: Bei Dateien bedeutet es, dass diese Datei ausgeführt werden kann. Die Datei muss dazu auch eine Programmdatei sein. Bei einem Verzeichnis bedeutet dieses Recht, dass dieses Verzeichnis betreten werden darf. Man kann also in dieses Verzeichnis wechseln.

```
-rw-r--r--  1 nobody  root           163 Oct 31  2003 index.html
```

Die obige Zeile liest sich also folgendermaßen:

- Der Besitzer der Datei ist *nobody*.
- Die Datei gehört der Gruppe *root*, die übrigens so gut wie nichts mit dem gleichnamigen Benutzer *root* zu tun hat.
- Der Besitzer dieser Datei darf sie lesen und schreiben.
- Die Mitglieder der Gruppe *root* dürfen diese Datei lesen.
- Alle anderen Benutzer Ihres Linux-Systems dürfen diese Datei ebenfalls lesen.

In der Praxis

XAMPP ist so vorbereitet, dass Sie über den Benutzer *nobody* auf das `htdocs`-Verzeichnis zugreifen und dort Dateien ändern und hochladen können. Für die ersten Schritte sollte das reichen. In der Regel werden Sie aber eigene Benutzer anlegen wollen, vielleicht sogar mehrere, und diese mit unterschiedlichsten Rechten versehen. Verwenden Sie zum Anlegen der Benutzer die Administrationswerkzeuge Ihrer Linux-Distribution. Meist werden Sie die entsprechenden Funktionen unter `BENUTZER VERWALTUNG` oder `BENUTZER UND GRUPPEN` finden.

Zum Ändern der Zugriffsrechte einer Datei verwenden Sie entweder eines der folgenden Shell-Kommandos oder den Dateimanager Ihrer grafischen Oberfläche.

```
chmod
```

Mit diesem Befehl können Sie die Zugriffsrechte einer Datei ändern.

Zum Beispiel:

```
chmod g+w index.html
```

Damit geben Sie den Mitgliedern der Gruppe, zu der diese Datei gehört, Schreibrechte. Der erste Buchstabe nach dem Befehl (in diesem Fall das `g`) symbolisiert den Bereich für den Sie die Zugriffsrechte ändern wollen: Ein `u` steht für den Eigentümer-Bereich, das `g` für den Gruppen-Bereich und ein `o` für alle anderen Benutzer des Systems.

Nach dem `g` folgt in dem Beispiel oben ein Plus-Zeichen. Damit weisen Sie den `chmod`-Befehl an, dass Rechte hinzugefügt werden sollen. Wenn Sie Rechte entfernen wollen, dann verwenden Sie an dieser Stelle ein Minus-Zeichen.

Das letzte Zeichen (im Beispiel oben ein `w`) bestimmt welches Recht hinzugefügt werden soll.

```
chown
```

Mit diesem Befehl ändern Sie den Eigentümer einer Datei.

Zum Beispiel:

```
chown oswald index.html
```

So setzen Sie für die Datei `index.html` den neuen Eigentümer *oswald*.

```
chgrp
```

Mit diesem Befehl können Sie die Gruppe einer Datei ändern.

Der `chgrp`-Befehl funktioniert analog zum `chmod`-Befehl. Zum Beispiel:

```
chgrp friends index.html
```

Die Datei `index.html` wird der Gruppe *friends* zugeordnet.

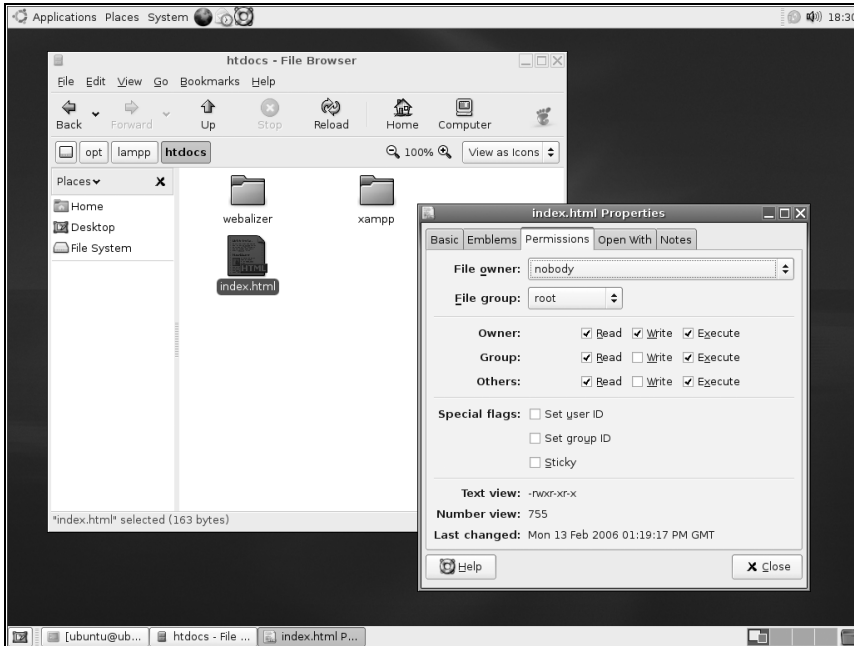


Abbildung 10.5: Mit dem Dateimanager Ihres Linux-Systems können Sie ganz einfach den Besitzer, die Zugriffsrechte oder die Gruppe einer Datei ändern. Im Screenshot sieht man den Gnome Nautilus unter Ubuntu 5.10.

10.2.3 Firewalls

Gerade wenn Sie einen Root-Server im Internet gemietet haben, werden Sie nicht die Möglichkeit haben, auf eine Hardware-Firewall zugreifen zu können. Trotzdem sollte Ihr System unbedingt von einer Firewall geschützt werden. Bei Windows gibt es dazu die sog. *Personal Firewalls* (auch generell *Software-Firewalls* genannt). Bei Linux ist die Firewall-Funktionalität schon fest im Betriebssystem eingebaut und muss von Ihnen nur noch aktiviert und konfiguriert werden. Nicht ohne Grund basieren ja auch viele Hardware-Firewalls auf Linux.

Am einfachsten wird die Firewall-Konfiguration über die grafische Administrationsoberfläche Ihrer Linux-Distribution sein. Im Screenshot sehen Sie die Firewall-Konfigurationssicht von Fedora Core 5. Aber auch openSUSE bietet in seinem YaST eine ähnliche Oberfläche zur Konfiguration der Firewall an.

Geben Sie über Ihre Firewall nur die Ports der Dienste frei, die für die Außenwelt bestimmt sind. In der Regel werden das die Ports 80 und 22 sein. 80, weil dort Ihr Webserver zu erreichen ist, und 22, weil dort der SSH-Server läuft und Sie ja vielleicht auch darüber Ihren Server administrieren wollen.



Abbildung 10.6: Einfacher geht's kaum noch: Firewall-Konfiguration über die grafische Oberfläche. Screenshot von Fedora Core 5 (Test 2)

Schwieriger wird es allerdings, wenn Sie auf Ihrem Linux-System keine grafische Oberfläche haben oder es in Ihrer Distribution überhaupt kein Tool zur Konfiguration der Firewall gibt. Und noch schwieriger wird es, wenn Sie Ihren Server selbst nur übers Netzwerk erreichen können. In diesem Fall besteht die Gefahr, dass Sie sich durch die Firewall von Ihrem eigenen Server ausschließen.

In diesem Fall sei folgende Minifirewall vorgeschlagen, die seit XAMPP für Linux 1.5.2 mit im XAMPP-Paket enthalten ist. Sie ist einfach und bietet einen guten grundlegenden Firewall-Schutz.

Listing 10.1: Eine Minifirewall unter Linux

```
#!/bin/sh

case "$1" in
    start)
        echo "Setting up firewall"
        iptables -P INPUT DROP
        iptables -P OUTPUT ACCEPT
        iptables -F
```

```

iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT
iptables -A INPUT -p tcp --syn --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --syn --dport 443 -j ACCEPT
iptables -A INPUT -p tcp --syn --dport 22 -j ACCEPT
;;
stop)
    echo "Shutting down firewall"
    iptables -F INPUT ACCEPT
    iptables -F OUTPUT ACCEPT
    iptables -F
    ;;
*)
    echo "Usage: $0 {start|stop}"
    exit 1
    ;;
esac

```

Ziemlich am Anfang finden Sie die folgenden drei Zeilen:

```

iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT

```

Damit wird erreicht, dass alle eingehenden Verbindungen abgelehnt und alle ausgehenden Verbindungen erlaubt werden. Da eine ausgehende Verbindung ja auch Antworten von außen erwartet und somit auch eine eingehende Verbindung ist, erlaubt die dritte Zeile (die mit ESTABLISHED), dass Daten für bestehende Verbindungen auch hereinkommen dürfen.

Die folgenden drei Zeilen bilden dann den Kern der Firewall:

```

iptables -A INPUT -p tcp --syn --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --syn --dport 443 -j ACCEPT
iptables -A INPUT -p tcp --syn --dport 22 -j ACCEPT

```

Mit ihnen wird der Zugriff auf die Ports 80, 443 und 22 von außen erlaubt. Der Zugriff auf alle anderen Ports wird verboten. Wenn Sie weitere Ports von außen erreichbar machen möchten, dann fügen Sie einfach eine analoge Zeile mit der entsprechenden Portnummer hinzu.

Sie finden dieses Firewall-Skript in der Datei `/opt/lampp/share/lampp/firewall` und können es mit folgendem Befehl starten:

```
/opt/lampp/share/lampp/firewall start
```

Zum Beenden der Firewall rufen Sie folgenden Befehl auf:

```
/opt/lampp/share/lampp/firewall stop
```


Bitte seien Sie aber vorsichtig, wenn Sie diese Firewall auf einem Server verwenden, den Sie nur übers Netzwerk erreichen können. Sie sollten sicherstellen, dass Sie die Möglichkeit haben, entweder den Rechner remote booten zu können oder über eine Konsole an den Rechner zu kommen.

10.2.4 Backups

Seit XAMPP für Linux 1.4.2 ist in die XAMPP-Installation eine Backup-Lösung eingebaut, die eine sehr einfache Sicherung Ihrer XAMPP-Daten ermöglicht.

Backup erstellen

Zum Erstellen des Backups rufen Sie den folgenden Befehl auf:

```
/opt/lampp/lampp backup
```

Sollten Sie für Ihre MySQL-Datenbank ein Passwort gesetzt haben, dann geben Sie es bitte als Parameter beim Aufruf dieses Befehls an. Zum Beispiel:

```
/opt/lampp/lampp backup geheim
```

Nun wird ein SQL-Dump der Datenbank und eine Sicherung der wichtigsten Daten Ihres XAMPP erstellt:

```
Backing up databases...
Backing up configuration, log and htdocs files...
Calculating checksums...
Building final backup file...
Backup finished.
Take care of /opt/lampp/backup/xampp-backup-03-03-06.sh
```

Folgende Verzeichnisse bzw. Dateien werden in diesem Schritt von der Backup-Funktion gesichert:

```
/opt/lampp/etc
/opt/lampp/htdocs
/opt/lampp/logs
/opt/lampp/phpmyadmin/config.inc.php
MySQL-Datenbanken
MySQL-Benutzer
```

Ihr Backup befindet sich nun in folgender Datei:

```
/opt/lampp/backup/xampp-backup-03-03-06.sh
```

Diese Datei sollten Sie nun auf eine CD brennen, auf eine externe Platte oder auf ein ganz anderes System sichern.

Backup wieder einspielen

Die Backup-Datei ist ein eigenständiges Shell-Skript, das selbst in der Lage ist, sich wieder einzuspielen. Alles, was Sie brauchen, ist eine »leere« frische XAMPP-Installation. Die Versionsnummer muss der des Backups entsprechen. Sie haben zwar auch die Möglichkeit, ein Backup auf eine andere Version einzuspielen, allerdings ist dann nicht wirklich sicher, ob das Backup auch funktionieren wird.

Starten Sie die neue XAMPP-Installation und rufen Sie Ihre Backup-Datei folgendermaßen auf:

```
sh /opt/lampp/backup/xampp-backup-03-03-06.sh
```

Wenn Sie die folgende Fehlermeldung erhalten, dann haben Sie XAMPP entweder nicht gestartet oder Ihr MySQL-Server hat bereits ein Passwort gesetzt:

```
Can't access your MySQL database.
```

```
a) Did you start it?
```

```
b) Add the MySQL root password to end of command line
```

Starten Sie XAMPP oder geben Sie in diesem Fall das MySQL bei Aufruf des Backup-Skripts mit an:

```
sh /opt/lampp/backup/xampp-backup-03-03-06.sh geheim
```

Nun sollten Sie folgende Meldungen sehen:

```
Checking integrity of files...
```

```
Restoring configuration, log and htdocs files...
```

```
Checking versions...
```

```
Installed: XAMPP 1.5.3
```

```
Backup from: XAMPP 1.5.3
```

```
Restoring MySQL databases...
```

```
Restoring MySQL user databases...
```

```
Backup complete. Have fun!
```

```
You may need to restart XAMPP to complete the restore.
```

Das Backup ist nun eingespielt. Um sicherzugehen, dass nun auch die neuen Konfigurationsdateien von den Servern eingelesen werden, sollten Sie XAMPP jetzt noch einmal neu starten:

```
# /opt/lampp/lampp restart
```

```
Stopping XAMPP for Linux 1.5.3...
```

```
XAMPP: Stopping Apache with SSL...
```

```
XAMPP: Stopping MySQL...
```

```
XAMPP: Stopping ProFTPD...
```

```
XAMPP stopped.
```

```
Starting XAMPP for Linux 1.5.3...
```

```
XAMPP: Starting Apache with SSL (and PHP5)...
```

```
XAMPP: Starting MySQL...
XAMPP: Starting ProFTPD...
XAMPP for Linux started.
```

Fertig, das Backup ist eingespielt und die Server laufen wieder mit der gleichen Konfiguration.

10.3 Windows

Dieses Kapitel behandelt die charakteristischen Sicherheitsfragen für XAMPP unter Windows. Allerdings können wir nicht die Frage beantworten, wie Sie Ihr Windows-System generell vor Angriffen schützen. Die Lösung dieser Frage ist sehr umfangreich und nicht Bestandteil dieses Buches. Doch möchten wir Ihnen Tipps geben und vielleicht so manche Anregung rund um das Thema Sicherheit vermitteln.

10.3.1 Eine generelle Absicherung

Die folgenden Maßnahmen sorgen für eine allgemeine Absicherung Ihrer *XAMPP* für *Windows*-Installation.

1. Starten Sie nur die Server, die Sie benötigen.
2. Definieren Sie für den MySQL-Administrator *root* ein Passwort. Das können Sie auch alternativ in der *XAMPP Security Konsole* unter der URL <http://localhost/security/xamppsecurity.php> durchführen. Oder über die Konsole:

```
D:\Programme\xampp\mysql\bin\mysql -h localhost -u root -p
mysql>use mysql;
mysql>UPDATE user SET Password=PASSWORD('xampp45rt') WHERE user='root';
mysql>FLUSH PRIVILEGES;
mysql>quit
```

3. Passen Sie für *phpMyAdmin* die `<xampp>\phpmyadmin\config.inc.php` an. Hier sollte der Wert `auth_type` von `config` wahlweise auf `http` für eine HTTP-basierte oder auf `cookie` für eine Cookie-basierte Authentifikation eingestellt sein. Wenn Sie die Einstellungen über die *Security Konsole* im XAMPP durchführen, wird die `config.inc.php` automatisch angepasst. Ein Auszug der `config.inc.php` bei einer Cookie-Authentifizierung könnte so aussehen:

```
$cfg['Servers'][$i]['auth_type']    = 'cookie';
$cfg['Servers'][$i]['user']        = '';
$cfg['Servers'][$i]['password']    = '';
```

4. Sichern Sie Ihre nicht öffentlichen Webverzeichnisse. Für den Schutz des XAMPP-Verzeichnisses können Sie erneut die *XAMPP Security Konsole* lokal aufrufen. Für eine schnelle Absicherung weiterer Ordner kopieren Sie einfach die `.htaccess` unter `htdocs\xampp` in die zu schützenden Verzeichnisse.

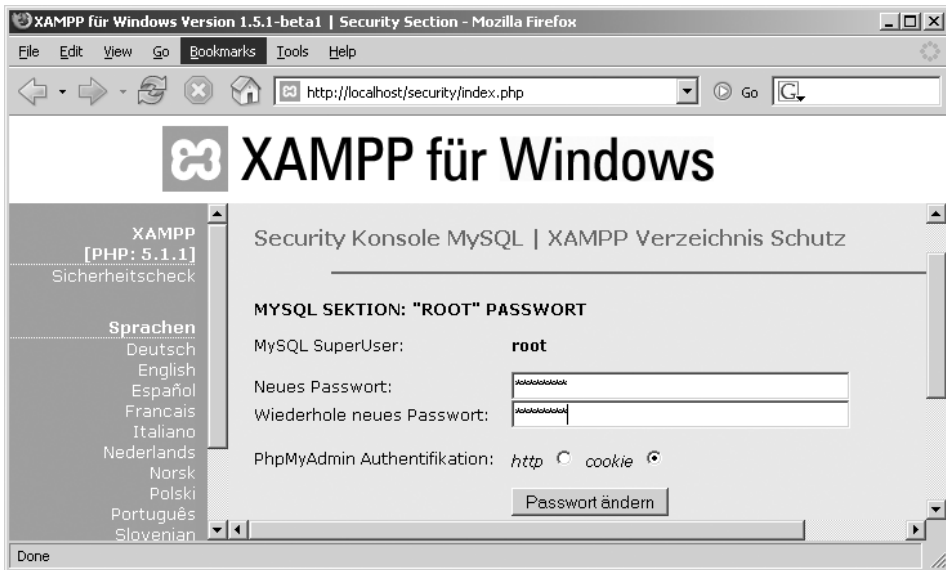


Abbildung 10.7: Die webbasierte XAMPP Security Konsole ist nur über *localhost* abrufbar.

Tipp



Sollten Sie mit der XAMPP Security Konsole arbeiten, können die Passwörter als Klartext-Dateien archiviert werden:

- Passwort für *root* (MySQL): `<xampp>\security\mysqlrootpasswd.txt`
- Passwort für die *.htaccess*: `<xampp>\security\xamppdirpasswd.txt`

Dieser Ort liegt zwar außerhalb des öffentlich zugänglichen Dokumentenverzeichnisses. Trotzdem ist es ratsam, beide Dateien auf einer besonders geschützten Ebene zu sichern. Oder Sie verschlüsseln diese Dateien auf Ihrem System. Hierfür führen Sie einen Rechtsklick mit der Maus auf die Zieldatei aus und wählen **EIGENSCHAFTEN > ALLGEMEIN > ERWEITERT** aus. Damit gelangen Sie in ein neues Menü, das unter anderem ein Optionsfeld zur Verschlüsselung von Dateien enthält.

5. Deaktivieren Sie alle nicht notwendigen Apache-Funktionen wie *SSL* oder *WebDAV*. Hierfür kommentieren Sie die entsprechenden Konfigurationsdateien in der `httpd.conf` aus.

- Konfigurationsdatei: `<xampp>\apache\conf\httpd.conf`

Beispiel einer auskommentierten SSL-Konfiguration:

```
# Include conf/extra/httpd-ssl.conf
```

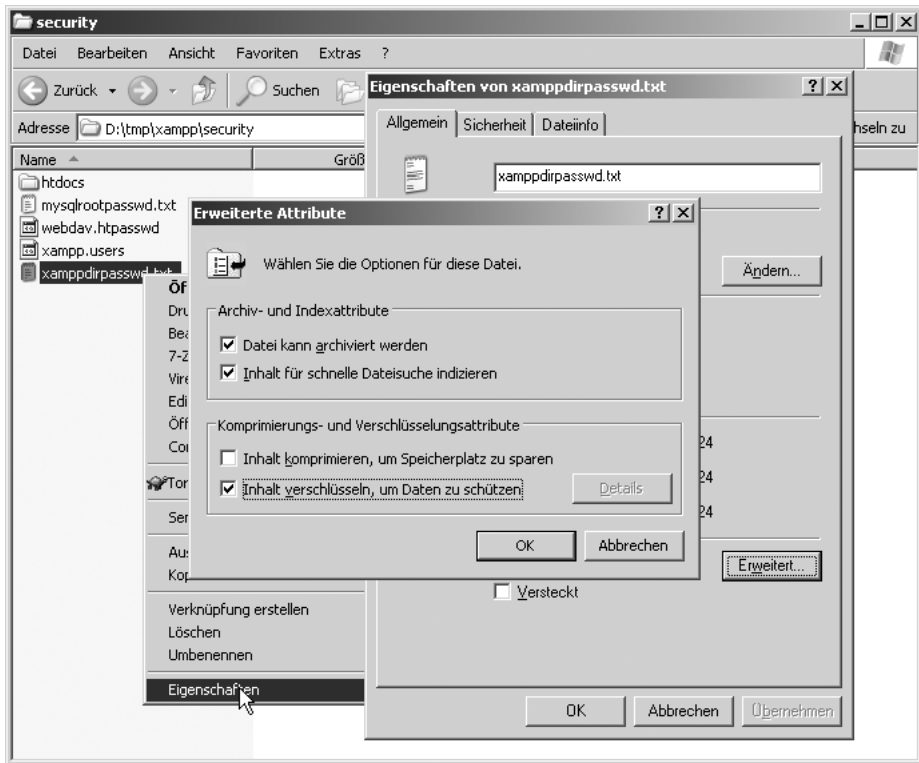


Abbildung 10.8: Verschlüsselung von Dateien über das Windows-System

6. Falls Sie *FileZilla FTP Server* benutzen, löschen oder ändern Sie bitte die bereits vorhandenen Benutzerkonten, insbesondere natürlich die Passwörter. Überlegen Sie sich, ob Sie das Angebot von *Anonymous FTP* auf Ihrem Server benötigen. Mehr Informationen zur Konfiguration des *FileZilla Servers* erhalten Sie in Kapitel 6, »FileZilla Server«.
7. Für den geplanten Einsatz des *Mercury Mail Servers* sollten vorweg auch hier alle beispielhaften Benutzerkonten verändert oder gelöscht werden. Weitere Hinweise für die Konfiguration des *Mercury Mail Transport Systems* erhalten Sie in Kapitel 7, »XAMPP und E-Mail«.

10.3.2 Benutzerkonten für die XAMPP-Server

Normalerweise laufen die Server im XAMPP über das *Systemkonto*. Das Systemkonto benutzt dabei seine administrativen Rechte. Das ist oft nicht wünschenswert. Zu empfehlen ist dagegen die Ausführung über ein *normales Benutzerkonto*. Das Konto hat dann zwar das Recht, seine eigenen Komponenten zu modifizieren, kann aber eben nicht unmittelbar Systembestandteile verändern. Am Beispiel des Apache 2-Dienstes schildern wir nun, wie Sie diesem Serverdienst ein normales Benutzerkonto zuweisen.

Hinweis



Bei Linux ist diese Konfiguration nicht notwendig, da dort die Server unter der Benutzerkennung *nobody* gestartet werden. Unter Linux ist dieser Benutzer genau für diesen Zweck bestimmt und hat absichtlich möglichst wenig Rechte.

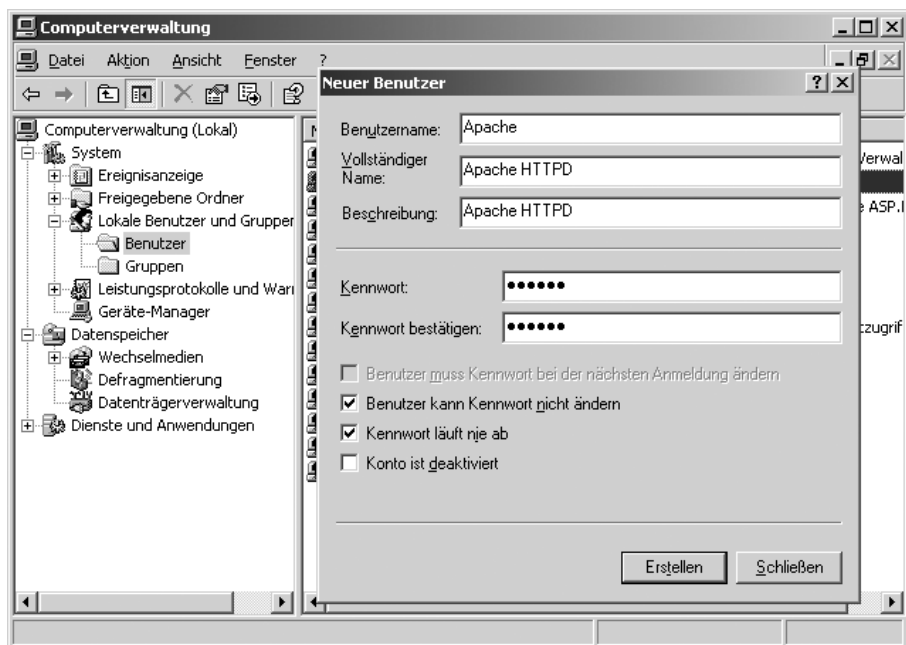


Abbildung 10.9: Das neue Hauptbenutzer-Konto Apache

1. Richten Sie ganz normal den Apache 2 als Dienst ein.
2. Richten Sie unter SYSTEMSTEUERUNG > VERWALTUNG > COMPUTERVERWALTUNG > BENUTZER einen neuen *Hauptbenutzer* ein. In diesem Fall nennen wir diesen Benutzer *Apache* und vergeben ihm ein nicht veränderbares Passwort.
3. Unter PROFIL richten Sie für den neuen Benutzer noch seinen *Basisordner* ein. Dieser ist identisch mit dem Apache-Verzeichnis im XAMPP.
4. Jetzt passen Sie den *Apache 2-Dienst* an. Dafür öffnen Sie die Dienste-Verwaltung unter SYSTEMSTEUERUNG > VERWALTUNG > DIENSTE. Unter EIGENSCHAFTEN und ANMELDEN ordnen Sie das neue Benutzerkonto dem Dienst zu.
5. Starten Sie den Dienst neu.

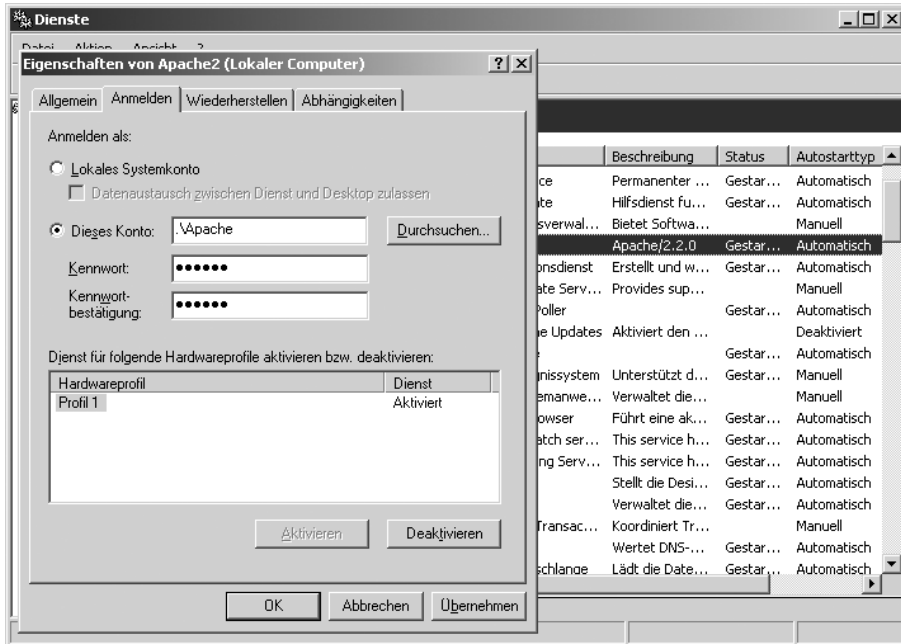


Abbildung 10.10: Das neue Benutzerkonto *Apache* für den Dienst *Apache 2*

10.3.3 Firewalls

Firewall ist nicht gleich Firewall. Eine einfache Firewall-Lösung blockt nicht legitimierte Anfragen von außerhalb. Komplexere Firewall-Lösungen beobachten darüber hinaus auch die Verbindungen des lokalen Rechners nach außerhalb. Diese Kontrolle wird dann notwendig, wenn der Server zugleich auch als Arbeitsplatzrechner oder Gateway fungiert. Denn die meisten Angriffe erfolgen durch sog. *Backdoor-Programme*. Besonders bekannt sind Trojaner und Viren, die über E-Mail auf den lokalen Rechner gelangen und dort über einen internen Port ihr Unwesen treiben. Ein solches Szenario stellt im strengen Sinne keinen externen Portangriff dar.

Bei den meisten Personal Firewalls oder kommerziellen Softwarelösungen handelt es sich um umfassende Applikationen zum Schutz des PC. Einfachere »Portblocker« sind dagegen vor allem in vielen Hardware-Routern zu finden.

Brauchen Sie einen umfassenden Schutz? Wenn Sie Ihren Rechner wirklich nur als Server und damit weder als Arbeitsplatzrechner noch zusätzlich als Gateway einsetzen wollen, reicht generell ein einfacher Blocker für die betroffenen Ports aus. Eine *Personal Firewall* kombiniert mit einem *Virens Scanner* reicht in der Regel auch bei einem darüber hinausgehenden Einsatz aus. Ob überhaupt und wenn ja, welcher Schutz dann notwendig wird, hängt aber grundsätzlich von den folgenden Fragen ab:

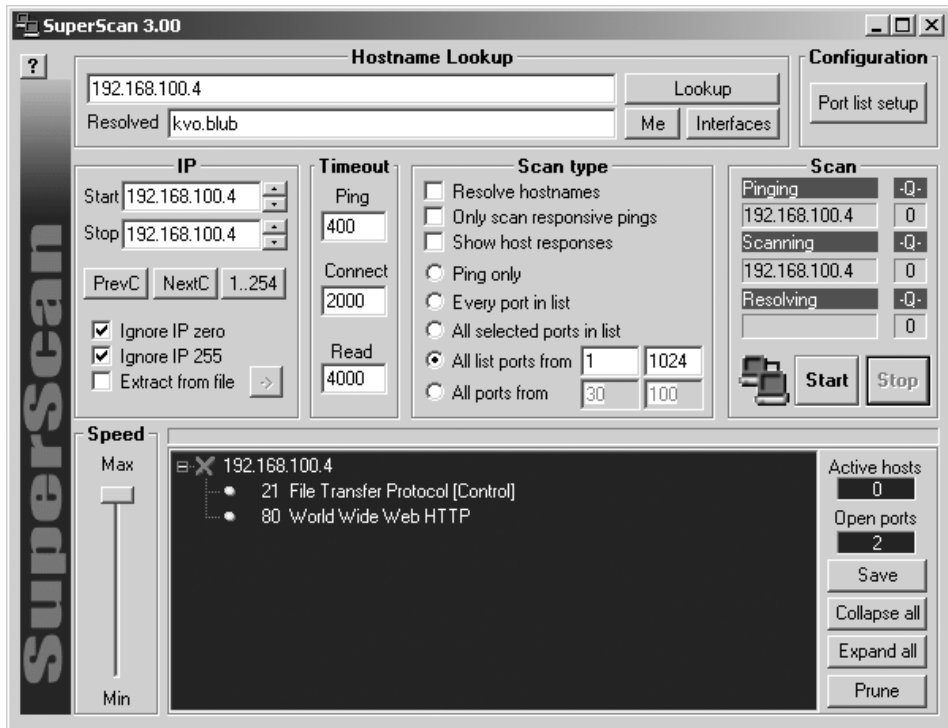


Abbildung 10.11: Zur Kontrolle ein Portscan im Netzwerk des Rechners kvo

- Welchen Wirkungsgrad soll der Server haben (Lokal, Intranet, Internet)?
- Welche Serverprozesse sind aktiv? Und welche sind als kritisch einzuschätzen?
- Wer kann direkt auf das System zugreifen? Beispielsweise über SSH oder FTP?

Wie testen Sie Ihren Rechner? Zuallererst mit einem Portscanner. Ein nennenswerter Portscanner für Windows heißt *SuperScan* von McAfee Foundstone (<http://www.foundstone.com>). Aber auch ohne zusätzliche Software kann über den Browser ein umfangreiches Scanning gestartet werden, beispielsweise über die *Sygate Online Services* (<http://scan.sygatetech.com>).

Die Windows XP-Firewall

Die *Windows XP-Firewall* ist ein integrierter Bestandteil des *XP Service Pack 2*. Sie finden das Programm unter SYSTEMSTEUERUNG > WINDOWS FIREWALL. Das Programm unterscheidet dabei wie bei allen Personal Firewalls die Programmebene und die Portebene. Wird beispielsweise der MySQL-Server gestartet, fragt Sie die Firewall nach der Legitimation des Prozesses für die Belegung des Ports 3306. Sie können nun der Applikation über NICHT MEHR BLOCKEN eine generelle Genehmigung für die lokale Belegung dieses Ports geben. MySQL erscheint danach in der *Ausnahmeliste*.

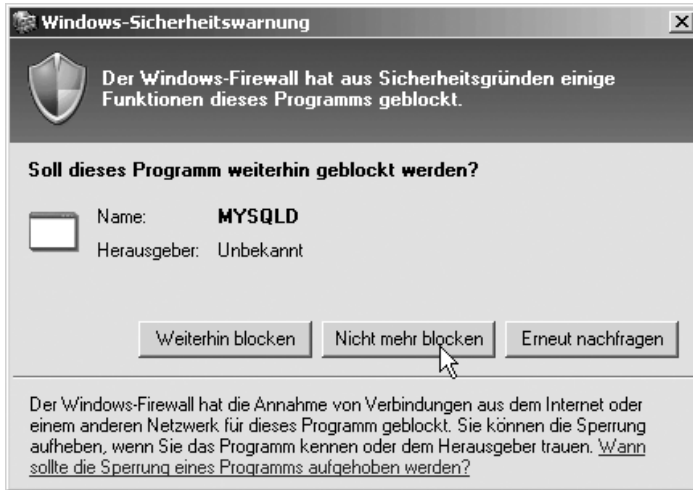


Abbildung 10.12: MySQL der Ausnahmeliste hinzufügen

Um auch öffentlich erreichbar zu sein, müssen Sie Filterregeln für Ihre Netzwerkschnittstellen definieren. Haben Sie zwei Netzwerkkarten und eine PPoP-Verbindung in das Internet, können Sie die Filterregeln der drei Verbindungen über Eigenschaften separat definieren. In den erweiterten Einstellungen können Sie dann Regeln für die Dienste TCP (*Transmission Control Protocol*) und UDP (*User Datagram Protocol*) sowie ICMP (*Internet Control Message Protocol*) bestimmen. ICMP fungiert als Diagnose-segment im Netzwerk, bekannt vor allem durch den sog. »Echo Reply« Ping. Aktivierungen von Anwendungsschichten aus diesem Segment sind für die freizugebenden TCP- bzw. UDP-Anwendungen nicht notwendig.

Sie können nun nach der Porttabelle, die in diesem Kapitel eingangs aufgeführt wurde, Ihre benötigten Dienste freischalten. Denken Sie dabei daran, UDP bei Bedarf zusätzlich aufzumachen. Leider lassen sich bei der XP-Firewall die TCP- und UDP-Ports nicht gleichzeitig öffnen. Auch konnte bei unseren Versuchen keine *Port-Range* eingerichtet werden.

Achtung



Die XP-Firewall definiert in bestimmten Fällen mögliche Datei- und Druckerfreigaben über die Ausnahmeliste global und damit für alle Schnittstellen. Jeder Internetnutzer könnte in so einem Fall direkt auf Ihren Rechner zugreifen (Quelle: *Heise News*, 16.09.2004).

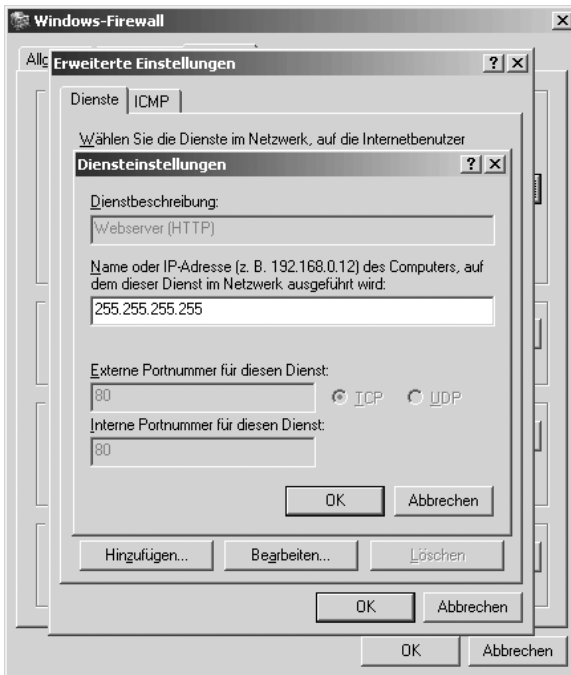


Abbildung 10.13: Der Port 80 ist nun für alle IP-Nummern dieser Schnittstellen erreichbar.

Alternative Firewall-Lösungen

Natürlich gibt es neben der Windows XP-Firewall auch umfassendere und bisweilen benutzerfreundlichere Softwarelösungen. Allerdings neigen viele Programme mit der Zeit zur Kommerzialisierung. Ein Beispiel hierfür ist die ehemals freie *Kerio Personal Firewall* Ende des Jahres 2005 vom Anbieter Sunbelt übernommen, erscheint sie nun kostenpflichtig als *Sunbelt Kerio Personal Firewall*. Anfang des Jahres 2006 waren u. a. zwei *Personal Firewalls* kostenfrei erhältlich:

- *ZoneAlarm* von Zone Labs: <http://www.zonelabs.com>
- *Schönherr Personal Firewall*: <http://www.gratis-firewall.de>

Bei der Schönherr Personal Firewall handelt es sich um ein deutschsprachiges Produkt, weshalb wir das Programm für alle vielleicht nicht so sprachbewanderten Leser erwähnen.

Eine weitere Filterlösung wäre der Einsatz einer sog. *Hardware-Firewall*, häufig integriert in externe Router. Je nach Ausstattung Ihres Routers sowie den Anforderungen in Ihrem Netzwerk kann so eine Lösung durchaus eine bequeme und ausreichende Schutzmaßnahme darstellen. Ein Beispiel für die Einrichtung eines Routers erhalten Sie übrigens in Kapitel 11, »Online mit DynDNS«.

Lokal oder Remote?

Mitte des Jahres 2003 war es so weit. Der *SQLSlammer* schlug zu. Dieser Wurm nutzte dabei eine Schwachstelle im ungepatchten *Microsoft SQL 2000 Server*. Über eine massive Attacke (d.h. mit massiver Trafficeinwirkung) auf dem UDP-Port 1434 riss er eine Lücke über einen Buffer Overflow im Serverprozess. Hatte der Wurm die Kontrolle über den Rechner erlangt, versuchte er über die neu eroberte Basis ebenfalls weitere MS SQL Server zu infiltrieren.

Dieses Szenario wäre nicht eingetreten, wäre der Port geschlossen gewesen. Damit ist das Beispiel ein klares Plädoyer für einen restriktiven und lokal beschränkten Zugriff. Öffentliche Dienste (*Remote-Server*) muss es aber ebenso geben. Ein ausschließlich lokaler HTTP-Dienst macht meist kaum Sinn. Für alle Remote-Anwendungen gilt jedoch auch: Die Gefahr z. B. von DoS-Attacken (*Denial of Service*) lässt sich nicht komplett ausschließen. Allerdings ist das Gefährdungspotential unterschiedlich. SQL Server mit ihrem Authentifizierungssystem sind meist anfälliger für feindliche Übernahmen als die hiervor weitestgehend gefeiten HTTP-Server.

Passwörter

Bei kritischen Anwendungen mit einem öffentlichen Login sind sichere Passwörter Pflicht. So versucht (vermutlich) bis heute ein Klon des Wurms *Voyager Alpha Force* sich mit verschiedenen Passwörtern als Systemadministrator (*sa*) auf fremden MS SQL Servern anzumelden. Dabei benutzt er nicht zuletzt die Passwort-Liste der bereits infizierten Microsoft SQL Server.

Im Umgang mit Passwörtern insbesondere bei sehr sensiblen Anwendungen beachten Sie bitte die folgenden grundsätzlichen Regeln:

- Je länger das Passwort ist, desto besser. Mindestens 7 Zeichen werden empfohlen.
- Das Passwort sollte komplett inhaltsleer sein. Also keinen Namen oder eine Bezeichnung innerhalb des Kennworts verwenden.
- Das Passwort sollte aus Groß- und Kleinschreibung, Ziffern (0 bis 9) sowie nicht-alphabetischen Zeichen (wie \$, !, # und %) bestehen.

Zugegeben, nicht alle Logins werden selbst von uns nach Lehrbuch vergeben. Doch wer sehr sensible Bereiche schützen möchte, sollte die oben beschriebenen Leitlinien beherzigen.

10.3.4 Backups

Eine beispielhafte Ausgangssituation: Ihr XAMPP existiert unter `D:\Programme\xampp` und Ihr Sicherungsverzeichnis lautet `D:\sicherung`. Das Komprimieren von Verzeichnissen erfolgt über den Konsolenmodus von 7-zip (<http://www.7-zip.org>). In diesem Fall liegt das 7-zip-Programm als `C:\Programme\7-Zip\7z` vor.

**Tipp**

Eine Alternative zum 7-zip-Programm wäre der Einsatz der *Unix Tools für Windows* (<http://unxutils.sourceforge.net>) und damit eine kombinierte Ausführung der Programme `tar.exe`, `gzip.exe` und `gunzip.exe`.

Automatisiertes Backup ausgewählter Bereiche im laufenden Betrieb

Auch im laufenden Betrieb können Sicherungen von bestimmten Bereichen vorgenommen werden. Zum einen können Sie ein Backup des gesamten `htdocs`-Verzeichnisses erstellen. Zum anderen ist eine unkritische Sicherung aller *MySQL*-Datenbanken mit `mysqldump` möglich.

1. Batch-Datei für das Backup anlegen

Legen Sie eine Batch-Datei z.B. mit dem Namen `segment-sicherung.bat` an. Sie hat den folgenden Inhalt:

Listing 10.2: Die Batch-Datei `segment-sicherung.bat`

```
D:\Programme\xampp\mysql\bin\mysqldump --user=root --password= --all-databases >
d:\sicherung\alle_datenbanken.sql
C:\Programme\7-Zip\7z a -t7z "D:\sicherung\htdocs.7z" "D:\Programme\xampp\htdocs\*" -
r -m0=BCJ -m1=LZMA:d=21 -ms -mmt
```

```
C:\WINNT\System32\cmd.exe
Compressing webalizer\webalizer.conf
Compressing webalizer\webalizer.exe
Compressing webalizer\webalizer.html
Compressing webalizer\webalizer.php
Compressing webalizer\webalizer.png
Compressing webalizer\zlib1.dll
Compressing wehdav
Compressing wehdav\index.html
Compressing wehdav\wehdav.txt
Compressing xampp-changes.txt
Compressing xampp-control.exe
Compressing xampp-portcheck.exe
Compressing xampp_restart.exe
Compressing xampp_start.exe
Compressing xampp_stop.exe

Everything is Ok
C:\>
```

Abbildung 10.14: Die Kompression mit 7-zip auf der Kommandozeile

Mit der ersten Zeile *dumpen* Sie alle Datenbanken Ihres MySQL-Servers. Hier hat der Benutzer *root* kein Passwort. Ansonsten ergänzen Sie diese Option mit Ihrem gültigen Passwort. Die zweite Zeile packt das gesamte *htdocs*-Verzeichnis als 7-zip-Format mit *LZMA-Kompression*. Beide Backups werden in diesem Fall als *alle_datenbanken.sql* sowie *htdocs.7z* im Ordner *D:\sicherung* gespeichert.

Tipp



Bevorzugen Sie anstatt *7-zip-Archiv* ein einfaches *ZIP-Archiv*? Dann verändern Sie bitte die 7-zip-Anweisung nach dem folgenden Muster:

```
C:\Programme\7-Zip\7z a -tzip "D:\sicherung\htdocs.zip"
"D:\Programme\xampp\htdocs\*" -r
```

2. Automatisieren

Nachdem Sie die Batch-Datei manuell erfolgreich getestet haben, ist es an der Zeit, einen automatisierten Task zu planen. Hierfür richten Sie über *SYSTEMSTEUERUNG > GEPLANTE TASKS* einen neuen Task über den Assistenten ein. Bitte denken Sie im Vorfeld daran, dass alle Tasks über die Rechte eines Benutzerkontos ausgeführt werden. Ein solches Konto sollte dann auch über die für eine Ausführung notwendigen Rechte verfügen. Bei der Frage nach dem zeitlichen Intervall markieren Sie beispielsweise *täglich*. Eine noch differenziertere Zeitangabe können Sie erst nach der Einrichtung über die *EIGENSCHAFTEN* eines fertigen Tasks definieren. Zum Abschluss starten Sie Ihren soeben erstellten Task über *AUSFÜHREN*, um die korrekte Funktion zu überprüfen.

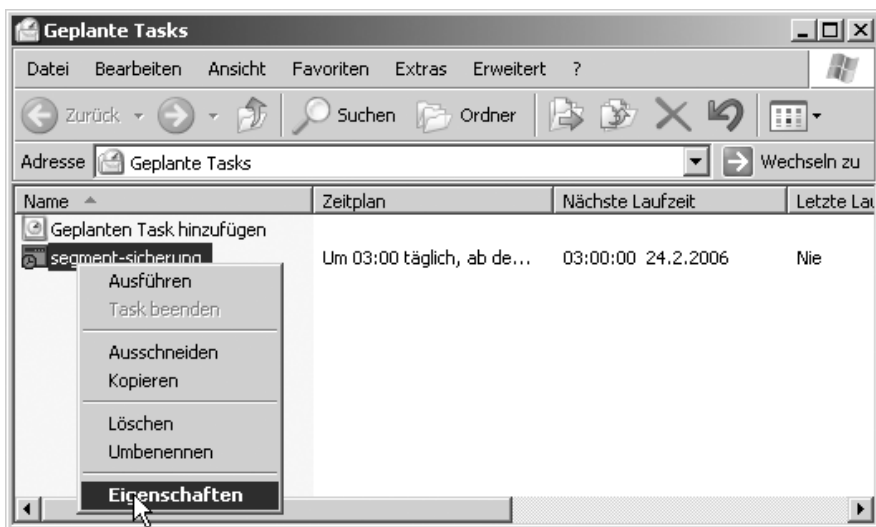


Abbildung 10.15: Über *Eigenschaften* die Feineinstellungen Ihres Tasks festlegen

3. Wiederherstellen

Um bei Bedarf den *SQL-Dump* von *MySQL* auf Ihren Server einzulesen, führen Sie auf der *Kommandozeile* Folgendes aus:

```
D:\Programme\xampp\mysql\bin\mysqldump --user=root --password= --
all-databases < d:\sicherung\alle_datenbanken.sql
```

Um die *htdocs.7z* wiederherzustellen, können Sie selbstverständlich die GUI von *7-zip* benutzen. Wir beschreiben hier aber die Anweisung über die Konsole, mit der alle Dateien ungefragt mit der Sicherung überschrieben werden (Option *-aoa*).

```
C:\Programme\7-Zip\7z x D:\sicherung\htdocs.7z -
oD:\Programme\xampp\htdocs -r -aoa
```

Scheduler mit komplettem Backup

Stellen Sie sich bitte das folgende Szenario vor: Sie haben *Apache*, *MySQL* und *FileZilla Server* als Dienste installiert. Außerdem haben Sie auch das *Mercury Mail System* permanent gestartet. Nun möchten Sie ein komplettes Backup vom *XAMPP* erstellen. Hierfür müssen allerdings für kurze Zeit alle Server und Programme gestoppt sein.

1. Batch-Datei für das Backup anlegen

In diesem Skript stoppen Sie die oben erwähnten vier Server, packen den *XAMPP* über *7-zip* und starten danach alle Server neu.

Listing 10.3: Die Batch-Datei komplette-sicherung.bat

```
net stop mysql
net stop apache2
net stop "FileZilla Server"
D:\Programme\xampp\apache\bin\pv -f -k mercury.exe -q
C:\Programme\7-Zip\7z a -t7z "D:\sicherung\xampp.7z"
"D:\Programme\xampp\*" -r -m0=BCJ -m1=LZMA:d=21 -ms -mmt
net start mysql
net start apache2
net start "FileZilla Server"
D:\Programme\xampp\MercuryMail\mercury.exe
```

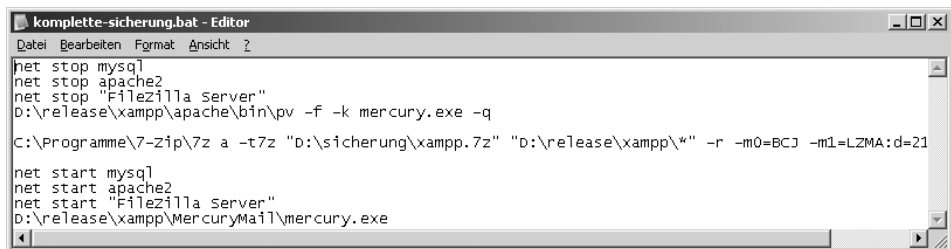


Abbildung 10.16: Die Batch-Datei für eine komplette Sicherung im Editor

2. Automatisieren

Nachdem Ihre Batch-Datei zu Ihrer Zufriedenheit geprüft wurde, erstellen Sie wieder einen neuen Task über den Task-Planer (GEPLANTE TASKS). Sie gehen hier so vor wie im vorherigen Abschnitt beschrieben. Als Intervall könnten Sie beispielsweise täglich um 3:30 Uhr angeben. In dieser Zeit ist in den europäischen Gefilden »Tiefschlafphase«, so dass ein ca. 5-minütiger Ausfall Ihrer gesamten XAMPP-Server durchaus verkraftbar erscheint. Die Zeitspanne der kompletten Ausführung dieses Tasks ist selbstverständlich abhängig von der Hardwareausstattung Ihres Rechners.

3. Wiederherstellen

Für eine komplette Wiederherstellung sollten Sie unbedingt beachten, dass alle Server und Prozesse des XAMPP gestoppt bzw. beendet sind. Danach können Sie den XAMPP insgesamt mit dem Backup überspielen, und zwar von der Konsole mit 7-zip:

```
C:\Programme\7-Zip\7z x D:\sicherung\xampp.7z -oD:\Programme\xampp -r -aoa
```

Bitte vergessen Sie jetzt nicht, alle Server erneut zu starten.

Sicherungen per FTP beziehen

Nun haben Sie eine Backup-Lösung unter Linux oder Windows etabliert, möchten aber die Backups zusätzlich auf einen weiteren Rechner vorhalten. Hierfür bietet sich ebenfalls eine Durchführung mit Hilfe einer Batch-Datei an.

1. Batch-Datei für den FTP-Zugriff

Eigentlich werden hier zwei Dateien benötigt, erstens die eigentliche Batch-Datei (hier `get-meindomain.bat` genannt) mit folgendem Inhalt:

```
ftp -i -s:D:\sicherung\get-meinedomain.ftp
```

Hiermit wird das FTP-Konsolenprogramm angewiesen, eine externe Datei als Quelle für das Login sowie den gesamten Transfer zu benutzen: Die zweite Datei `D:\sicherung\get-meinedomain.ftp` hat den folgenden Inhalt:

```
OPEN meinedomain.de
Benutzername
Passwort
LITERAL pasv
Binary
Get meinxampp.tar.gz
BYE
```

2. Automatisieren

Die Automatisierung vollziehen Sie erneut über den *Task-Planer*, wie in den Abschnitten zuvor bereits dargestellt.

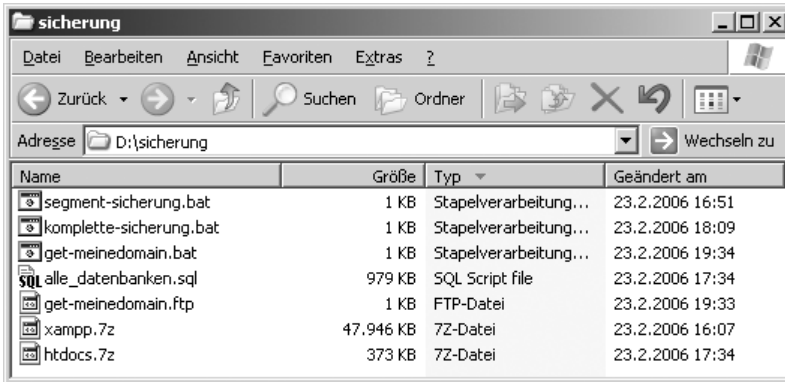


Abbildung 10.17: Das Sicherungsverzeichnis

10.4 »Moderne« Gefahren

Wirft man einen Blick in die einschlägige Sicherheitsliteratur, dann geht es dort hauptsächlich um sicherheitsrelevante Konfigurationen – wie zum Beispiel gute Passwörter oder Firewalls – und um potentielle Sicherheitslücken in der Serversoftware. Die Rede ist dann oft von *Buffer Overflow-Fehlern* oder *Denial-of-Service-Angriffen*.

Aus Sicht des Anwenders – und in diesem Fall nicht des Endanwenders, sondern des Systemadministrators oder Softwareentwicklers – sind diese Sicherheitslücken eher von passiver Natur. Sicher sind sie präsent und gefährlich, aber dem Administrator ist es prinzipiell egal, warum eine Software unsicher ist. Alles, was ihn interessiert, ist, wie er sein System wieder sicher bekommt. Ähnlich geht es dem Entwickler: Einen Buffer Overflow-Fehler im Apache nimmt er gerne zur Kenntnis, aber es ist ja nichts, was er entwickelt hat. Warum sollte es ihn also kümmern?

Viel wichtiger für diese Zielgruppe der XAMPP-Benutzer sind die folgenden, nennen wir sie mal: *modernen Gefahren* – *SQL-Injections* und *Cross-Site Scripting (XSS)*.

10.4.1 SQL-Injections

SQL-Injections nennen sich Sicherheitslücken, über die Angreifer SQL-Anweisungen von außen in die Datenbank einer Applikation schleusen können. Dadurch kann der Angreifer entweder an vertrauliche Daten gelangen, bestehende Daten manipulieren oder sogar bestehende Sicherheitsbarrieren überlisten.

Normalerweise wird PHP durch sog. *magic quotes* vor solchen Angriffen geschützt. D. h., wenn über ein HTML-Formular Daten an PHP übermittelt werden, dann werden besondere Zeichen, wie z. B. das Apostroph, durch einen Backslash (\) geschützt, so dass sie ihre besondere Bedeutung verlieren. Aber nicht immer klappt dieser Schutz und nicht für alle Wege, über die Daten in PHP gelangen können, ist dieser Schutz vorgesehen.

Als schönes Beispiel für eine solche SQL-Injection soll das PHP-Skript zur Benutzer-authentifizierung aus dem PHP-Kapitel herhalten. Das Skript in Kapitel 4 ist allerdings so programmiert, dass es nicht anfällig für eine SQL-Injection ist. Daher hier eine leicht vereinfachte Version dieses Skripts, die anfällig für SQL-Injections ist:

Listing 10.4: Ein leicht verändertes Benutzerauthentifizierungs-Skript aus Kapitel 4. Die Veränderung gegenüber Kapitel 4 ist hier fett hervorgehoben.

```
<?php
```

```

$username=$_SERVER['PHP_AUTH_USER'];
$password=$_SERVER['PHP_AUTH_PW'];
mysql_connect("localhost","root","");
mysql_select_db("cdcol");

```

```

$results = mysql_query("SELECT * FROM cds_auth WHERE username='$username' AND
password LIKE '$password'");

```

```

if(mysql_num_rows($results)==0)
{
    header( 'WWW-Authenticate: Basic realm="cds" ');
    header( 'HTTP/1.0 401 Unauthorized' );

    echo "Paßwort erforderlich!";
    die();
}

```

```
?>
```

Auf den ersten Blick sieht dieses Skript eigentlich gut und sicher aus. Wenn sich ein Benutzer mit `oswald` und `geheim` anzumelden versucht, dann wird folgende SQL-Anfrage an die Datenbank geschickt:

```
SELECT * FROM cds_auth WHERE username='oswald' AND password LIKE 'geheim'
```

Sind Benutzername und Passwort richtig, dann findet SELECT etwas. Sind sie nicht richtig, dann findet es nichts. So weit, so gut. Aber was passiert nun, wenn sich jemand mit dem Benutzernamen `egal` und anstelle des Passworts mit `x' OR '1'='1'` anmelden will?

Die SELECT-Anweisung würde dann so aussehen:

```
SELECT * FROM cds_auth WHERE username='egal' AND password LIKE 'x' OR '1'='1'
```

Da die Bedingung für dieses SELECT immer wahr ist (`1=1`), wird es immer etwas in der Tabelle finden, egal ob Benutzername und Passwort richtig sind. Und so wird der Teil `if(mysql_num_rows(...))` in der folgenden Zeile davon ausgehen, dass der Benutzer ein gültiges Benutzername/Passwort-Paar angegeben hat, und ihn »reinlassen«.

Gegenwehr

Im Originalskript aus Kapitel 4 findet man schon eine gute Lösung zu diesem Problem:

```
$username=addslashes($_SERVER['PHP_AUTH_USER']);  
$password=addslashes($_SERVER['PHP_AUTH_PW']);
```

In den Eingaben des Benutzers werden durch die PHP-Funktion `addslashes()` die Zeichen durch einen Backslash geschützt, die in der SQL-Anfrage gefährlich werden könnten. Das gefährliche Passwort von oben würde in der SQL-Anfrage nun so aussehen:

```
SELECT * FROM cds_auth WHERE username='egal' AND password LIKE 'x\' OR \'1\'=\'1'
```

Vor jedes Apostroph wird ein Backslash gesetzt und so verliert dieses Zeichen seine besondere Bedeutung.

So einfach diese »Gegenwehr« hier erscheint, ist sie aber leider oft nicht. SQL-Injections kommen heutzutage sehr oft vor, auch bei Programmen, die so einige Jahre auf dem Buckel haben und bei denen diese Sicherheitslücke nie auffiel. Oft sind die Zusammenhänge von Benutzereingaben und SQL-Anfragen nicht so direkt wie in diesem Beispiel, sondern tief im Quellcode verborgen.

10.4.2 Cross-Site Scripting (XSS)

Während eine SQL-Injection eine direkte Gefahr für einen Website-Betreiber darstellt, ist das sog. *Cross-Site Scripting* auf den ersten Blick nur eine indirekte Gefahr. In Wirklichkeit ist aber auch das XSS eine direkte und nicht zu unterschätzende Gefahr für den Betreiber.

Diese Fehleinschätzung kommt daher, dass XSS eigentlich nur ein Angriff auf einen Besucher einer Website ist. Beim XSS hat der Angreifer die Möglichkeit, über irgendwelche Lücken in der Webapplikation für andere Besucher der Website schädlichen Code (meist JavaScript) einzufügen. Dieser schädliche Code wird dann ausgeführt, wenn ein normaler Besucher die Website betritt. Schädlich ist das auf den ersten Blick nur für den unbedarften Benutzer.

Allerdings ist es in den meisten Fällen auch schädlich für den Betreiber einer Website: Der Angreifer kann so an vertrauliche Daten der anderen Benutzer kommen, zum Beispiel auch an deren Passwörter oder Session-Daten, und sich auf diese Weise Zugänge verschaffen, die er normalerweise nicht hätte.

Ein typisches Beispiel

Auf der Website X gibt es ein Gästebuch. Dieses Gästebuch ist anfällig gegen XSS. Der Angreifer A fügt in dieses Gästebuch einen gefährlichen JavaScript-Code ein. Der Redakteur R von Website X meldet sich an und wirft einen Blick in das Gästebuch.

Mit dem Aufruf des Gästebuchs wird der schädliche JavaScript-Code in seinem Browser ausgeführt. Dieser Code überträgt die Session-Daten, mit denen R angemeldet ist, an den Angreifer A. Der Angreifer A bekommt nun so die Rechte, die R hat, und kann zum Beispiel Inhalte der Website X verändern.

Dies ist ein einfaches und auch sehr realistisches Beispiel.

Gegenwehr

Ähnlich wie bei SQL-Injections müssen die Eingaben des Benutzers, die irgendwann auch anderen Benutzern gezeigt werden, auf Besonderheiten überprüft werden. Waren es bei der SQL-Injection Zeichen, die innerhalb von SQL irgendwelche besonderen Bedeutungen haben, ist es bei XSS einfach nur HTML-Code, der gefährlich sein kann.

Ein schönes Beispiel findet man in der CD-Verwaltungsdemo von XAMPP:

```
$titel=htmlentities($_REQUEST['titel']);
$interpret=htmlentities($_REQUEST['interpret']);
$jahr=htmlentities($_REQUEST['jahr']);
mysql_query("INSERT INTO cds (titel,interpret,jahr)
            VALUES('$titel','$interpret','$jahr');");
```

Bevor die Eingaben des Benutzers für Titel, Interpret und Jahr in das SQL-Insert eingesetzt werden, werden sie mit der Funktion `htmlentities()` bearbeitet. Diese Funktion sorgt dafür, dass alle Zeichen, die irgendwie eine besondere Bedeutung bekommen können, in HTML-Entities umgewandelt werden und so keine Gefahr mehr darstellen. Aus einem `<` wird zum Beispiel ein `<`.

Würde man diese Umwandlung nicht durchführen, dann könnte man in die CD-Verwaltung HTML-/JavaScript-Code einschleusen und dieser würde dann von anderen Besuchern der Website direkt gesehen bzw. in deren Browsern ausgeführt werden.



11 Online mit DynDNS

DynDNS steht allgemein für *dynamisches Domain Name System*. Damit können Rechner mit wechselnden IP-Adressen die Zuordnung des Domainnamens zur momentan verwendeten IP-Adresse dynamisch und kontinuierlich aktualisieren. Das geschieht in der Regel über eine Software für das DNS-Update (DNS-Client).

Dynamisches DNS ist nicht die Regel im World Wide Web. Normalerweise wird nach wie vor die Zuordnung eines Domainnamens über eine festgelegte IP-Adresse vollzogen, die dem Nameserver der jeweiligen Zone als A und MX Record Eintrag bekannt ist. Dieser Service wird damit auch nur von wenigen Anbietern kostenlos zur Verfügung gestellt.

Mehr Informationen über das DNS erhalten Sie in Kapitel 25, »Der BIND-Nameserver«.

Sie benötigen dynamisches DNS also insbesondere dann, wenn Sie einen oder mehrere Dienste öffentlich anbieten wollen, die aus den verschiedenen Gründen nur über lokale Rechner sowie eine »normale« ADSL-Verbindung verfügbar sein sollen.

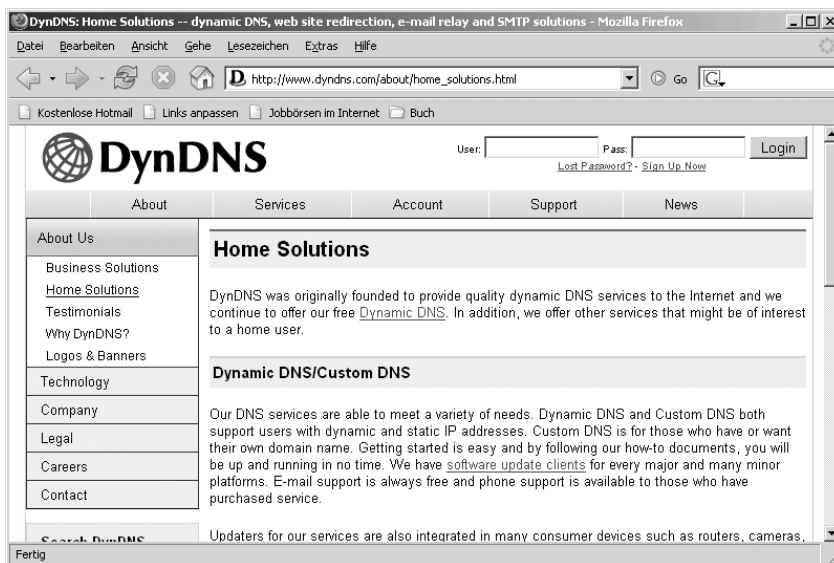


Abbildung 11.1: Kostenfreie Heimlösungen gibt es z.B. bei DynDNS.

Folgende Anbieter bieten grundlegende dynamische IP-Dienste kostenfrei an:

- DynDNS: <http://www.dyndns.com>
- No-IP.com: <http://www.no-ip.com>

Wir arbeiten in diesem Kapitel exemplarisch mit dem kostenfreien Dienst des bekannten Anbieters *DynDNS.com*.

11.1 Überlegungen

Wenn Sie mit dem Gedanken spielen, mit *DynDNS* Ihren Rechner bzw. Server öffentlich zu machen, müssen zwei grundsätzliche Bedingungen erfüllt sein.

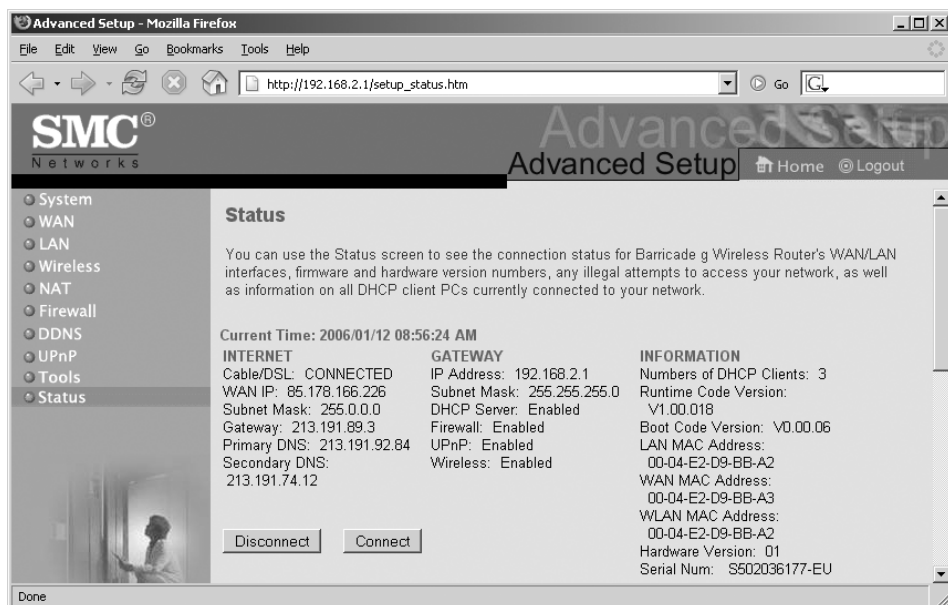


Abbildung 11.2: Die aktuelle IP im WAN lautet 85.178.166.226. NAT wird auch unterstützt.

- Sie besitzen einen Internetanschluss

Das erscheint trivial. Trotzdem nur ein kurzer Hinweis, dass es sich bei einer von außen kommenden Anfrage zu Ihrem Rechner um sog. Upstream handelt. D. h., der Upstream-Wert Ihrer Verbindung wird die gesamte Performance beeinflussen. Und wer sehr viel vorhat, sollte sich zudem eine Flatrate zulegen.

- Sie wählen sich über ein Modem oder einen Hardware-Router ein

Dieser Punkt ist unbedingt zu beachten. Denn wenn Sie einen Hardware-Router für den Internetzugang benutzen, sollte der mindestens Port-Forwarding unterstützen. Das bedeutet, dass Sie Dienstanfragen auf einen anderen Rechner in Ihrem

Netzwerk weiterleiten können. Dringend zu empfehlen ist auch ein integrierter DynDNS-Client. Ohne einen Client ist eine laufende Aktualisierung der aktuellen IP-Nummer schwieriger. Sie müssten das Update dann immer manuell über den DynDNS-Account auf der Webseite vollziehen. Für eine kurze Simulation reicht das aus, wäre aber für ein langfristiges dynamisches DNS-System sehr umständlich.

11.1.1 Der Test der DynDNS-Fähigkeit in drei Schritten

Bevor Sie einen Account bei DynDNS anlegen, prüfen Sie am besten zu Beginn, ob Ihr Equipment überhaupt zu dynamischem DNS in der Lage ist. Zwei Fälle werden nun behandelt:

- Der erste Fall betrifft einen Rechner, der über ein Modem direkte Internetverbindung hat.
- Im zweiten Fall hat der Rechner keine direkte Internetverbindung, sondern gelangt über einen Router als Gateway ins Netz.

Fall 1: Direkter Internetzugang

Hier sollte eigentlich alles für DynDNS passen. Wir führen trotzdem einen kleinen Test zur Feststellung der öffentlichen Erreichbarkeit durch.

1. Falls Sie eine Firewall benutzen, deaktivieren Sie diese am besten für kurze Zeit.
2. Wählen Sie sich ins Internet und starten Sie den Apache (falls noch nicht geschehen).
3. Finden Sie die IP-Adresse heraus, die Ihr Provider Ihnen momentan zugeteilt hat. Sie finden Ihre IP-Adresse beispielsweise über die Shell unter Linux oder in der Eingabeaufforderung unter Windows mit:

Linux:

```
ifconfig -a
```

Windows:

```
ipconfig /all
```

4. Geben Sie diese IP-Adresse in Ihren Browser ein.
Die URL sieht dann ähnlich aus wie <http://213.239.205.48>. Hier gelangen Sie allerdings zur Homepage von www.apachefriends.org. Wenn bei Ihnen alles gut gegangen ist, sollten Sie stattdessen Ihre XAMPP-Startseite sehen.

Wenn das der Fall ist, ist Ihr System für die Einrichtung von DynDNS geeignet.



Abbildung 11.3: Im WWW erreichbar. Also alles klar!

Fall 2: Internetzugang über einen Router

Sie wissen das sicherlich: Router ist nicht gleich Router. Benutzen Sie einen Hardware-Router, muss dieser die Fähigkeit zum Port-Forwarding bereitstellen. Manchmal wird diese Funktionalität auch *Adressübersetzung* oder NAT genannt.

Sie finden diese Optionen in der Regel über die Konfigurationsebene Ihres Routers. Der folgende Test wurde mit einem SMC-Wireless-Router durchgeführt. Der dabei zu erreichende Rechner mit XAMPP hatte die IP-Nummer 192.168.2.164.

1. Schalten Sie Ihre Firewall (falls vorhanden) für den Test ab.
2. Definieren Sie das Port-Forwarding/NAT über den Router (siehe Abbildung 11.4).
3. Finden Sie Ihre momentane IP-Adresse im Internet heraus (zum Beispiel über Router Status).
4. Geben Sie die IP-Adresse in Ihrem Browser ein.

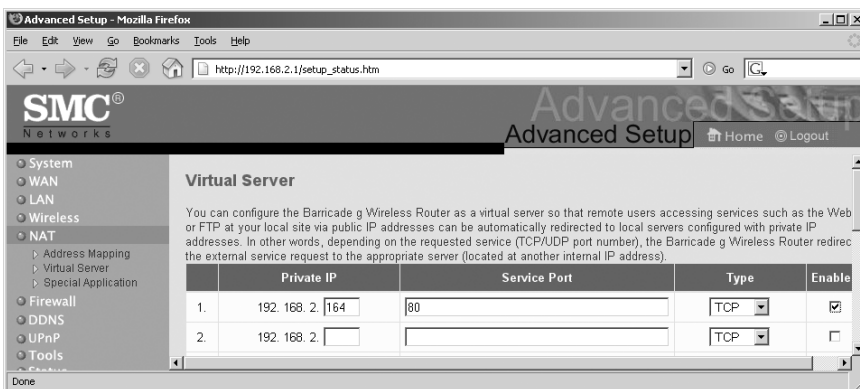
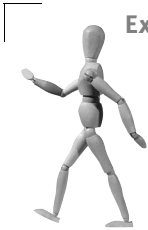


Abbildung 11.4: Das Port-Forwarding über den Router einstellen



Exkurs

Was passiert aber, wenn Sie einen Rechner als Gateway besitzen? Dann handelt es sich streng genommen um Fall 1, nämlich um eine direkte Internetanbindung. Die einfachste Möglichkeit ist also, diesen Router-Rechner mit XAMPP und der DynDNS-Software zu bestücken. Etwas kompliziert wird es dann, wenn Sie in diesem Szenario einen dahinter gelagerten Rechner ansprechen wollen. Mindestens Linux hat mit iptables einen starken Netzwerkfilter, der seit Kernel 2.4 zum »Lieferumfang« gehört. Der DynDNS-Client (z.B. *ddclient*) kommt dann auf Ihren Linux-Router. Über eine Filterregel in iptables leiten Sie dann den entsprechenden Port auf einen sekundären Rechner innerhalb Ihres Netzwerks weiter. Ein sog. »Forwarding« von Port 80 zum Zielrechner mit der IP-Nummer 192.168.2.164 könnte unter SUSE so aussehen:

```
/sbin/modprobe iptable_nat
echo 1 > /proc/sys/net/ipv4/ip_forward
iptables -t nat -A PREROUTING -p tcp --syn --destination-port 80 -j
DNAT -to -destination 192.168.2.164:80
```

Auch für Windows gibt es Softwarelösungen. Beispielsweise wird unter *www.nat32.com* ein NAT32 IP Router für Microsoft Windows angeboten. Mit diesem Programm ist sowohl IP Address Translation als auch Transparent Port Mapping möglich.

11.1.2 Sicherheit

Machen Sie sich bitte bewusst, dass Ihr Rechner öffentlich zugänglich ist oder werden soll. Und damit auch angreifbar. Wobei man die Sicherheitsgefährdung abhängig von der Netzwerktopologie unterschiedlich einschätzt. Ist Ihr zukünftiger DynDNS-Rechner hinter einem Router und Sie adressieren mit NAT, fungiert der Router mindestens teilweise als Firewall. Alle nicht weitergeleiteten Ports sind in der Regel vor direkten externen Angriffen sicher. Ist Ihr Rechner dagegen direkt im Internet, so sollte hier in jedem Fall eine Firewall schützen. Freie Desktop-Firewalls für Windows sind beispielsweise *ZoneAlarm* oder die *Schönherr Personal Firewall*. Unter Linux bieten die einzelnen Distributionen ebenfalls Personal-Firewalls an, die fast ausschließlich auf dem mächtigen iptables basieren. Mehr zum Thema XAMPP und Sicherheit erfahren Sie in Kapitel 10 »Sicherheit«.

11.2 DynDNS einrichten

11.2.1 Ein DynDNS-Konto anlegen

Zu Beginn richten Sie ein Konto bei *DynDNS* ein. Sie erhalten danach eine E-Mail mit einem Link zur Bestätigung Ihrer Anmeldung. Damit kann es nun losgehen. Loggen Sie sich bei *www.dyndns.com* ein. Übrigens: Der erste Kontakt mit dem neuen Account verwirrt oft. Viele und meist kostenpflichtige Dienste werden hier angeboten. Nein, bezahlen wollen wir nichts! Also schlagen Sie in der Navigation Ihres Accounts den Bereich für das dynamische DNS unter *MY SERVICES > MY HOSTS > DYNAMIC DNS* auf. Dort richten Sie einen neuen dynamischen Host ein. In diesem Beispiel entschieden wir uns für den Hostnamen *xamppbook* mit der Domain *dyndns.org*. DynDNS bietet ja zahlreiche mögliche Domainnamen an. Sie können hier also ganz nach Ihrem persönlichen Geschmack verfahren.

Enable Wildcard

Wenn Sie mehrere verschiedene Rechner ansprechen wollen, können Sie über Wildcards und einen eingerichteten DNS-Server das Routing intern bestimmen. Damit wäre beispielsweise *host1.xamppbook.dyndns.org* ein anderer Rechner als *host2.xamppbook.dyndns.org* usw. Gut sind Wildcards auch für nicht IP-gebundene virtuelle Hosts des Apache-Webservers.

Mail Exchanger (optional)

Der *Mail Exchanger* ist der MX-Eintrag Ihres E-Mail-Servers für den Posteingang (POP3 und IMAP). Wollen Sie einen eigenen Mailserver wie beispielsweise den Mercury in XAMPP benutzen? Dann brauchen Sie hier keinen Eintrag, wenn der E-Mail-Server auf dem Zielrechner läuft. Optional können Sie auch einen anderen E-Mail-Server angeben, der die von Ihnen verwendete Domain akzeptiert. In diesem Fall wäre auch ein *MX Update* notwendig.

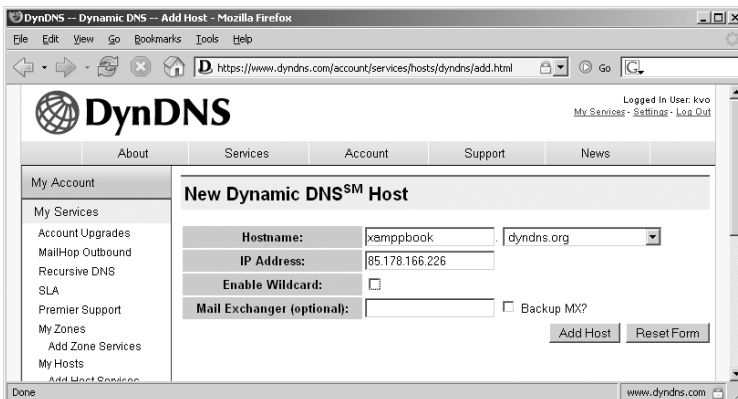


Abbildung 11.5: Die Einrichtung des dynamischen DNS Hosts bei DynDNS

11.2.2 DynDNS mit direkter PPP-Verbindung

In diesem Fall hat Ihr Zielrechner mit seinen Serverdiensten einen direkten Zugang über PPP bzw. PPPoE für DSL zum Internet. Darüber hinaus sind die Ports für die zukünftig verfügbaren Dienste in Ihrer Firewall geöffnet. Um nun ein regelmäßiges Update der Ihnen vom Provider zugewiesenen IP-Adresse zu realisieren, bietet die Homepage von DynDNS sog. *Update-Clients* an. Die folgenden Plattformen wurden Anfang des Jahres 2006 unterstützt: Windows, Linux und MacOS X.

Windows

Für dieses Beispiel verwendeten wir den *Official Windows Update Client* in der Version v3.1.0.7s. Wichtige Features waren bei dieser Version beispielsweise:

- als Dienst installierbar
- automatischer Start bei Systemstart
- auch mit einem vorgeschalteten Proxy-Server einsetzbar

Den letzten Punkt, das Routing über einen vorgeschalteten Proxy, sollten Sie bei Bedarf auf die Praxistauglichkeit überprüfen. Erfahrungsgemäß spielt hier der Typ des Proxy-Servers eine nicht unwesentliche Rolle.



Abbildung 11.6: Das Aktualisierungsintervall im DynDNS-Update

Linux

Für Linux bietet DynDNS zwei Update-Clients an: zum einen den *ddclient*, ein Perl-Programm mit Shellskript und zentraler Konfigurationsdatei. Zum anderen gibt es ein C-Programm mit dem Namen *ez-ipupdate* für das Selbstkompilieren mit dem GCC und make.

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

Hier gehen wir kurz auf den *ddclient* in der Version 3.6.6 ein. Als *su* installieren Sie wie gewohnt von der Konsole, und zwar

- mit der apt-Paketverwaltung:

```
apt-get install ddclient
```

- oder durch mehr »Handarbeit«:

```
tar xvfz ddclient.tar.gz
cd ddclient-3.6.6
cp ddclient /usr/sbin
cp sample-etc_ddclient.conf /etc/ddclient.conf
cp sample-etc_rc.d_init.d_ddclient /etc/rc.d/init.d/ddclient
```

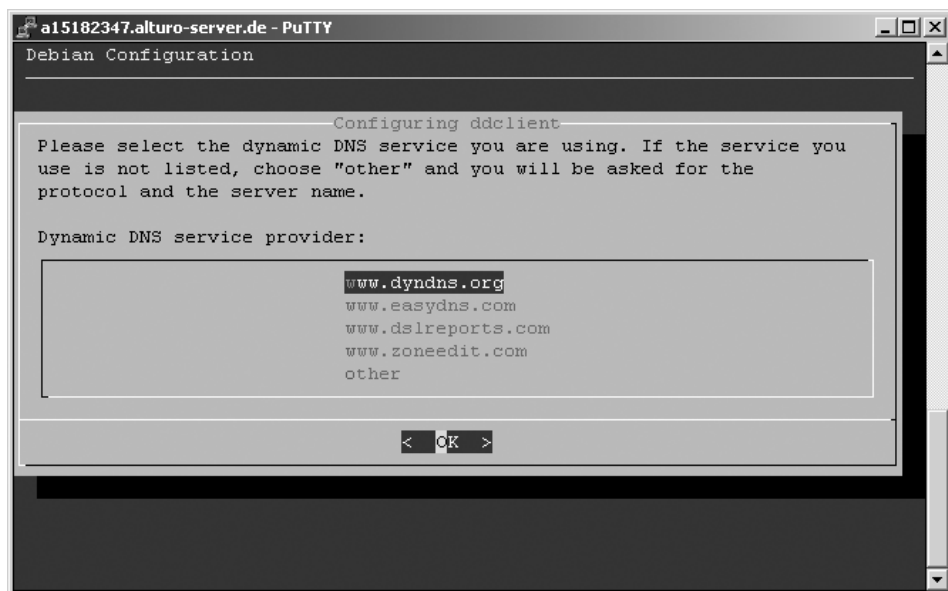


Abbildung 11.7: Eine DynDNS-Domain als Zusatzdomain auf einem Root-Server. Warum nicht?

Für alle Linux-Freunde sind diese Zeilen gleich einem offenen Buch. Das Perl-Programm kommt in das `/usr/sbin`-Verzeichnis. Die Konfiguration wird als `/etc/ddclient.conf` hinterlegt. Und mit der letzten Zeile initialisiert sich der Client zukünftig über den `initd`-Prozess. Es bleiben aber noch die Einstellungen in der `/etc/ddclient.conf` übrig.

Listing 11.1: Beispiel für die ddclient.conf

```
# Ein Beispiel für DynDNS.com
daemon=300# check every 300 seconds
syslog=yes# log update msgs to syslog
mail=root# mail all msgs to root
mail-failure=root# mail failed update msgs to root
pid=/var/run/ddclient.pid# record PID in file.
#
#use=ip, ip=127.0.0.1# via static IP's
#use=if, if=eth0# via interfaces
use=web# via web
protocol=dyndns2          # default protocol
#proxy=fasthttp.sympatico.ca:80# default proxy
server=members.dyndns.org # default dyndns server
login=kvo                 # default login
password=meinpasswort     # default password
#mx=mx.for.your.host# default MX
#backupmx=yes|no# host is primary MX?
wildcard=yes# add wildcard CNAME?
xamppbook.dyndns.org# my Host.Domain-Name
```

Sie starten den `ddclient` über folgenden Befehl:

```
# /etc/init.d/ddclient start
```

Den Erfolg oder Misserfolg der Update-Bemühungen finden Sie in dieser Konfiguration in der `Syslog`-Datei protokolliert.

11.2.3 DynDNS mit einem Hardware-Router als Gateway

Zuerst müssen Sie sich Ihre Filterregeln überlegen. In diesem Beispiel wollen wir das volle E-Mail-Programm, d.h. POP3/IMAP zum Empfangen und SMTP für den Versand sowie den Standardport des Apache HTTPD auf Port 80 auf den Zielrechner 192.168.2.164 legen. Das Port-Forwarding in Ihrem Router in einem C-Netzwerk (hier 192.168.2.255) könnte also so aussehen:

```
FTP (Data):192.168.2.164:20
FTP (Control):192.168.2.164:21
HTTP:192.168.2.164:80
POP3: 192.168.2.164:110
IMAP: 192.168.2.164:143
FTP (Port-Range passiv):30000-65535
```

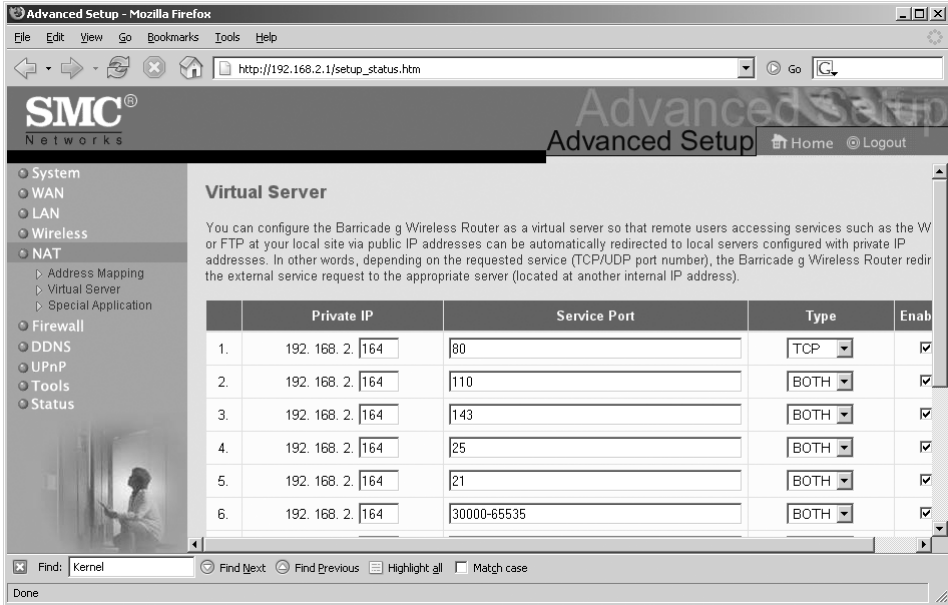


Abbildung 11.8: So mancher Port wird zum Zielrechner 192.168.2.164 durchgereicht.

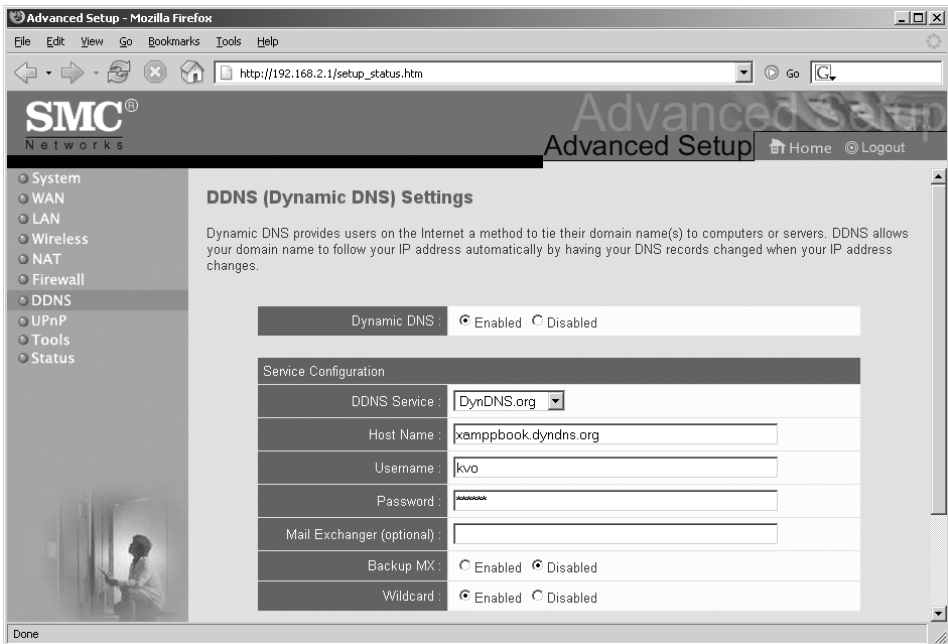


Abbildung 11.9: Praktisch! Der Router besitzt bereits einen DynDNS-Client.

Falls Ihr Router über einen DynDNS-Client verfügt, richten Sie ihn analog zu der Konfiguration unter *www.dyndns.com* ein. Ein solcher integrierter Client ist äußerst praktisch. Bei jeder Einwahl übergibt er Ihre vom Provider dynamisch zugeteilte IP-Adresse dem DynDNS-Server. Damit sollte die DNS-Auflösung immer aktuell sein.

11.3 Optimierungen

11.3.1 Forever online!

In Zeiten von Strom sparenden und immer leiseren Mini-PCs überlegt sich vielleicht so mancher, sein Angebot über DynDNS für 24 Stunden und damit ohne Unterbrechungen anzubieten. Doch das scheitert oft an den unkalkulierbaren Trennungen der Verbindung durch den Provider. Die Lösung ist einfach. Ein Cronjob (Linux) oder Taskjob (Windows) muss her für regelmäßige Anfragen im WWW. Hierfür muss auf Ihrem System oder auf dem von Ihnen genutzten Router eine automatische Internet-einwahl auf Anforderung aktiviert sein.

Ziel: Ihr Server soll alle 5 Minuten kurz ins Internet gehen. Das realisieren Sie z. B. mit der Anweisung `ping`.

Linux: `ping -c 3 www.google.de`

Hierfür gibt es die *Cronjobs*. Sehr einfach geht das übrigens mit *Webmin*, siehe Abbildung 11.10.

Windows: `ping -n 3 www.google.de`

Für eine Automatisierung unter Windows NT, 2000 und XP steht der Taskplaner-Dienst bereit. Starten Sie *Geplante Tasks* im START-Menü unter PROGRAMME > ZUBEHÖR > SYSTEMPROGRAMME. Dort legen Sie einen neuen Task für eine tägliche Wiederholung an. Ihre `ping`-Anweisung können Sie übrigens direkt unter AUSFÜHREN hineinschreiben.

Wundern Sie sich bitte nicht, dass bei der Einrichtung eines neuen Tasks keine zeitliche Feineinstellung möglich ist. Diese Konfigurationsebene erreichen Sie erst nach dem Erstellen über Rechtsklick und die Wahl EIGENSCHAFTEN im Kontextmenü (siehe Abbildung 11.11).

Hinweis



Vergewissern Sie sich auch, ob der Taskplaner-Dienst auf Ihrem System gestartet ist.

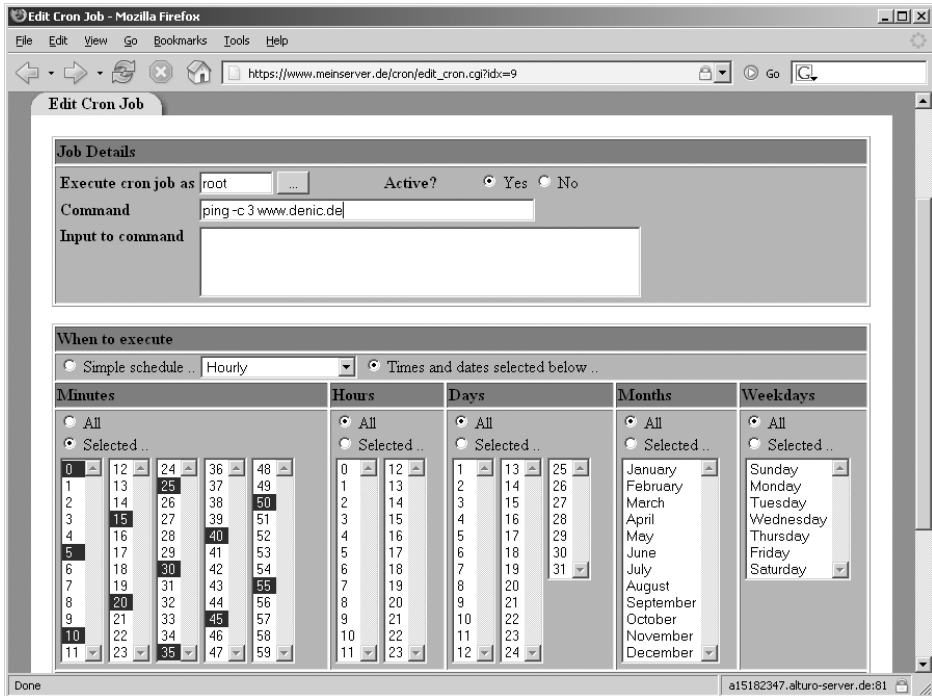


Abbildung 11.10: Ein neuer Cron Job mit Hilfe des Webmin Interfaces

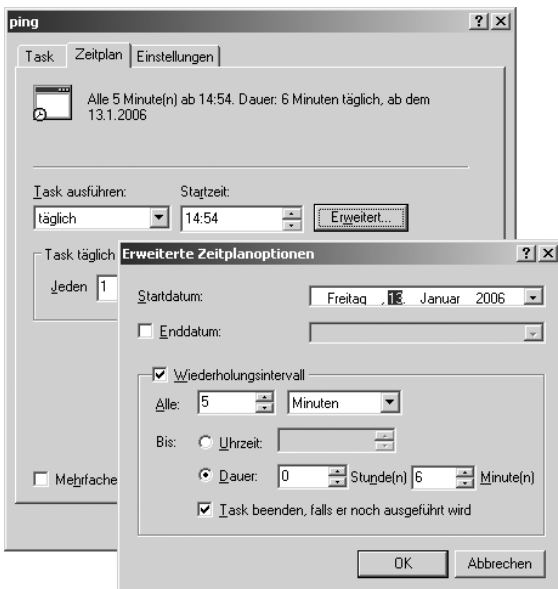


Abbildung 11.11: Der Ping-Job im Taskplaner unter Windows XP



12 XAMPP im Unterricht

Der Einsatz von XAMPP als bequemes Werkzeug für Unterricht und Lehre hat sich in den letzten Jahren immer weiter verbreitet. Der Lehrer/Dozent hat ein einfach zu administrierendes Werkzeug, das ihm das Unterrichten von Themenbereichen wie PHP, MySQL, HTML und Webadministration ermöglicht.

Insbesondere der XAMPP Lite unter Windows fällt hier positiv auf, da für den Einsatz des Apache- und MySQL-Servers die Standardbenutzerrechte ausreichen. Nach Abschluss des Kurses kann XAMPP ohne Probleme gelöscht und durch eine neue XAMPP-Version ersetzt werden. Außerdem wird den Schulungsteilnehmern der eigenständige Umgang mit dem XAMPP-Serverpaket nahe gebracht. Der private Gebrauch von XAMPP stellt für sie nach Kursende damit wohl kein Problem mehr da.

Der Nachteil dieser *dezentralen* Schulungsmethode liegt an einem Mangel an Transparenz. Gerne möchte der Dozent dem Schüler über die Schulter schauen und gegebenenfalls bei möglichen Problemen auch direkt eingreifen können.

Die Umsetzung einer *zentralen Lösung*, bei der auf dem Dozentenrechner der XAMPP allein agiert, besticht hingegen durch die Möglichkeit, direkt am Arbeitsfluss der Kursteilnehmer teilzuhaben. Da die dezentrale Methode nach den vorangegangenen Kapiteln keiner weiteren Erklärung bedarf, wollen wir uns näher mit dem komplexeren zentralen Einsatz des XAMPP im Netzwerk beschäftigen.

12.1 Zentrale Netzwerklösungen

Die in diesem Kapitel vorgestellten Lösungen funktionieren in der beschriebenen Form ohne einen zentralen Windows-Server bzw. Domain-Controller und sind in dieser Form mit einem einfachen Netzwerk aus Windows-Arbeitsplätzen realisierbar.

Das hat zwei einfache Gründe: Erstens fehlt in vielen Schulungsräumen die Möglichkeit für den Einsatz eines Windows-Servers. Und zweitens sind die hier vorgestellten Beispiele mit einem zentralen Windows-Server inklusive Domain-Controller vielfach einfacher als dargestellt einzurichten. Die Lösungen dieses Kapitels sollten also ohne Probleme in Ihre Netzwerksystematik übertragbar sein.

Hinweis**Warum nicht Linux?**

Prinzipiell ist natürlich auch alles, was in diesem Kapitel beschrieben wird, unter Linux möglich. Da Schulungscomputer nach wie vor überwiegend mit Microsoft-Software ausgerüstet sind, haben wir uns in diesem Kapitel auf ein Windows-Netzwerk beschränkt.

12.1.1 Benutzerzugriff unter Windows

Auch in der *XP Arbeitsgruppe* ohne einen *Domain-Controller*, wie er vom NT-, 2000-, XP-Server von Microsoft zur Verfügung gestellt wird, besitzen Sie die Möglichkeit, zentrale Zugriffe auf einen zentralen Rechner zu ermöglichen bzw. zu beschränken. Bitte beachten Sie dabei die Einschränkung, dass eine zentrale Netzwerkfreigabe eines Clientrechners beispielsweise durch die Betriebssysteme XP Home oder XP Professional maximal 10 unterschiedliche Benutzeranmeldungen zulässt.

Alternativ können Sie auch einen zentralen FTP-Server im XAMPP verwenden. Damit sind Sie von der Netzwerktopologie und den administrativen Gegebenheiten eindeutig unabhängiger. Für Dozenten mit Kursen an häufig wechselnden Standorten ist diese FTP-Variante im Vergleich zu einer stationären Workgroup-Installation sicherlich zu bevorzugen. Auch dieses Thema wird in diesem Kapitel behandelt.

Ziel

Sie möchten für maximal 10 gleichzeitig agierende Benutzer über einen zentralen Clientrechner 10 Arbeitsverzeichnisse zur Verfügung stellen. Der jeweilige Benutzer darf hier nur sein freigegebenes Arbeitsverzeichnis bearbeiten. Zugriffe auf die Arbeitsverzeichnisse anderer Benutzer werden verweigert. Sie dagegen auf Ihrem lokalen Dozentenrechner (dem »Quasiserver«) besitzen den vollen Zugriff auf alle Arbeitsverzeichnisse. Auf dem Dozentenrechner ist auch der XAMPP installiert. Die einfachen Clients besitzen hingegen keinen eigenen XAMPP. Über eine Konfiguration der `httpd-userdir.conf` kann der Benutzer aber Webseiten zentral veröffentlichen und über `http://Dozentenechner/~benutzername` aufrufen und somit testen. Außerdem richten Sie auf dem ebenfalls zentralen MySQL-Server jedem Benutzer eine eigene Datenbank ein. Diese darf der Benutzer unter der URL `http://Dozentenechner/phpmyadmin/` selbst administrieren.

Voraussetzungen

Sie haben alle beteiligten Rechner in derselben Arbeitsgruppe in dem gleichen Netzwerk der Klasse A, B oder C eingerichtet. Darüber hinaus müssen alle Rechner entweder über DHCP oder fest eingestellt unverwechselbare IP-Adressen besitzen. Von Vorteil wäre es, wenn alle Rechner über die Rechnernamen aufrufbar wären. Das kann optional über den DHCP-Server und dem Netbios-Protokoll geschehen oder an

jedem Rechner in der `hosts`-Datei per Hand eingetragen werden. Alternativ benutzen Sie hierfür einen eigenen DNS-Server wie *BIND*.

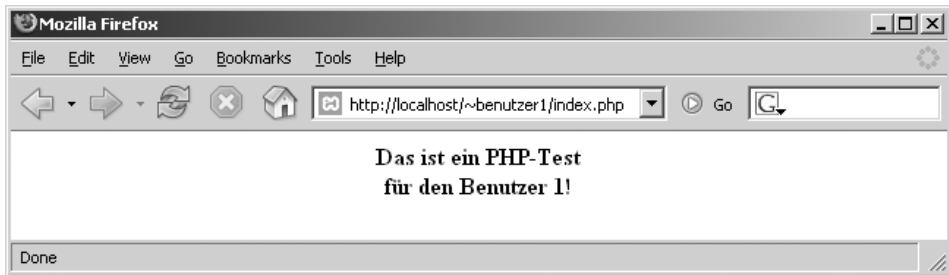


Abbildung 12.1: Erster Test mit der *Apache-usedir*-Funktion für Benutzer 1

Als Test richten Sie eine Netzwerkfreigabe auf dem zukünftigen Dozentenrechner für alle Clientbenutzer ein, die sich dort mit Kennwort anmelden müssen. Nun sollten Sie auf einem beliebigen Client darauf zugreifen können.

Sollten Sie sich für einen FTP-Server-basierten Zugriff entscheiden, reicht sicherlich die IP-Adresse des Zielrechners, d.h. Ihres Dozentenrechners, aus. Sie finden Ihre aktuelle IP-Adresse unter XP über die Eingabeforderung mit:

```
ipconfig /all
```

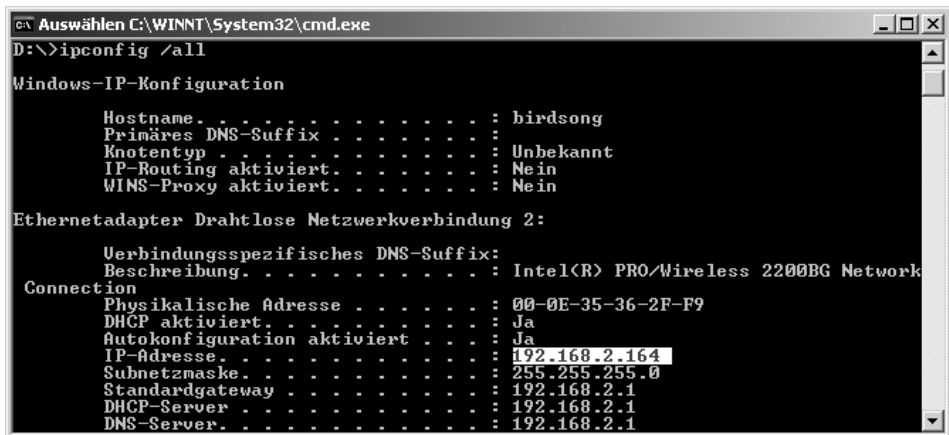


Abbildung 12.2: Die aktuelle IP-Adresse unter Windows mit *ipconfig /all* ermitteln

Die Arbeitsgruppe in der Praxis

Sie haben ein Netzwerk mit drei Rechnern: einem Dozentenrechner und zwei einfachen Clientrechnern. Auf den beiden Clientrechnern richten Sie jeweils *benutzer1* (auf Arbeitsplatzrechner 1) und *benutzer2* (auf Arbeitsplatzrechner 2) mit Passwörtern und

den Attributen *Benutzer darf Kennwort nicht ändern* und *Konto läuft nie ab* ein. Die neuen Benutzer besitzen selbstverständlich nur einfache Benutzerrechte. Beide neuen Teilnehmer richten Sie ebenfalls auf Ihrem Dozentenrechner identisch ein. Denken Sie daran: Das Kennwort muss gleich dem Passwort der Einrichtung des jeweiligen Benutzers auf Ihrem Dozentenrechner sein! Alle Kursteilnehmer müssen also auf dem Dozentenrechner angelegt werden. Auf den Arbeitsplatzrechnern reicht in der Regel die Einrichtung des zugehörigen Benutzers, der mit dem eingerichteten Benutzer (d.h. Kursteilnehmer) auf Ihrem Dozentenrechner identisch ist.

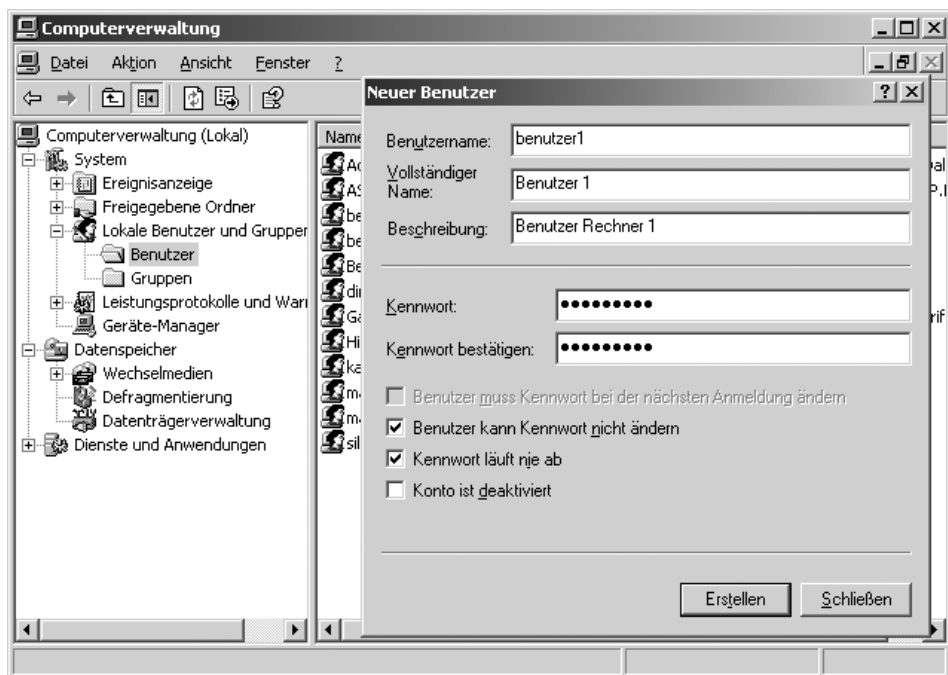


Abbildung 12.3: Die Einrichtung von Benutzer 1 auf dem Dozentenrechner



Hinweis

Wenn Sie nicht die Benutzer auf den Arbeitsplatzrechnern einrichten wollen, würde trotzdem eine Anmeldung auf Ihrem Dozentenrechner möglich sein. Hierfür melden Sie sich mit einem x-beliebigen Benutzer an dem Arbeitsplatzrechner an. Über ARBEITPLATZ wählen Sie in der Adressleiste den Computernamen oder die IP-Adresse Ihres Dozentenrechners aus. Beispiele:

\\dozentenrechner

\\192.168.100.10

Nun sollten Sie nach einem auf dem Zielrechner gültigen Benutzer und einem Kennwort gefragt werden. Hier geben Sie den Namen und das Kennwort nach folgendem Muster ein:

Dozentenrechner\Benutzer

Kennwort

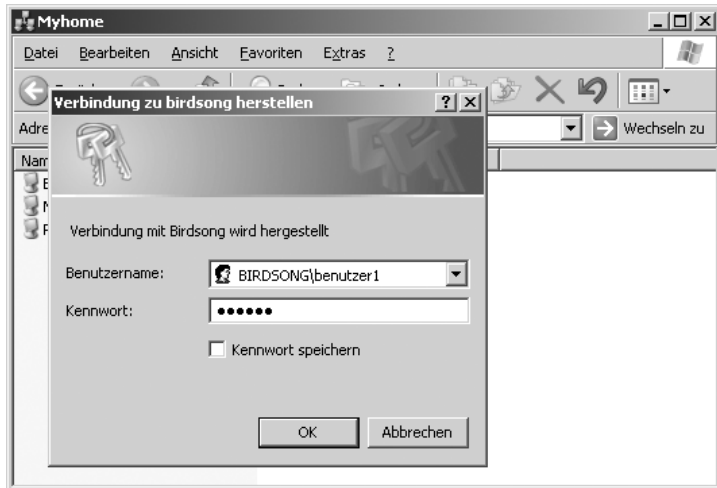


Abbildung 12.4: Anmeldung an dem Zielrechner mit neuem Benutzer

Nun gehen Sie daran, die Freigaben auf dem Dozentenrechner zu setzen. Ein Beispiel:

D:\webs

D:\webs\benutzer1

D:\webs\benutzer2

Das Arbeitsverzeichnis `d:\webs\benutzer1` (bitte nicht das Oberverzeichnis `D:\webs`) wird nun für *benutzer1* freigegeben und für *benutzer2* gesperrt. Hierfür klicken Sie mit der rechten Maustaste auf den betreffenden Ordner und suchen den Eintrag **EIGENSCHAFTEN**. Im **EIGENSCHAFTEN**-Fenster klicken Sie auf den Reiter **FREIGABE** und markieren das Optionsfeld *Diesen Ordner freigeben*. Jetzt benutzen Sie *Berechtigungen*, um neue Benutzer zuzufügen. Die Gruppe *Jeder* kann so bleiben und sollte nicht gelöscht werden. Zu Beginn fügen Sie hier *benutzer2* hinzu und geben diesem die Rechte *Verweigern: Lesen, Schreiben und Vollzugriff*. Den gleichen Vorgang, aber mit der Option *Zulassen*, richten Sie für *benutzer1* ein. Damit wird klar: *benutzer1* darf mindestens lesen und schreiben, *benutzer2* hingegen nichts. Für das Arbeitsverzeichnis für *benutzer2* konfigurieren Sie die Freigabe danach umgekehrt: *benutzer2* darf alles, *benutzer1* darf hier nichts.

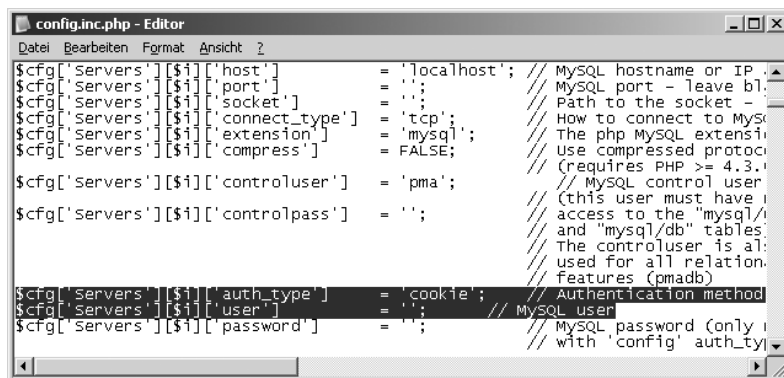


Abbildung 12.5: Das Verzeichnis auf dem Rechner Birdsong für benutzer1 aktivieren

Bei 10 Kursteilnehmern bedeutet das also einige Handarbeit auf Ihrem Dozentenrechner. Sie müssen die 10 Benutzer mit Kennwort auf Ihrem Rechner sowie auf den jeweiligen Arbeitsplatzrechnern mit identischen Kennwörtern anlegen. Danach müssen Sie 10 Arbeitsverzeichnisse aufsetzen. Der Zugriff auf jedes dieser Arbeitsverzeichnisse muss in dem Fall für 9 Benutzer verweigert und dem vorgesehenen Benutzer selbstverständlich erlaubt werden. Abschließend können Sie sich überlegen, ob Sie das freigegebene Verzeichnis für den jeweiligen Arbeitsplatzrechner als Netzwerklaufwerk einrichten. Doch so weit wollen wir mit unserer Beschreibung nicht gehen und überlassen die weiteren Feinheiten Ihnen.

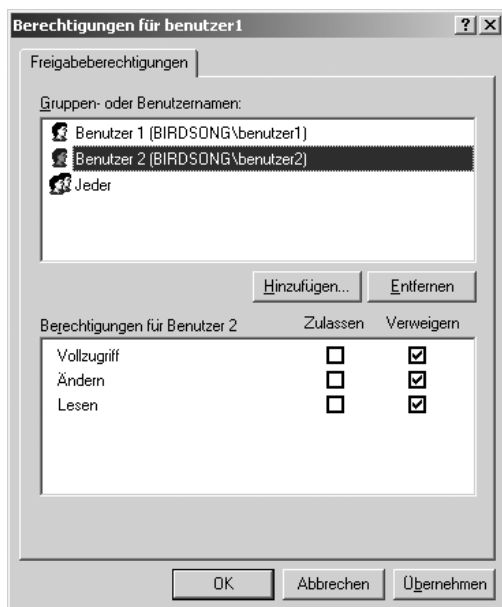


Abbildung 12.6: benutzer2 darf im benutzer1-Verzeichnis natürlich nichts!

FTP als flexiblere Alternative zur Verzeichnisfreigabe verwenden

Das ist sicherlich eine Überlegung wert. Der Vorteil liegt auf der Hand: Nicht nur die aufwendige Einrichtung entfällt. Sie sind mit einem eigenen FTP-Server beispielsweise auf Ihrem Notebook unabhängiger. Nehmen wir einmal an, Sie sind als Dozent viel unterwegs und wissen nichts über die Zugriffsmöglichkeiten der Lehrereinrichtungen, an denen Sie in Kürze unterrichten werden. Um den zuständigen Administrator nicht belästigen zu müssen, benutzen Sie Ihren FTP-Server im XAMPP, um jedem einzelnen Teilnehmer einen Zugriff auf ein Arbeitsverzeichnis auf Ihrem Rechner zu erteilen. Der Nachteil dieser Methode könnte vielleicht aber auch ein Vorteil sein: Die Teilnehmer lernen mit einem FTP-Client umzugehen.

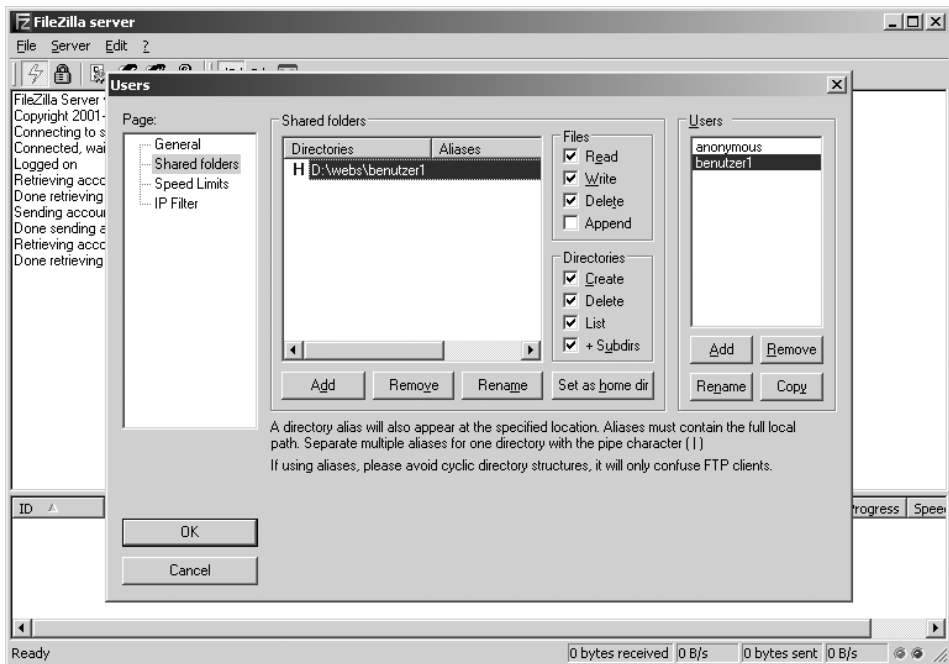


Abbildung 12.7: benutzer1 mit aktiviertem FTP-Heimatverzeichnis in FileZilla

Standardmäßig haben alle NT-/2000-/XP-Betriebssysteme einen FTP-Client als Konsolenprogramm. Weitere Informationen zur Einrichtung der im XAMPP enthaltenen FTP-Server erhalten Sie in Kapitel 5, »ProFTPD«, bzw. Kapitel 6, »FileZilla Server«.



Abbildung 12.8: Der FTP-Client im Einsatz unter XP Professional

Webverzeichnisse der Benutzer über den Apache HTTPD standardisieren

Sie haben XAMPP auf Ihrem Rechner installiert. Zudem haben Sie beispielsweise unter `d:\webs` die Benutzerverzeichnisse eingerichtet und den Zugriff per Freigabe oder per FTP wie bereits beschrieben eingerichtet. In diesem Beispiel ist der Name des jeweiligen Arbeitsverzeichnisses gleich dem Benutzernamen. In jedem der Benutzerverzeichnisse existiert ein `htdocs`-Ordner, der dem Teilnehmer als Webverzeichnis für seine Veröffentlichung im lokalen Netzwerk dient. Für *benutzer1* und *benutzer2* heißen die Webverzeichnisse somit:

```

D:\web\benutzer1\htdocs
D:\web\benutzer2\htdocs

```

Mit dem Apache-Modul `mod_userdir` können Sie nun dafür sorgen, dass diese `htdocs`-Verzeichnisse automatisch in Ihrem lokalen Web nach dem Muster `http://dozenten-rechner/~benutzer` aufrufbar sind.

Konfigurationsdatei: `<xampp>\apache\conf\httpd.conf`

```
LoadModule userdir_module modules/mod_userdir.so
```

Nun machen Sie sich an die Konfiguration der `httpd-userdir.conf`. Ein Beispiel könnte so aussehen:

Konfigurationsdatei: `<xampp>\apache\conf\extra\httpd-userdir.conf`

Listing 12.1: Konfiguration von `mod_userdir` in der Apache-Konfigurationsdatei

```

<IfModule userdir_module>
UserDir "d:/webs/*/htdocs"
<Directory "d:/webs">
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options MultiViews Indexes SymLinksIfOwnerMatch ExecCGI
    <Limit GET POST OPTIONS>
        Order allow,deny
        Allow from all
    </Limit>
</Directory>
</IfModule>

```



```

    </Limit>
</Directory>
</IfModule>

```

Das Zeichen * fungiert als Wildcard für den Benutzernamen. Zum Schluss kommentieren Sie die Konfigurationsdatei in der `httpd.conf` aus und starten Ihren Apache-Server neu.

Konfigurationsdatei: `<xampp>\apache\conf\extra\httpd-userdir.conf`

```
Include conf/extra/httpd-userdir.conf
```

Durch Ihre Konfiguration werden sowohl CGI als auch PHP ausgeführt. Haben Sie zudem das *XAMPP Perl-Add-On* unter Windows zusätzlich installiert, können Sie Perl-Skripts über `mod_perl` über die Dateieindung *.pl aktivieren. Hierfür setzen Sie im `<directory>`-Container Ihrer `httpd-userdir.conf` die folgenden Zeilen hinzu:

Listing 12.2: Aktivierung von Perl für die Benutzerverzeichnisse

```

<Files *.pl>
    SetHandler perl-script
    PerlResponseHandler ModPerl::Registry
    PerlOptions +ParseHeaders
</Files>

```

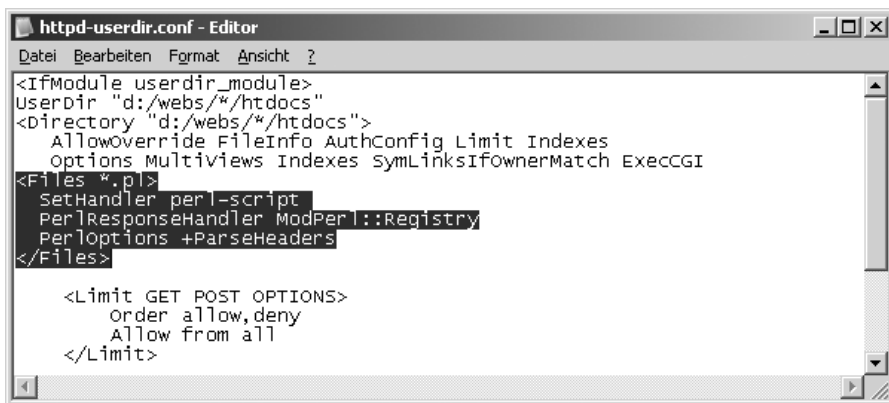


Abbildung 12.9: Die `httpd-userdir.conf` mit der Aktivierung des PerlHandlers

Vergessen Sie den Apache-Neustart nach dem Eingriff nicht! Danach testen Sie sicherheitshalber Ihre Modifikation über ein Verzeichnis eines beliebigen Benutzers bzw. zukünftigen Seminarteilnehmers. Entwerfen Sie dafür eine `test.php` (PHP), eine `test.cgi` (CGI) und ggf. eine `test.pl` (*mod_perl*) und prüfen Sie diese Scripts über die URL `http://localhost/~benutzer`. Wenn alle Programme zu Ihrer Zufriedenheit ausgeführt werden, können Sie sicher sein, dass Sie zukünftig mit Ihren Schülern effektiv arbeiten können.

Jedem Benutzer seine eigene MySQL-Datenbank

Wie eingangs bereits erwähnt, wollen wir noch schnell jedem Benutzer seine eigene Datenbank einrichten. Dafür nutzen Sie auf Ihrem Dozentenrechner das Konsolenprogramm *mysql*, also den sog. *MySQL-Monitor*. Melden Sie sich hier als *root* an. Das Passwort für *root* sollte für Ihre Arbeit in jedem Fall gesetzt sein!

```
<xampp>\mysql\bin\mysql -h localhost -u root -p
mysql>CREATE DATABASE benutzer1db;
mysql>GRANT ALL PRIVILEGES ON benutzer1db.localhost TO benutzer1 IDENTIFIED BY 'geheimt';
```

Hier wird der Zugriff über Ihren Rechner gestattet, was in der Regel für Webseiten und die Konfiguration über phpMyAdmin ausreicht. Soll ein vollständiger Zugriff über das Netzwerk auf die Datenbank für den Benutzer möglich sein, so brauchen Sie nur folgende Zeile hinzuzufügen:

```
mysql>GRANT ALL PRIVILEGES ON benutzer1db.% TO benutzer1 IDENTIFIED BY 'geheimt';
```

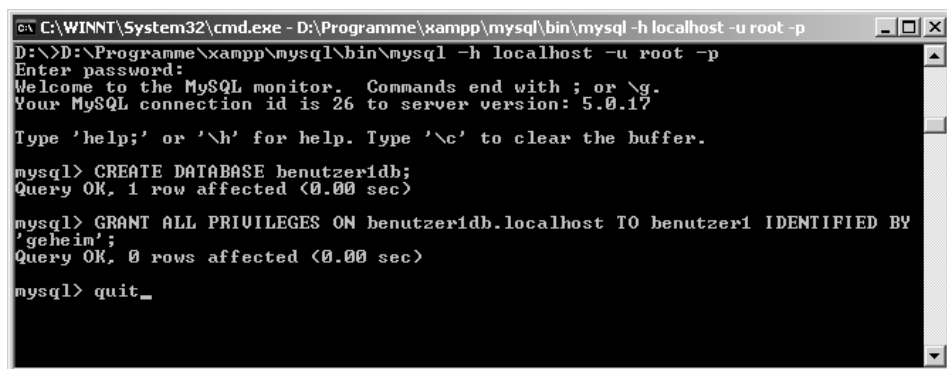


Abbildung 12.10: Im MySQL-Monitor eine Datenbank für Benutzer₁ einrichten

Nun darf selbstverständlich jeder der Benutzer seine Datenbank über die URL <http://dozentenrechner/phpmyadmin> durch eine zuvor erfolgte erfolgreiche Authentifikation administrieren. Eine Minimallösung in der *config.inc.php* von phpMyAdmin ist die Änderung der *auth_type*-Variablen wahlweise in eine Cookie-basierte oder HTTP-basierte Anmeldung. Der bisherige Wert *config* und *root* für den Standardbenutzer sollte damit entfernt werden.

Konfigurationsdatei: <xampp>\phpMyAdmin\config.inc.php

```
$cfg['Servers'][$i]['auth_type'] = 'cookie';
$cfg['Servers'][$i]['user'] = '';
```

Oder:

```
$cfg['Servers'][$i]['auth_type']    = 'http';
$cfg['Servers'][$i]['user']        = '';
```

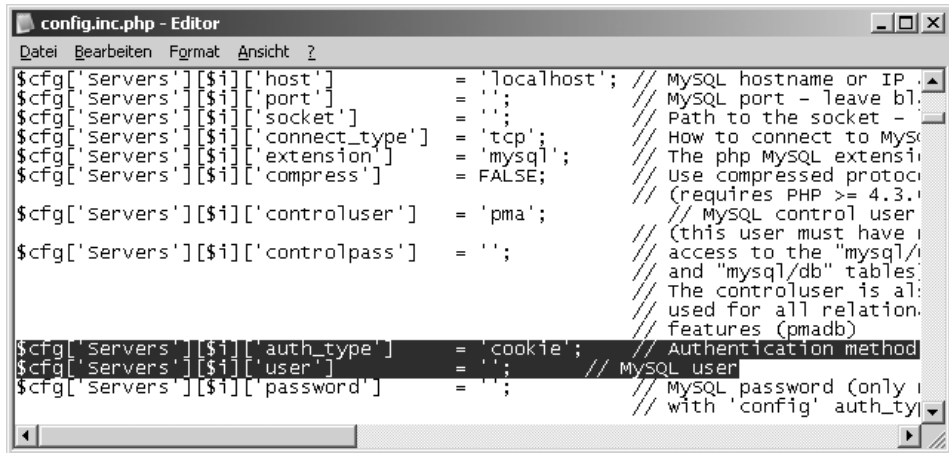
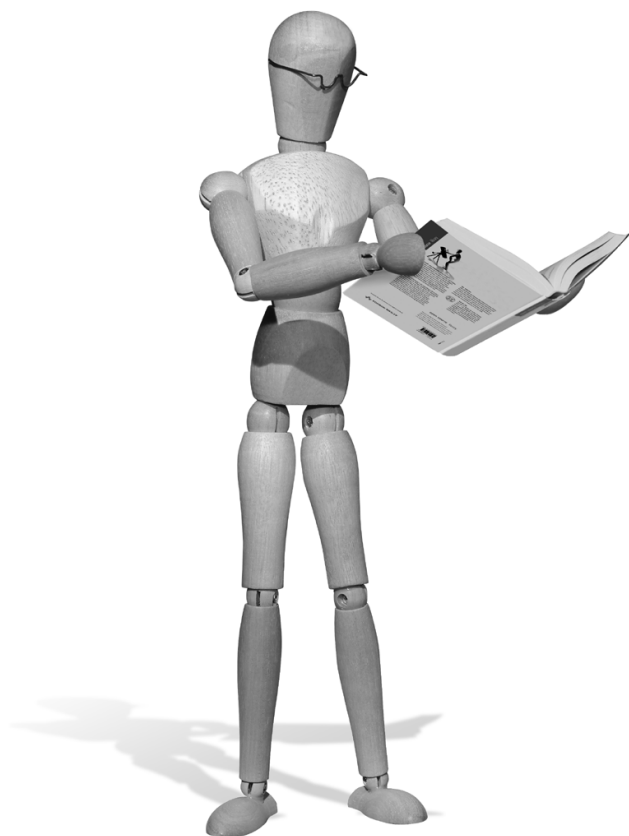


Abbildung 12.11: Die config.inc.php für eine Benutzerauthentifizierung anpassen

Teil III

Datenbanken





13 MySQL

Als das Unternehmen *TcX* im Jahr 1996 über die neu gegründete Firma *MySQL AB* die *MySQL Version 3.11.1* veröffentlichte, war den Machern wohl noch nicht bewusst, welche Popularität dieser Server schon bald erringen sollte. Das Open-Source-Projekt mit den sehr freizügigen Lizenzbestimmungen werde sich bei Webanwendungen durchsetzen, später dann in die unternehmenskritischen Bereiche vordringen, meinte dann auch der Analyst *Kevin Harvey* von *Benchmark Capital* im Jahr 2003. Mit MySQL haben Datenbankserver das Dasein als Nischenprodukt beendet. Damit konnte auch die Einschätzung des Oracle-Produktstrategen Ken Jacobs eindrucksvoll widerlegt werden, der sich damals selbstsicher zu diesem Thema äußerte: »Seit Jahren reden die Leute davon, dass Datenbanken zu einer Massenware werden, aber ich glaube das nicht« (Quelle: *Heise News*, 09.07.2003).



Hinweis

Besonderheit bei XAMPP für Windows

MySQL wird auch im *XAMPP für Windows* als Standard mitgeliefert. Einzig der Umfang des Programms unterscheidet sich. So ist MySQL vollständig im XAMPP-Hauptpaket enthalten. In der kleineren Lite-Version von XAMPP kommt hingegen ein reduziertes MySQL zum Tragen. So fehlen im *XAMPP Lite* einige Hilfsprogramme für MySQL. Trotzdem ist der Server auch in der Lite-Version voll funktionsstüchtig. Sie benutzen für dieses Kapitel natürlich den »großen« *XAMPP*.

13.1 Starten und Stoppen

13.1.1 Linux

Unter Linux wird der MySQL-Server automatisch beim Start von XAMPP mitgestartet und beim Stoppen von XAMPP ebenso automatisch heruntergefahren. Wollen Sie MySQL trotzdem unabhängig von XAMPP stoppen und starten, dann können Sie dies über die folgenden beiden Befehle erreichen:

Server starten: `/opt/lampp/lampp startmysql`

Server stoppen: `/opt/lampp/lampp stopmysql`

13.1.2 Windows

Unter Windows haben Sie drei Möglichkeiten, MySQL zu starten: von Hand, als Dienst und über das Control Panel.

Von Hand

Wenn Sie MySQL nur ab und zu brauchen, dann können Sie es über die XAMPP-Batch-Dateien starten und stoppen:

Start: `<xampp>\mysql_start.bat`

Stopp: `<xampp>\mysql_stop.bat`

Als Dienst

Sollten Sie MySQL eigentlich immer benötigen, dann empfiehlt sich die Installation von MySQL als Dienst. Dazu existieren die folgenden Batch-Dateien:

Installation: `<xampp>\mysql\mysql_installservice.bat`

Deinstallation: `<xampp>\mysql\mysql_uninstallservice.bat`

Über das Control Panel

Über das XAMPP Control Panel können Sie noch bequemer MySQL starten und stoppen oder es als Dienst installieren oder deinstallieren.

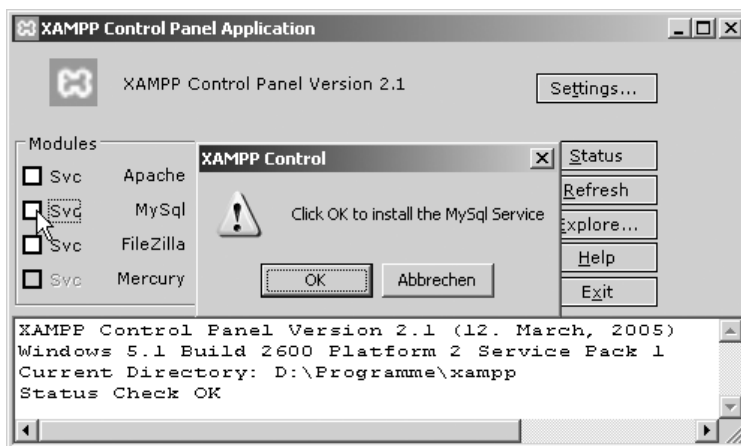


Abbildung 13.1: Dienste installieren Sie am einfachsten mit dem XAMPP Control Panel.

13.2 Administrative Werkzeuge auf Kommandoebene

MySQL enthält eine Reihe von Tools für die administrative Arbeit über die Kommandoebene. Sie finden diese Programme im XAMPP:

Linux: `/opt/lampp/bin`

Windows: `<xampp>\mysql\bin`

Wir beschreiben hier kurz die Funktion und die Möglichkeiten in der Anwendung der einzelnen Hilfsmittel. Konkrete Hilfe zu den jeweiligen Tools erhalten Sie in der Regel mit den Optionen `-?` oder `--help`.

Um sicherzugehen, dass Sie auch die MySQL-Befehle von XAMPP verwenden, sollten Sie entweder `/opt/lampp/bin` bzw. `C:\Programme\xampp\mysql\bin` in die PATH-Variablen des Systems aufnehmen oder diesen Pfad den Befehlen voranstellen:

Im Falle von Linux rufen Sie also nicht einfach `mysql` auf, sondern `/opt/lampp/bin/mysql`.

Und unter Windows lautet der Aufruf `C:\Programme\xampp\mysql\bin\mysql`.

Gerade bei Linux ist es wichtig, immer an die volle Pfadangabe zu denken. Dort kann es auch bereits eine MySQL-Installation durch die Linux-Distribution geben und wenn Sie den Pfad nicht mit angeben, verwenden Sie vielleicht eine ganz andere MySQL-Installation, als Sie eigentlich wollten.

13.2.1 mysql

Der sog. *MySQL Monitor* ist der Standardclient auf der Kommandozeile. Mit diesem Programm melden Sie sich auf dem Server ein und führen weitere SQL-relevante oder administrative Arbeiten aus. Das Login auf dem lokalen Rechner erfolgt über:

```
mysql -hlocalhost -uroot -p
```

Beendet wird das Programm mit:

```
quit
```

Im MySQL Monitor können Sie SQL-Befehle an den MySQL-Server senden. Ein Beispiel: die Änderung des Passworts des *root*-Administrators.

Listing 13.1: Protokoll eines MySQL Monitor-Aufrufs, um das Root-Passwort zu ändern

```
mysql -hlocalhost -uroot -p
Enter password:
mysql>use mysql;
mysql>UPDATE user SET Password=PASSWORD('meinpasswort') WHERE user='root';
mysql>FLUSH PRIVILEGES;
mysql>quit
```

13.2.2 mysqladmin

Dies ist das Administrationstool für den MySQL-Server. Hierüber lässt sich der Daemon »reloaden« und »refreshen«, stoppen oder einfach nur der Status abfragen. Weitere Hilfe erhalten Sie hier mit dem Parameter `--help`.

```
mysqladmin -h localhost -u root -p status
```

13.2.3 mysqlbinlog

Dieses Tool transferiert binäre Log-Dateien in lesbare Textformate. Im XAMPP ist das binäre *log-bin* seit der Version 1.4.11 deaktiviert. Ein Beispiel unter Linux:

```
/opt/lampp/bin/mysqlbinlog binlog.0000001
```

13.2.4 mysqlcheck

Mit diesem Zusatztool werden MySQL-Tabellen getestet, analysiert, repariert und optimiert. Ein Beispiel:

```
mysqlcheck. -a -h localhost -B mysql -u root -p
```

13.2.5 mysqldump

Mit diesem Dienstprogramm sichern Sie Datenbanken und übertragen Tabellen oder Datenbanken auf einen anderen SQL-Server. Folgende Anweisung erstellt eine Sicherung der MySQL-Systemdatenbank als `mysql.sql`:

```
mysqldump --opt mysql > mysql.sql -h localhost -u root -p
```

13.2.6 mysqlimport

Dieses Tool importiert SQL-Statements aus Textdateien. Es wird häufig benutzt, um neue Tabellen in eine Datenbank einzulesen. Wie gewohnt hier ein kleines Beispiel:

```
mysqlimport -L tabelle.txt -h localhost -u root -p
```

13.2.7 MySqlManager

Hierbei handelt es sich um ein grafisches Benutzer-Interface (GUI) für die Ausführung von SQL-Anweisungen für Microsoft Windows.

13.2.8 mysqlshow

Dieses Tool zeigt die strukturellen Merkmale der Datenbanken. Folgende Anweisung zeigt alle Tabellen der Datenbank `phpmyadmin`.

```
mysqlshow phpmyadmin -h localhost -u root -p
```

13.2.9 mysqltest

Mit diesem Tool führen Sie individualisierte Tests von SQL-Statements aus. Die SQL-Statements für die Ausführung der Tests werden über eine Textdatei eingelesen. In diesem Beispiel wird ein Statement für die Datenbank `test` simuliert.

```
mysqltest -D test -h localhost -u root -p < teststatement.sql
```

13.3 MySQL GUIs

Die auf den letzten Seiten vorgestellten Kommandozeilen-Tools sind fester Bestandteil von MySQL und deswegen auch im XAMPP mit dabei. Aber auch grafische Werkzeuge, die das Arbeiten mit MySQL erleichtern, bieten die MySQL-Entwickler an: MySQL Administrator, MySQL Migration Toolkit und MySQL Query Browser. Die Downloads finden Sie unter der URL <http://dev.mysql.com/downloads/>.

Die Installation des *MySQL Administrator* ist unter Windows denkbar einfach. Wer nicht den Installer benutzen möchte, nimmt einfach das ZIP. Entpacken Sie das ZIP in ein beliebiges Verzeichnis und starten Sie das Programm über die `MySQLAdministrator.exe`. Erfolgreich eingeloggt auf dem Server z. B. auf Ihrem lokalen Rechner (*localhost*), steht Ihnen eine Reihe von Möglichkeiten für eine Administration bereit. Insbesondere bei einer Dienstinstallation von MySQL sollten Sie zusätzlich den MySQL System Tray Monitor (`MySQLSystemTrayMonitor.exe`) für eine erweiterte Konfiguration benutzen. Dem MySQL Administrator ist es selbstverständlich egal, auf welcher Plattform der MySQL-Server läuft. Sie sollten allerdings Super-User Rechte (*root*) auf dem verbundenen MySQL-Server besitzen, um die administrative Ebene nutzen zu können. Für erweiterte Einstellungen können die XML-Dateien im gleichnamigen Ordner angepasst werden.

Sehr interessant ist das *MySQL Migration Toolkit Project*. Mit diesem Framework können Datenbanken aus Oracle, MS Access und auch weiteren SQL-Servern in MySQL eingebunden oder importiert werden.

Für die Praxis in SQL (Structured Query Language, strukturierte Abfragesprache) eignet sich neben *phpMyAdmin* auch der *MySQL Query Browser*. Der Zugriff von *root* ohne Passwort ist hier übrigens nicht erlaubt. Die Installation ist so simpel wie bei dem MySQL Administrator und sollte damit keine größeren Schwierigkeiten bereiten.

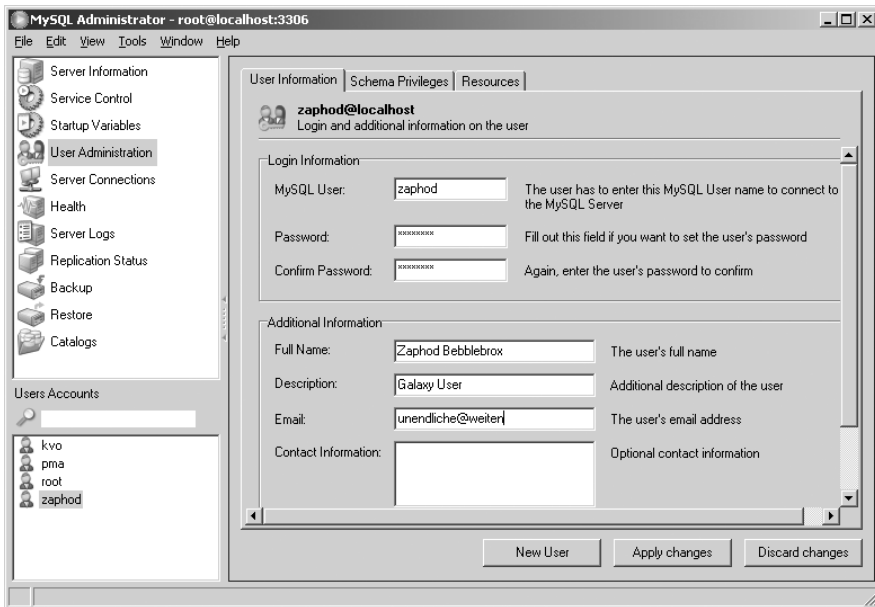


Abbildung 13.2: Inside MySQL Administrator

13.4 phpMyAdmin

phpMyAdmin ist sicherlich die bekannteste webbasierte Benutzeroberfläche für MySQL.



Abbildung 13.3: phpMyAdmin im XAMPP-Thema

Da XAMPP eigentlich zur Regel hat, nur Serverprogramme und Programmiersprachen zu beinhalten, ist phpMyAdmin die die Regel bestätigende Ausnahme. In der Windows-Version befindet sich das Tool unter `<xampp>\phpmyadmin`, unter Linux unter `/opt/lampp/phpmyadmin`. Dort finden Sie ebenfalls die zentrale Konfigurationsdatei von phpMyAdmin: `config.inc.php`. XAMPP enthält bereits eine offene Standardkonfiguration. Sie können also sofort das Programm unter `http://localhost/phpmyadmin` testen.

13.5 Beispielbenutzer und -datenbank einrichten

In den anderen Datenbank-Kapiteln in diesem Buch verwenden wir immer die CD-Verwaltung, um beispielhaft mit der jeweiligen Datenbank zu arbeiten. In XAMPP ist diese CD-Verwaltung aber schon als Beispiel für MySQL vorhanden. Aus diesem Grund verwenden wir hier einfach das SQLite-Telefonbuch-Beispiel und wenden es auf MySQL an.

1. Starten Sie den MySQL-Monitor, um mit ihm SQL-Befehle an die Datenbank senden zu können:

Unter Linux:

```
/opt/lampp/bin/mysql -uroot -p
```

Unter Windows:

```
C:\Programme\xampp\mysql\bin\mysql -uroot -p
```

2. Erstellen Sie im MySQL-Monitor eine neue Datenbank. Geben Sie dazu den folgenden Befehl ein:

```
CREATE DATABASE phonebook;
```

3. Wechseln Sie in diese neue Datenbank:

```
USE phonebook;
```

4. Im nächsten Schritt erstellen Sie eine neue Tabelle in dieser Datenbank. Geben Sie dazu den folgenden SQL-Befehl in den MySQL-Monitor ein:

```
CREATE TABLE users ( id INT NOT NULL PRIMARY KEY AUTO_INCREMENT, firstname ↵  
VARCHAR(200), lastname VARCHAR(200), phone VARCHAR(200) );
```

5. Als Nächstes füllen Sie diese Tabelle mit ein paar Beispieldaten:

```
INSERT INTO users (id, firstname, lastname, phone) ↵  
VALUES(1,'Zaphod','Beeblebrox','555-47110815');  
INSERT INTO users (id, firstname, lastname, phone) ↵  
VALUES(2,'Kai','Seidler','0123/987654');  
INSERT INTO users (id, firstname, lastname, phone) ↵  
VALUES(3,'Kay','Vogelgesang','0123/456789');
```

6. Im letzten Schritt richten Sie noch einen Benutzer *oswald* ein, der auf die Datenbank lokal zugreifen darf:

```
GRANT ALL PRIVILEGES ON phonebook.* TO 'oswald'@'localhost' IDENTIFIED BY 'geheim';
```

7. Nachdem Sie nun alles erledigt haben, können Sie den MySQL-Monitor beenden. Geben Sie dazu das folgende Kommando ein:

```
QUIT
```

Das war's! Eine neue Datenbank mit einer neuen Tabelle und ein neuer Benutzer sind eingerichtet. In den nächsten beiden Abschnitten sehen Sie nun, wie Sie von PHP und Perl aus auf diese Datenbank zugreifen können.

13.6 Der Datenbankzugriff mit PHP

Die MySQL-Unterstützung in PHP ist im XAMPP standardmäßig aktiviert und Sie können von PHP aus ohne weitere Schritte auf MySQL zugreifen.

Im folgenden Beispiel wird mit SELECT auf die Telefonbuch-Datenbank, die im letzten Abschnitt eingerichtet wurde, zugegriffen und das Ergebnis in einer ganz einfachen Tabelle ausgegeben.

Sie finden das folgende Beispielskript auch in Ihrer XAMPP-Installation unter `htdocs/xampp/contrib/mysql.php`, wenn Sie Linux verwenden, bzw. unter `htdocs\xampp\contrib\mysql.php`, wenn Sie mit Windows arbeiten.

Am Anfang des Skripts werden zunächst der HTML-Header, die Überschrift und die Kopfzeile der Tabelle ausgegeben:

```
<html>
<head>
<title>PHP und MySQL</title>
</head>
<body>
<h1>PHP und MySQL</h1>
<table border="1">
<tr>
  <th>Vorname</th>
  <th>Nachname</th>
  <th>Telefonnummer</th>
</tr>
```

Nun wird die Verbindung zur Datenbank aufgebaut. Dem `mysql_connect`-Befehl werden dabei drei Parameter übergeben:

1. der Rechner, auf dem MySQL läuft – es ist derselbe Rechner, auf dem auch Apache/PHP läuft: `localhost`

2. der Benutzer, unter dem zugegriffen werden soll: oswald
3. und sein Passwort: geheim

Danach wird mit `mysql_select_db()` die Datenbank `phonebook` ausgewählt.

```
<?php
mysql_connect("localhost", "oswald", "geheim");
mysql_select_db("phonebook");
```

Dieses Skript ist nur ein Beispiel. Im richtigen Leben sollten Sie hier auch noch die möglichen Fehlerfälle abfangen (Server nicht erreichbar, Benutzername/Passwort falsch, Datenbank gibt es nicht usw.) und eine entsprechende Fehlermeldung ausgeben.

Nun wird die eigentliche Anfrage an den Datenbankserver geschickt:

```
$query = "SELECT * FROM users";
$result = mysql_query($query);
```

Und danach wird das Ergebnis dieser Anfrage in einer `while()`-Schleife Zeile für Zeile ausgegeben:

```
while ($row = mysql_fetch_array ($result))
{
    echo "<tr>";
    echo "<td>".$row['firstname']. "</td>";
    echo "<td>".$row['lastname']. "</td>";
    echo "<td>".$row['phone']. "</td>";
    echo "</tr>";
}
```

Jetzt müssen nur noch das schließende PHP-Tag gesetzt und die noch offenen HTML-Tags geschlossen werden und das war's.

```
?>
</table>
</body>
</html>
```

Über <http://localhost/xampp/contrib/mysql.php> können Sie das Skript laufen lassen.

Hier zum besseren Überblick noch einmal das vollständige PHP-Skript:

Listing 13.2: Das komplette mysql.php

```
<html>
<head>
<title>PHP und MySQL</title>
</head>
<body>
```

```
<h1>PHP und MySQL</h1>
<table border="1">
<tr>
    <th>Vorname</th>
    <th>Nachname</th>
    <th>Telefonnummer</th>
</tr>
<?php
    mysql_connect("localhost", "oswald", "geheim");
    mysql_select_db("phonebook");

    $query = "SELECT * FROM users";
    $result = mysql_query($query);
    while ($row = mysql_fetch_array ($result))
    {
        echo "<tr>";
        echo "<td>".$row['firstname']. "</td>";
        echo "<td>".$row['lastname']. "</td>";
        echo "<td>".$row['phone']. "</td>";
        echo "</tr>";
    }

?>
</table>
</body>
</html>
```

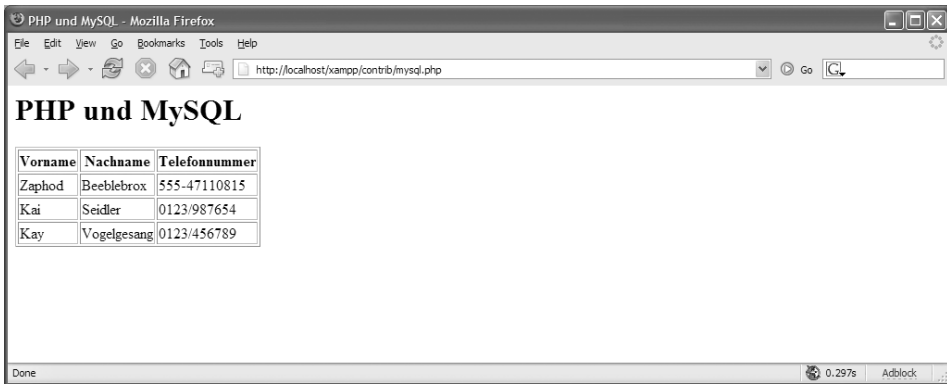


Abbildung 13.4: mysql.php in Aktion

13.7 Der Datenbankzugriff mit Perl

Bei *XAMPP für Windows* benötigen Sie zur Umsetzung dieses Beispiels das entsprechende *XAMPP Perl-Add-On*. In diesem Add-On steht Ihnen der DBD-Treiber *MySQL* zur Verfügung. In *XAMPP für Linux* sind Perl und der *MySQL-DBD-Treiber* standardmäßig enthalten.

Beim folgenden Beispiel wird der *DBD-MySQL-Treiber* verwendet. Sie finden dieses Beispiel unter dem Namen `mysql.pl` im Verzeichnis `htdocs\xampp\contrib` bzw. `htdocs/xampp/contrib`.

Am Anfang des Skripts finden Sie die folgende Zeile:

```
#!/opt/lampp/bin/perl
```

Diese Zeile wird unter Linux benötigt, damit zum Ausführen des Perl-Skripts auch das Perl von XAMPP verwendet wird. Unter Windows ist diese Zeile nicht nötig und wird ignoriert. Dort wird standardmäßig `mod_perl` benutzt, um Perl-Skripte auszuführen. In diesem Fall »weiß« der Apache bereits, wo Perl zu finden ist.

Danach wird das *DBI-Modul* für den Zugriff auf *MySQL* geladen und der *HTTP-Header Content-Type* ausgegeben.

```
use DBI;
```

```
print "Content-Type: text/html\n\n";
```

Im nächsten Schritt wird die Verbindung zur Datenbank aufgebaut. In der Variablen `$dsn` wird hier definiert, dass

- auf eine *MySQL-Datenbank* zugegriffen werden soll,
- die Datenbank `phonebook` heißt
- und auf dem Rechner `localhost` erreichbar ist.

Bei der `connect()`-Methode werden dann diese Variable, der Benutzername und das Passwort angegeben.

```
my $dsn="dbi:mysql:phonebook:localhost";
my $dbh=DBI->connect("$dsn","oswald","geheim") or
    die "Kann Datenbank nicht erreichen!";
```

War der Verbindungsaufbau nicht erfolgreich, wird eine Fehlermeldung ausgegeben und das Perl-Skript beendet.

War der Verbindungsaufbau erfolgreich, werden der *HTML-Header* und der *Tabelle*nkopf ausgegeben:

```
print "<html>";
print "<head>";
print "<title>Perl und MySQL</title>";
print "</head>";
print "<body>";
print "<h1>Perl und MySQL</h1>";
print "<table border=\"1\">";
print "<tr>";
print "<th>Vorname</th>";
print "<th>Nachname</th>";
print "<th>Telefonnummer</th>";
print "</tr>";
```

Nun wird die SQL-Anfrage an die Datenbank vorbereitet und an diese abgeschickt:

```
my $query="SELECT * FROM users";
```

```
my $prep_sql=$dbh->prepare($query) or die print "Can't prepare";
$prep_sql->execute() or die print "Can't execute";
```

War die Anfrage erfolgreich, wird das Ergebnis der Anfrage Zeile für Zeile in der HTML-Tabelle ausgegeben:

```
while (my @row = $prep_sql->fetchrow_array())
{
    print "<tr>";
    print "<td>".$row[1]."</td>";
    print "<td>".$row[2]."</td>";
    print "<td>".$row[3]."</td>";
    print "</tr>";
}
```

Danach wird die Verbindung zur Datenbank getrennt:

```
$prep_sql->finish();
$dbh->disconnect();
```

Und die Tabelle und sonstigen HTML-Elemente werden geschlossen.

```
print "</table>";
print "</body>";
print "</html>";
```

Das war's.

Über <http://localhost/xampp/contrib/mysql.pl> können Sie nun das Skript laufen lassen.

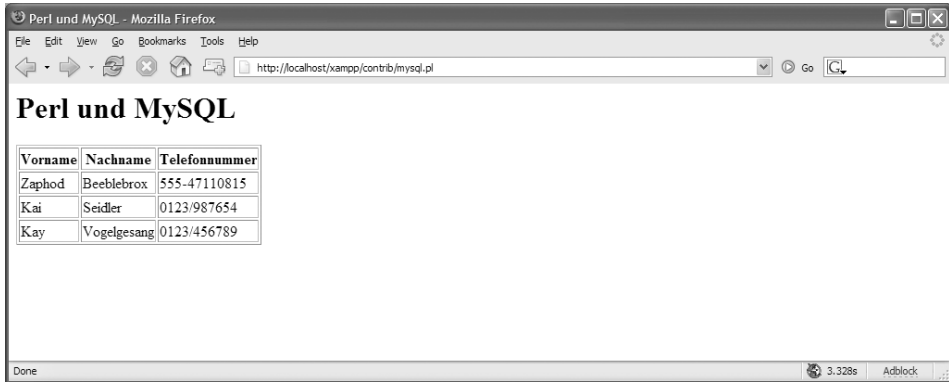


Abbildung 13.5: Einfachster MySQL-Zugriff mit Perl: *mysql.pl*

Hier noch einmal das komplette *mysql.pl*-Skript im Überblick:

*Listing 13.3: Das komplette *mysql.pl**

```
#!/opt/lampp/bin/perl
use DBI;

print "Content-Type: text/html\n\n";

my $dsn="dbi:mysql:phonebook:localhost";
my $dbh=DBI->connect("$dsn","oswald","geheim") or
    die "Kann Datenbank nicht erreichen!";

print "<html>";
print "<head>";
print "<title>Perl und MySQL</title>";
print "</head>";
print "<body>";
print "<h1>Perl und MySQL</h1>";
print "<table border=\"1\">";
print "<tr>";
print "<th>Vorname</th>";
print "<th>Nachname</th>";
print "<th>Telefonnummer</th>";
print "</tr>";

my $query="SELECT * FROM users";

my $prep_sql=$dbh->prepare($query) or die print "Can't prepare";
$prep_sql->execute() or die print "Can't execute";

while (my @row = $prep_sql->fetchrow_array())
{
```

```
        print "<tr>";
        print "<td>".$row[1]."</td>";
        print "<td>".$row[2]."</td>";
        print "<td>".$row[3]."</td>";
        print "</tr>";
    }

    $prep_sql->finish();
    $dbh->disconnect();

    print "</table>";
    print "</body>";
    print "</html>";
```



14 SQLite

In Bezug auf *SQLite* scheiden sich die Geister. Wie sagt man so schön, wenn die einen etwas innig lieben und die anderen es geradezu hassen? Etwas polarisiert – *SQLite* polarisiert.

SQLite ist eine SQL-Datenbank (bzw. richtig ausgedrückt: ein SQL-Datenbank-managementsystem), die komplett ohne Server und ohne Typen auskommt.

D. h. zum einen, dass man sich um keinen Server kümmern muss, und zum anderen, dass man sich nicht einmal Gedanken darüber machen muss, was man in den einzelnen Tabellen der Datenbank speichern möchte. Programmierern und Informatikern, die der reinen Lehre folgen, wird hier schon extrem flau im Bauch.

Aber grundsätzlich muss man sagen, dass *SQLite* für kleine und mittlere Projekte wunderbar geeignet ist. Nur bei besonders großen Datenmengen und Zugriffszahlen wird *SQLite* an seine Grenzen stoßen. Dann sind serverbasierte Datenbanken sicherlich im Vorteil.

Die Frage, was Groß und was Klein ist, ist natürlich nicht so einfach zu beantworten. Aber sagen wir es so: Die private CD-Verwaltung wird man auf alle Fälle wunderbar mit *SQLite* realisieren können. Auch wenn man Millionen von CDs hat. Und auch wenn die ganze Familie und Verwandtschaft gleichzeitig auf die Datenbank zugreifen möchte.

14.1 Installation

SQLite ist komplett in allen Versionen von XAMPP enthalten und es muss nichts weiter installiert werden.

14.2 Beispieldatenbank einrichten

In diesem Beispiel verwenden wir – wie auch in den anderen Datenbank-Kapiteln – die CD-Verwaltung als Beispielanwendung. Um eine Datenbank anzulegen, benutzen Sie das *SQLite*-Konsolenprogramm über die Shell bzw. die Windows-Eingabeaufforderung.

```

C:\WINDOWS\system32\cmd.exe

C:\>cd c:\Programme\XAMPP\htdocs\xampp\sqlite

C:\Programme\xampp\htdocs\xampp\sqlite>..\..\..\apache\bin\sqlite cdcol
SQLite version 2.8.15
Enter ".help" for instructions
sqlite> CREATE TABLE cds (id INT NOT NULL, titel VARCHAR(200), interpret VARCHAR
(200), jahr INT, PRIMARY KEY (id));
sqlite> INSERT INTO cds VALUES (1,'Beauty','Ryuichi Sakamoto',1990);
sqlite> INSERT INTO cds VALUES (2,'Goodbye Country (Hello Nightclub)','Groove Ar
mada',2001);
sqlite> INSERT INTO cds VALUES (3,'Glee','Bran Van 3000',1997);
sqlite> .quit

C:\Programme\xampp\htdocs\xampp\sqlite>_

```

Abbildung 14.1: Anlegen einer SQLite-Tabelle über die Eingabeaufforderung unter Windows

1. Rufen Sie eine Shell oder die Windows-Eingabeaufforderung auf.
2. Wechseln Sie in das Verzeichnis `/opt/lampp/htdocs/xampp/sqlite`, wenn Sie Linux verwenden, bzw. `htdocs\xampp\sqlite`, wenn Sie mit Windows arbeiten.
3. Erstellen Sie hier mit dem `sqlite`-Befehl eine neue Datenbank mit dem Namen `cdcol`. Vereinfacht kann man sagen, dass in SQLite eine Datenbank immer einer Datei entspricht.

Wenn Sie also mit dem folgenden Befehl eine »Datenbank« anlegen, dann legen Sie eigentlich nur eine SQLite-Datei mit dem Namen `cdcol` an.

Unter Linux:

```
/opt/lampp/bin/sqlite cdcol
```

Unter Windows:

```
<xampp>/apache/bin/sqlite cdcol
```

Nach Aufruf des Befehls erscheint der `SQL>`-Prompt und Sie können nun SQL-Befehle an diese neue Datenbank senden.

4. Legen Sie zunächst eine neue Tabelle mit `CREATE TABLE` an. Als Beispiel dient wieder einmal die CD-Verwaltung:

```
CREATE TABLE cds (id INT NOT NULL, titel VARCHAR(200), interpret VARCHAR(200),
jahr INT, PRIMARY KEY (id));
```

5. Füllen Sie nun die cds-Tabelle mit den üblichen Beispieldaten:

```
INSERT INTO cds VALUES (1,'Beauty','Ryuichi Sakamoto',1990);
INSERT INTO cds VALUES (2,'Goodbye Country (Hello Nightclub)','Groove
Armada',2001);
INSERT INTO cds VALUES (3,'Glee','Bran Van 3000',1997);
```

6. Wenn Sie alle INSERT-Befehle eingegeben haben, beenden Sie das sqlite-Kommando und geben nun .quit ein.

Bei der Anweisung .quit zum Beenden des SQL-Monitors handelt es sich um ein SQLite-spezifisches Kommando. Alle Anweisungen müssen mit einem Punkt beginnen.

Eine Übersicht über alle Anweisungen erhalten Sie mit dem Kommando .help:

Listing 14.1: Sämtliche Kommandos des SQLite-Monitors

.databases	List names and files of attached databases
.dump ?TABLE? ...	Dump the database in a text format
.echo ON OFF	Turn command echo on or off
.exit	Exit this program
.explain ON OFF	Turn output mode suitable for EXPLAIN on or off.
.header(s) ON OFF	Turn display of headers on or off
.help	Show this message
.indices TABLE	Show names of all indices on TABLE
.mode MODE	Set mode to one of "line(s)", "column(s)", "insert", "list", or "html"
.mode insert TABLE	Generate SQL insert statements for TABLE
.nullvalue STRING	Print STRING instead of nothing for NULL data
.output FILENAME	Send output to FILENAME
.output stdout	Send output to the screen
.prompt MAIN CONTINUE	Replace the standard prompts
.quit	Exit this program
.read FILENAME	Execute SQL in FILENAME
.schema ?TABLE?	Show the CREATE statements
.separator STRING	Change separator string for "list" mode
.show	Show the current values for various settings
.tables ?PATTERN?	List names of tables matching a pattern
.timeout MS	Try opening locked tables for MS milliseconds
.width NUM NUM ...	Set column widths for "column" mode

Hinweis



SQLite und die Zugriffsrechte

Da SQLite direkt in die Datenbank-Datei schreibt, muss diese selbstverständlich auch für den Apache-Prozess, in dem später ja PHP oder Perl auf die SQLite-Datenbank zugreift, schreib- und lesbar sein. Nicht so selbstverständlich, aber trotzdem notwendig: Auch das Verzeichnis, in dem sich die Datenbank-Datei befindet, muss für den Apache-Prozess lesbar und schreibbar sein.

14.3 Datenbankzugriff mit PHP

Die SQLite-Unterstützung von PHP ist standardmäßig aktiviert. Sie können also gleich mit der Programmierung beginnen.

Das Beispielskript befindet sich als `sqlite.php` im Verzeichnis `htdocs/xampp/contrib` bzw. `htdocs\xampp\contrib` in Ihrer XAMPP-Installation.

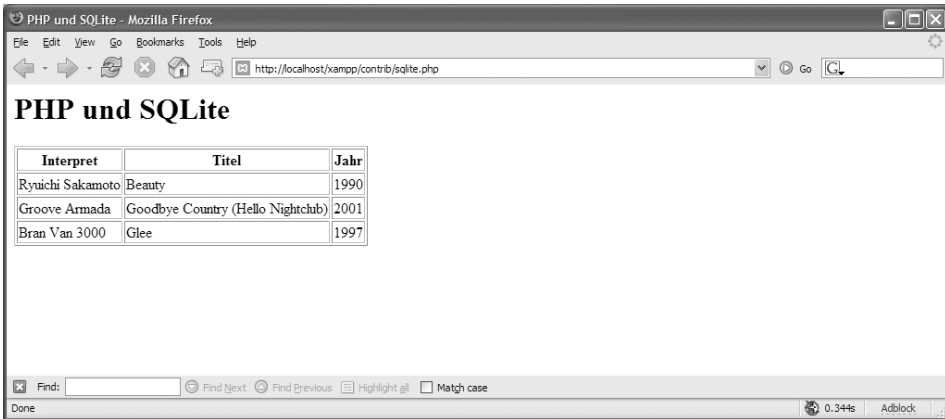


Abbildung 14.2: PHP und SQLite

Ganz am Anfang werden der HTML-Header und die Überschriften für die Tabelle ausgegeben:

```
<html>
<head>
<title>PHP und SQLite</title>
</head>
<body>
<h1>PHP und SQLite</h1>
<table border="1">
<tr>
  <th>Interpret</th>
  <th>Titel</th>
  <th>Jahr</th>
</tr>
<?php
```

Danach wird die Verbindung zur Datenbank(-Datei) hergestellt:

```
$db = sqlite_open('../sqlite/cdcol');
```


Im nächsten Schritt wird die SQL-Anfrage abgeschickt:

```
$query = "SELECT * FROM cds";
$result = sqlite_query($db,$query);
```

Ist die Anfrage erfolgreich, wird in einer while-Schleife das Ergebnis Zeile für Zeile in einer HTML-Tabelle ausgegeben.

```
if ($result)
{
    while ($row = sqlite_fetch_array($result))
    {
        echo "<tr>";
        echo "<td>".$row['interpret']."</td>";
        echo "<td>".$row['titel']."</td>";
        echo "<td>".$row['jahr']."</td>";
        echo "</tr>";
    }
}
```

Gab es bei der SQL-Anfrage einen Fehler, dann wird dieser hier ausgegeben:

```
else
{
    echo sqlite_error_string(sqlite_last_error($db));
}
?>
```

Schließlich werden noch Tabellen-, Body- und HTML-Tag geschlossen:

```
</table>
</body>
</html>
```

Dieses Skript hat zwei kleine Unschönheiten an sich:

1. Es wird kein Fehlerfall bei `sqlite_open()` abgefangen.
2. Bei `sqlite_query()` wird dann zwar der Fehlerfall abgefangen, aber die Fehlermeldung würde wüst im HTML-Code der Tabelle landen.

Diese beiden Schwächen haben alle PHP-Skripte in diesem Kapitel. Würde man sie »schöner« machen wollen, dann wären sie auch schnell viel länger und auch nicht mehr so übersichtlich in diesem Buch unterzubringen. Primär geht es hier um den Zugriff auf Datenbanken und nicht um sauberen Programmierstil.

Zum Schluss noch einmal das gesamte Listing:

Listing 14.2: Der Zugriff auf SQLite mit `sqlite.php`

```
<html>
<head>
<title>PHP und SQLite</title>
</head>
<body>
<h1>PHP und SQLite</h1>
<table border="1">
<tr>
    <th>Interpret</th>
    <th>Titel</th>
    <th>Jahr</th>
</tr>
<?php
    $db = sqlite_open('../sqlite/cdcol');

    $query = "SELECT * FROM cds";
    $result = sqlite_query($db,$query);
    if ($result)
    {
        while ($row = sqlite_fetch_array($result))
        {
            echo "<tr>";
            echo "<td>".$row['interpret']."</td>";
            echo "<td>".$row['titel']."</td>";
            echo "<td>".$row['jahr']."</td>";
            echo "</tr>";
        }
    }
    else
    {
        echo sqlite_error_string(sqlite_last_error($db));
    }

?>
</table>
</body>
</html>
```

14.4 Datenbankzugriff mit Perl

Bei XAMPP für Windows benötigen Sie zur Umsetzung dieses Beispiels das entsprechende *XAMPP Perl-Add-On*. In diesem Add-On stehen wahlweise die DBD-Treiber *SQLite* und *SQLite2* zur Verfügung. Im XAMPP für Linux sind Perl und die beiden *SQLite*-DBD-Treiber standardmäßig enthalten.

Die Benennung der Treiber ist ein wenig missverständlich und erfordert ein wenig Hintergrundwissen: *SQLite* liegt zurzeit in zwei Hauptversionen vor: 2 und 3. Version 3 ist zwar die aktuellste, aber Version 2 die zurzeit am weitesten verbreitete Version. Aus diesem Grund ist in XAMPP hauptsächlich Version 2 im Einsatz. Der DBD-Treiber *SQLite* benutzt allerdings schon Version 3 und ist deshalb mit der Datenbank, die Sie vorhin mit dem `sqlite`-Befehl erstellt haben, nicht kompatibel.

Aus diesem Grund wird bei dem folgenden Beispiel der *SQLite2*-Treiber verwendet. Sie finden dieses Beispiel unter dem Namen `sqlite.pl` im Verzeichnis `htdocs\xampp\contrib` bzw. `htdocs\xampp/contrib`.

*Listing 14.3: Das Skript `sqlite.pl` für die XAMPP-Linux-Version. In der Windows-Version muss die Pfadangabe zur *SQLite*-Datenbank absolut sein, also z.B. `/programme/xampp/htdocs/xampp/sqlite/cdcol`.*

```
#!/opt/lampp/bin/perl
use DBI;

print "Content-Type: text/html\n\n";

my $dsn="dbi:SQLite2:dbname=../sqlite/cdcol";
my $dbh=DBI->connect($dsn,"","") or
    die "Kann Datenbank nicht erreichen!";

print "<html>";
print "<head>";
print "<title>Perl und SQLite</title>";
print "</head>";
print "<body>";
print "<h1>Perl und SQLite</h1>";
print "<table border=\"1\">";
print "<tr>";
print "<th>Interpret</th>";
print "<th>Titel</th>";
print "<th>Jahr</th>";
print "</tr>";

my $query="SELECT * FROM cds";

my $prep_sql=$dbh->prepare($query) or die print "Can't prepare";
$prep_sql->execute() or die print "Can't execute";
```

```
while (my @row = $prep_sql->fetchrow_array())
{
    print "<tr>";
    print "<td>".$row[2]."</td>";
    print "<td>".$row[1]."</td>";
    print "<td>".$row[3]."</td>";
    print "</tr>";
}

$prep_sql->finish();
$dbh->disconnect();

print "</table>";
print "</body>";
print "</html>";
```

14.5 Tools

Zahlreiche Verweise auf freie oder auch kommerzielle Tools bieten die Entwickler von SQLite auf ihrer gut sortierten Website an:

<http://www.sqlite.org/cvstrac/wiki?p=SqliteTools>

14.5.1 SQLiteCC

Eines dieser Zusatzprogramme heißt *SQLiteCC*, ein einfaches und effektives grafisches Frontend für Linux, Windows und Mac OS X. Herunterladen können Sie es unter <http://bobmanc.home.comcast.net/sqlitecc.html>.

14.5.2 phpSQLiteAdmin

phpSQLiteAdmin ist ein Webinterface für die Administration von SQLite-Datenbanken ähnlich dem *phpMyAdmin*-Interface für MySQL. Allerdings sollten wir jetzt besser im Plural reden, denn: Es gibt mindestens zwei *phpSQLiteAdmin*. Und in beiden Fällen erschien das letzte Release im Jahr 2004. Das eine Paket wird unter <http://www.phpguru.org/static/phpSQLiteAdmin.html> angeboten und funktioniert nur mit PHP4, und damit leider auch nicht ohne Probleme. Benutzer und Passwort bei »PHPGuru« lauten übrigens *admin* und *password*.

Kommen wir also gleich zu dem zweiten Paket mit dem Namen *phpSQLiteAdmin*, das einen etwas solideren Eindruck hinterließ: Sie erhalten es unter der URL <http://phpsqliteadmin.sourceforge.net>. In der Linux-Version von XAMPP ist dieses Paket auch schon standardmäßig enthalten und Sie finden es auf den XAMPP-Demoseiten unter dem TOOLS-Menüpunkt.

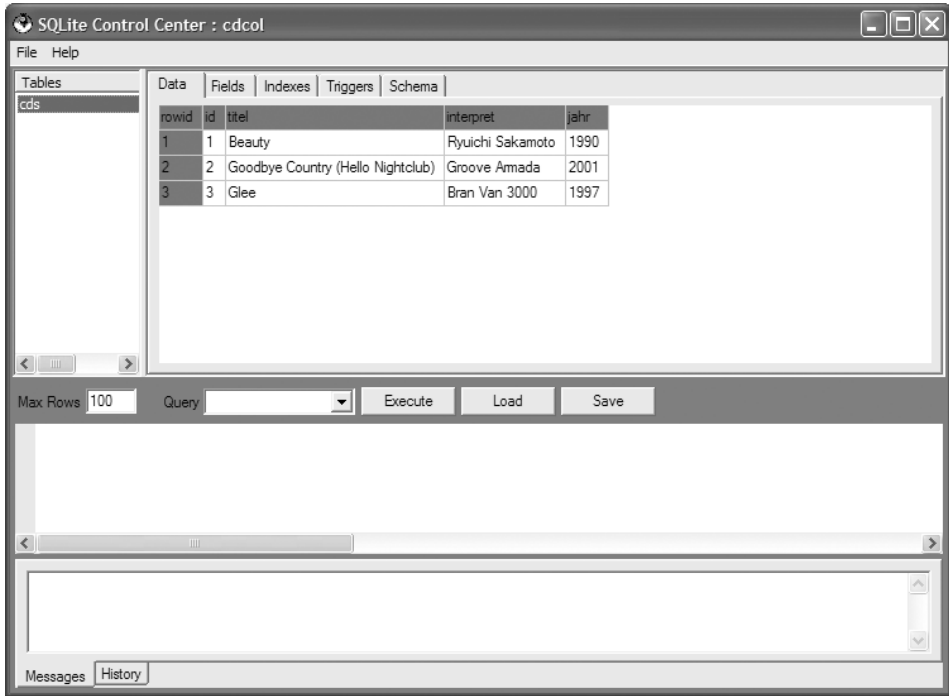


Abbildung 14.3: SQLiteCC: das SQLite Control Center

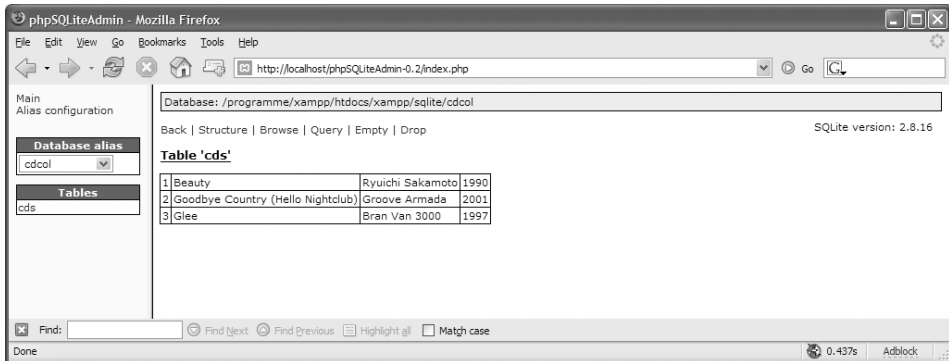


Abbildung 14.4: Das zweite von zwei phpSQLiteAdmins

Nach dem Auspacken der phpSQLiteAdmin-0.2.zip im Verzeichnis `<xampp>\htdocs` können Sie gleich loslegen. In puncto Sicherheit enthielt das Tool in der Konfiguration leider keine Einstellungsmöglichkeiten. Dafür sollten Sie also selbst sorgen, beispielsweise über eine `.htaccess`.

Nicht auf den ersten Blick verständlich: Über die ALIAS CONFIGURATION-Funktion bekommen Sie Zugriff auf Ihre SQLite-Datenbank-Datei.



Achtung

Obacht Windows-Benutzer: Bei den Pfadangaben muss immer / und nicht \ verwendet werden.



15 Microsoft SQL Server

Orientiert am Modell des *Sybase SQL Server* entwickelte Microsoft Ende der achtziger Jahre den *Microsoft SQL Server*, abgekürzt *MS SQL*. Mit der Version 4.2 im Jahr 1993 gelang dem Unternehmen erstmals ein ernstzunehmender Vorstoß gegen die Konkurrenz. Mittlerweile hat sich der MS SQL Server als relationales Datenbanksystem vorwiegend im Bereich der Mittelstandslösungen etabliert.

In diesem Kapitel erhalten Sie wichtige Hinweise bei der Installation der freien Express Edition des MS SQL Servers sowie der Konfiguration im XAMPP, um Zugriffe mit PHP oder Perl auf den Plattformen Windows und Linux realisieren zu können.

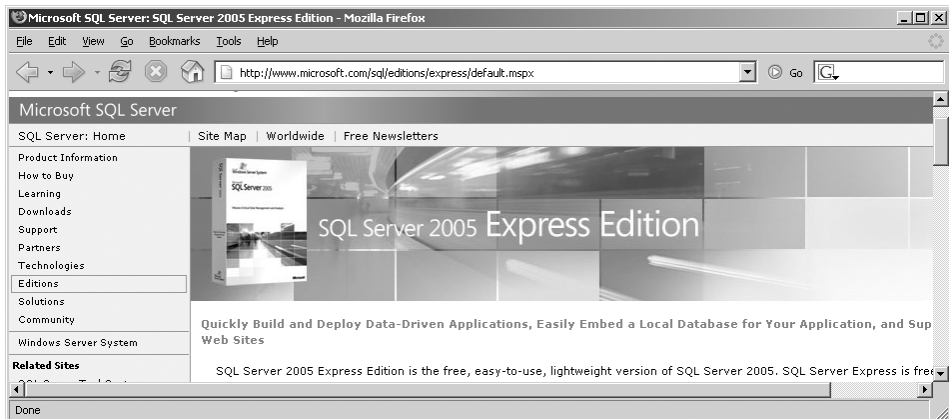


Abbildung 15.1: Die kostenfreie SQL Server 2005 Express Edition bei Microsoft

15.1 Die Installation

Der Microsoft SQL Server ist in zwei Varianten erhältlich. Neben der kostenpflichtigen Version des SQL Servers gibt es mit der *Microsoft SQL Server 2005 Express Edition* (früher *Microsoft SQL Desktop Engine*, kurz auch nur *MSDE* genannt) eine kleinere und kostenfreie Ausführung. Die Beschränkungen der freien Variante liegen vor allem in der Limitierung der Anzahl der gleichzeitigen Zugriffe. In der Express-Edi-

tion dürfen maximal fünf gleichzeitige Zugriffe erfolgen. Zudem fehlen so manche grafische Konfigurations- und Administrationsprogramme wie der *Enterprise Manager*. Da sich der SQL Server Express 2005 nicht im Zugriff und der zugrunde liegenden Technologie von seinem großen Bruder, dem Microsoft SQL Server unterscheidet, ist er ein geeigneter Einstieg in die Microsoft-SQL-Praxis.

Hinweis



Doch bevor Sie zur Tat schreiten, kurz ein sehr wichtiger Hinweis. Sollten Sie mit PHP arbeiten wollen, könnte es mit dem SQL Server 2005 Express Probleme geben. Denn mit der zum Zeitpunkt der Erstellung dieses Buches aktuellen PHP-Version 5.1.2 war es nicht möglich, auf die Serverinstanz zuzugreifen. Dieses Hindernis könnte aber bei den neueren PHP-Versionen bereits beseitigt sein. Alternativ benutzen Sie die ältere, aber ebenso freie *Microsoft SQL Server 2000 Desktop Engine* (MSDE) als SQL-Serverinstanz. Perl-Programmierer unter Windows haben in diesem Fall übrigens weniger Probleme. Sie greifen in unserem Beispiel über ein erstelltes System-DNS (ODBC) auf Perl-Programme zu.

Die kostenlose Microsoft SQL Server 2000 Desktop Engine, der SQL Server 2005 Express und Trial-Versionen des SQL Servers sind über das Downloadcenter von Microsoft erhältlich: <http://www.microsoft.com/downloads>. Ein weiteres nützliches und freies Programm analog zum *SQL Server Enterprise Manager* für die 2000-Edition von Microsoft heißt *Dbamgr2k* und kann unter <http://www.asql.biz/Dbamgr.shtml> bezogen werden. Die 2000-Edition entspricht der Versionsnummer 8, die 2005-Edition der Nummer 9.0.

15.1.1 MS SQL Server 2005 Express

Die Installation von SQL Express hat so ihre Tücken. Beginnen wir mit den Systemanforderungen:

- Windows XP mit Service Pack 2 oder höher
- Microsoft Windows 2000 Server mit Service Pack 4
- Windows Server 2003 Standard, Enterprise, oder Datacenter Editions mit Service Pack 1
- Intel Pentium III oder ein 600-Megahertz(MHz)-Prozessor oder höher, 1 Gigahertz oder höher wird empfohlen
- 256 Megabyte (Mbyte) Arbeitsspeicher (RAM), 512 Mbyte oder mehr werden empfohlen
- Microsoft .NET Framework 2.0
- Microsoft Installer 3.0

Sie wissen nicht, ob Sie über Microsoft .NET Framework 2.0 verfügen? In der Auflistung unter SYSTEMSTEUERUNG > SOFTWARE sollte das auf Ihrem System verfügbare Framework sichtbar sein. Das Microsoft .NET Framework 2.0 erhalten Sie übrigens wieder im Download-Bereich von Microsoft:

<http://msdn.microsoft.com/vstudio/express/sql/default.aspx>

SQL Server 2005 Express:

SQLEXPR_GER.EXE

SQL Server Management Studio Express:

SQLServer2005_SSMSEE.msi

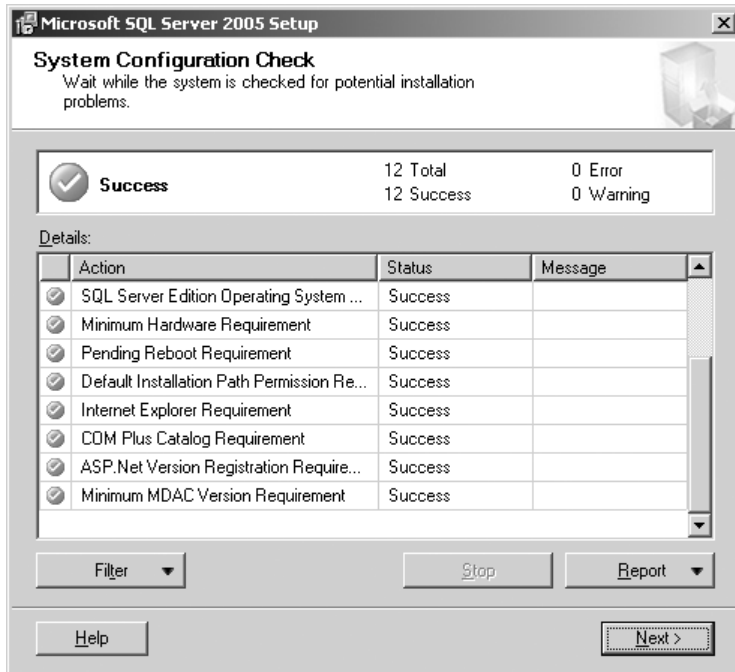


Abbildung 15.2: Das Setup vom SQL Express Server prüft zu Beginn die Komponenten.



Achtung

Sollten Sie Microsoft .NET Framework 2.0 nachrüsten müssen, bitte deinstallieren Sie zuerst das alte Framework und installieren danach das Framework 2.0! Das neue Framework benutzt zudem den Microsoft Installer in der Version 3.

Sie müssen für die weitere Installation Administratorrechte besitzen. Wenn Sie die oben angeführten Vorbereitungen umgesetzt haben, starten Sie die `SQLEXPR_GER.EXE` für das Setup.



Abbildung 15.3: Die Authentifizierung des SQL Servers unbedingt auf Gemischter Modus setzen!

Bei dem Setup müssen Sie unbedingt beachten, dass Sie die Authentifizierungsmethode auf GEMISCHTER MODUS setzen, wenn Sie die SQL Express-Instanz von weiteren Computern in Ihrem Netzwerk bzw. Nicht-Microsoft-Programmen wie Komponenten im XAMPP erreichen wollen.

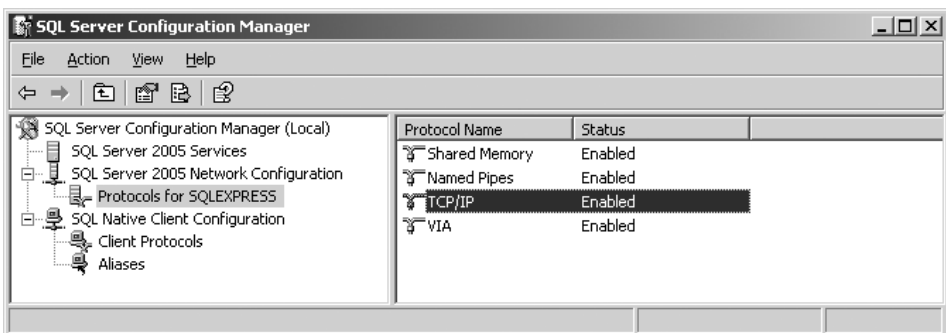


Abbildung 15.4: Alle Serverprotokolle sind in diesem Fall aktiviert

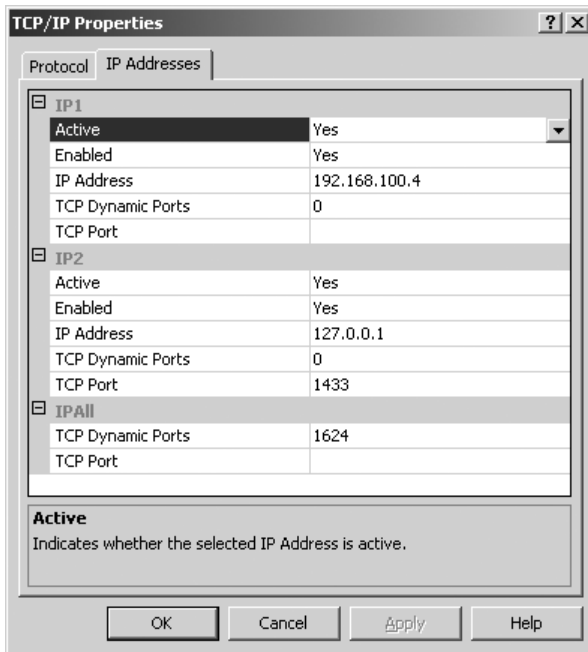


Abbildung 15.5: Vorteilhaft: Die IP-Adressen können einzeln behandelt werden

Haben Sie die Installation des SQL Express Servers erfolgreich hinter sich gebracht? Leider noch nicht ganz! Nachdem Sie den Server als Dienst gestartet haben, benutzen Sie den *SQL Server Configuration Manager*, um die notwendigen TCP/IP-Einstellungen vorzunehmen. Sie finden den Manager unter **START > PROGRAMME > MICROSOFT SQL SERVER 2005 > KONFIGURATIONSTOOLS**. Dort aktivieren Sie alle TCP/IP-Protokolle unter *Protocols for SQLEXPRESS* und *Client Protocols*. Bei Zugriffsproblemen mit anderen Programmen können Sie auch weitere Protokolle aktivieren. Danach testen Sie beispielsweise mit einem Portscanner, ob der SQL-Serverport 1433 erreichbar ist.

Zum guten Schluss fehlt noch die Installation von *SQL Server Management Studio Express*, des Administrationstools für den SQL-Administrator. Hierüber können Sie Datenbanken erstellen, Benutzer einrichten, Rechte vergeben, Sicherungen anlegen, SQL ausführen und vieles mehr.

Die Installation des *SQL Server Management Studio Express* sollte Ihnen keine größeren Probleme bereiten. Tipp: Loggen Sie sich ein und schauen Sie mal nach, was Sie mit diesem Programm alles machen können.



Abbildung 15.6: Die Anmeldung auf dem Rechner KVO über den SQL Server Manager

15.1.2 MS SQL Server 2000 Desktop Engine

Die Verwendung der freien *Microsoft SQL Server 2000 Desktop Engine* ist eine Alternative zur Installation des *MS SQL Server 2005 Express*. Das wird für Sie dann interessant, wenn Sie eine dem MS SQL Server 2000 ähnliche Umgebung schaffen und hierauf vielleicht auch mit unterschiedlichen PHP-Versionen zugreifen möchten. Das Paket erhalten Sie erneut über den Download-Bereich von Microsoft:

URL: <http://www.microsoft.com/downloads/>

Microsoft SQL Server 2000 Desktop Engine: GER_MSDE2000A.exe

Im Vorfeld der Installation des Release A (*MSDE2000A*) sollten Sie noch vor dem Setup unbedingt die `setup.ini` im bereits extrahierten `MSDERelA`-Ordner anpassen. Dringend erforderlich ist die Setzung eines Passworts für den Systemadministrator (*sa*) über den Wert `SAPWD` sowie die Festlegung des Namens dieser Dienstinstanz (`INSTANCENAME`). Darüber hinaus ist die Einstellungen `SECURITYMODE=SQL` für eine Benutzerverwaltung über den SQL Server sinnvoll. Der Parameter `DISABLENETWORKPROTOCOLS=0` lässt schließlich den Netzwerkzugriff über TCP/IP zu. So konfiguriert, kann die eigentliche Installation beginnen. Ein Neustart des Rechners lädt schließlich die neue SQL Server-Dienstinstanz.

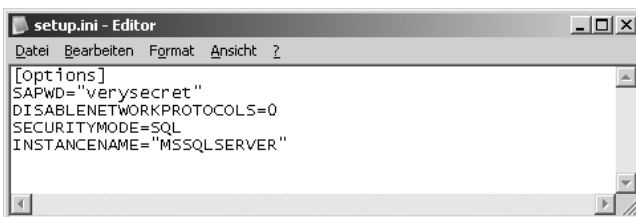


Abbildung 15.7: Die Anpassung der `setup.ini` vor der MSDE-Installation

Die Ports, die vom MS SQL Server belegt werden, lauten 1433 für den SQL Server und 1434 für den SQL Monitor. Mindestens der Port 1433 (TCP) sollte für einen erfolgreichen Netzwerkzugriff geöffnet sein. Darauf sollten Sie achten, wenn Sie Ihre Firewall neu justieren.

Das Fehlen eines SQL Management Systems können Sie übrigens mit dem freien Administrationstool *DbaMgr2k* etwas ausgleichen. Die URL der Homepage von DbaMgr2k lautet <http://www.asql.biz>.

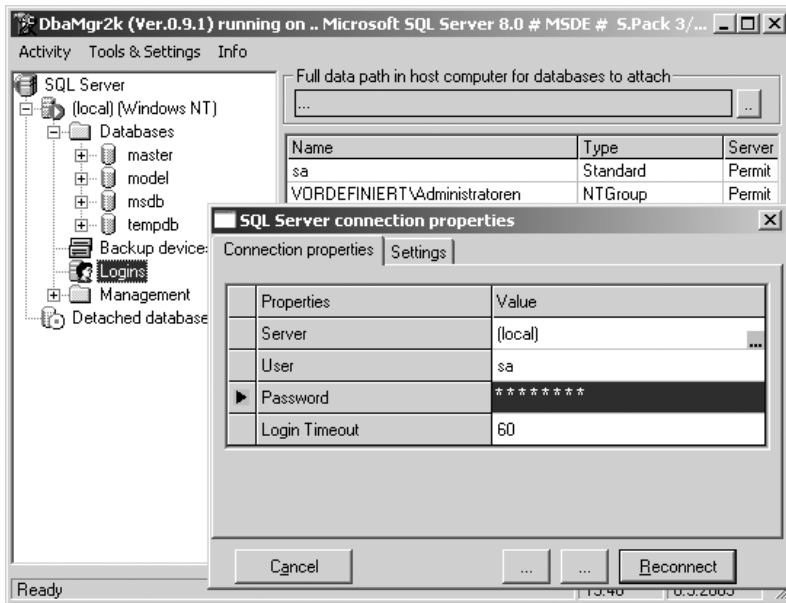


Abbildung 15.8: Der Zugriff auf MSDE mit freien Konfigurationstool DbaMgr2k.

15.2 Eine Beispieldatenbank einrichten

Für dieses Beispiel benutzen wir (nur) den SQL Express Server 2005 sowie das SQL Server Management Studio Express mit seinem SQL Query Editor.

Loggen Sie sich in das SQL Server Management Studio ein und legen Sie eine neue SQL-Anweisung über *New Query* an. Richten Sie nun eine neue Datenbank mit dem Namen *cdcol* sowie die Tabelle *cds* inklusive eines Datensatzes ein. Sie kennen diese Datenbank schon? Macht nichts. Trotzdem Achtung! Microsoft verlangt hier konsequent *T-SQL* (Transact SQL). Wenn Sie bereits mit SQL, beispielsweise mit dem MySQL-Server, Erfahrungen gesammelt haben, können Sie hierfür nach der folgenden Faustregel vorgehen: *SQL-Anweisung 1; GO > SQL-Anweisung 2; GO* usw. Alle SQL-Anweisungen in T-SQL brauchen also jeweils den Befehl *GO* für eine Ausführung.

```

CREATE DATABASE cdcol;
GO
USE cdcol;
GO
CREATE TABLE cds (
    id INT NOT NULL IDENTITY,
    titel VARCHAR(200) NOT NULL,
    interpret text,
    jahr INT default NULL,
    PRIMARY KEY (id)
);

GO
INSERT INTO cds VALUES ('Beauty', 'Ryuichi Sakamoto', 1990);
GO

```

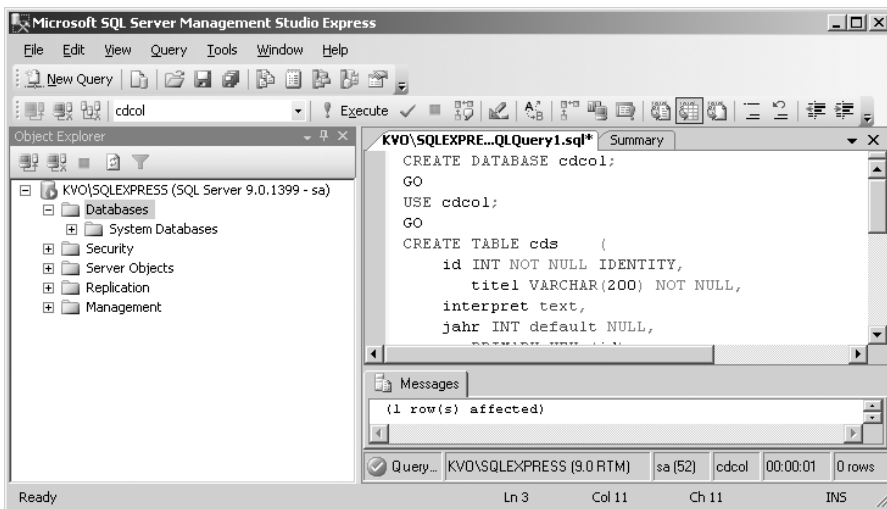


Abbildung 15.9: Ein SQL Query in T-SQL im SQL Server Management Studio

Über *Execute* führen Sie Ihre T-SQL-Syntax aus. Danach aktualisieren Sie die gesamte Sicht mit der Taste **[F5]**. Und schon sehen Sie Ihre neue Datenbank *cdcol*. Optional können Sie übrigens auch auf das Konsolenprogramm *SQLCMD* zurückgreifen. Das Programm finden Sie im SQL Server-Verzeichnis unter *90\Tools\bin*. Im folgenden Beispiel wird der neue Benutzer *zaphod* als *sysadmin* auf der Kommandozeile eingerichtet.

```

<pfad>\Microsoft SQL Server\90\Tools\bin>SQLCMD -S localhost -U sa -P geheim
1> sp_addLogin 'zaphod', 'password'
2> GO
1> sp_addsrvrolemember 'zaphod', 'sysadmin'
2> GO
1> EXIT

```

15.3 Datenbankzugriff mit PHP unter Linux

Für einen Zugriff auf den MS SQL Server unter Linux arbeitet XAMPP mit der freien Bibliothek *FreeTDS*. FreeTDS ist eine Open-Source-Umsetzung basierend auf dem TDS-Protokoll (Tabular Data Stream). Dieses Protokoll sowie seine Varianten werden von den Microsoft SQL Server-Clients für die Kommunikation benutzt. Die FreeTDS-Homepage lautet <http://www.freetds.org>.

Im XAMPP ist die FreeTDS-Bibliothek standardmäßig bereits in PHP einkompiliert: Das Modul wird sowohl über `phpinfo()` unter <http://localhost/xampp/phpinfo.php> als auch die Information von PHP über Konsole mit

```
/opt/lampp/bin/php -i
```

angezeigt. Damit sind alle Voraussetzungen gegeben, es fehlen noch die korrekten Einstellungen für Ihren konkreten Zugriff.



Hinweis

Die während der Arbeit an diesem Buch aktuelle *FreeTDS*-Version in der *Version 0.63* unterstützte nicht den SQL Server 2005 (= Version 9.0). Auch das kann sich geändert haben und der neueste XAMPP besitzt nun eine Version, die mit dem SQL Server 2005 kompatibel ist. Hier können wir Ihnen nur eins raten: Ausprobieren. Das hier vorliegende Beispiel bezieht sich also auf den MS SQL Server 2000.

Die zentrale Datei für die Konfiguration von FreeTDS heißt `freetds.conf`. Sie finden diese Konfiguration im XAMPP als `/opt/lampp/etc/freetds.conf`. Die Datei unterteilt sich in die globalen und damit übergreifenden Einstellungen und die vom Benutzer definierten Abschnitte. In der Regel funktionieren die Vorgaben im globalen Abschnitt `[global]` reibungslos, weshalb Sie sich der Definition Ihres individuellen Servers widmen können. In diesem Szenario ist ein *Windows 2000 Server* mit einem *MS SQL Server 2000* mit der IP-Nummer 192.168.100.20 im Einsatz. Unser *XAMPP für Linux* soll via PHP auf die Datenbanken zugreifen. Dafür suchen Sie am besten ein vorgefertigtes Beispiel wie `[MyServer70]` in der `freetds.conf`, passen diesen Abschnitt nach Ihren Erfordernissen an und aktivieren die Zeilen durch Entfernen der Semikola am Zeilenanfang. Folgende Einstellungen werden für dieses Beispiel vorgenommen:

```
[MyMSQLServer2000]
    host = 192.168.100.20
    port = 1433
    tds version = 7.0
```

```

# Command and connection timeouts
;
timeout = 10;
;
connect timeout = 10;

# This is a Sybase hosted database server, if you are directly on the
# net you can use it to test.
[Sybase]
host = 192.138.151.39
port = 4444
tds version = 5.0

# A typical Microsoft SQL Server 7.0 configuration
[MySQLServer2000]
host = 192.168.100.20
port = 1433
tds version = 7.0

# If you get out of memory errors, it may mean that your client
# is trying to allocate a huge buffer for a TEXT field.
# (Microsoft servers sometimes pretend TEXT columns are
# 4 GB wide!) If you have this problem, try setting
# 'text size' to a more reasonable limit

```

Abbildung 15.10: Die *freetds.conf* im VI(M) Editor unter Linux

Der Name des Parameters `[MySQLServer2000]` kann frei gewählt werden. Der Name entspricht einer späteren Zuordnung zu den nachfolgenden Parametern bei einem konkreten PHP-Zugriff. Dazu kommen wir aber noch. Die weiteren Einstellungen sind nahezu selbsterklärend. Einzig die Festlegung *TDS-Version* muss hervorgehoben werden. Hier wurde nämlich nicht eine Version 8.0 (= 2000) gewählt, sondern bewusst die Version 7.0. Bei der Anfertigung dieses Kapitels lag *FreeTDS* in der Version 0.63 vor, in der die Unterstützung für *MS SQL 2000* (= Version 8.0) noch experimenteller Natur war. Insbesondere mit *DateTime*- oder *Money*-Feldformaten im *MS SQL Server 2000* kam es bei der Protokollversion 8.0 zu unvorhersehbaren Schwierigkeiten. Tatsächlich können diese Probleme aber nach der Drucklegung dieses Buches bereits behoben sein. Im Zweifelsfall ist jedoch `tds version = 7.0` vorzuziehen.

Um die Änderungen in der *freetds.conf* zu laden, starten Sie unbedingt den Apache HTTPD neu:

```
/opt/lampp/lampp restartapache
```

Für den Test der neuen Verbindung reicht nun ein simples PHP-Skript aus.

Listing 15.1: Das Programm freetds.php

```

<?php
/* Verbindung herstellen auf Grundlage der definierten Parametern in der
freetds.conf [MyMSQLServer2000] */
if(@mssql_connect("MyMSQLServer2000","sa","passwort"))
{
    echo "Erfolgreich verbunden mit MS SQL Server";
    mssql_close();
}
else
{
    echo "Fehler! Konnte keine Verbindung zu MS SQL Server herstellen!";
}
exit;
?>

```

Der Versuch eines Verbindungsaufbaus über die Zeile

```
mssql_connect("192.168.100.20","sa","passwort")
```

hingegen scheitert, da unter Linux im Gegensatz zu XAMPP unter Windows die Kommunikation über *FreeTDS* (*freetds.conf*) geschieht. Weitere Einstellungen in der *php.ini* sind damit hinfällig. In unserem Beispiel korrekt ist der Zugriff mit:

```
mssql_connect("MyMSQLServer2000","sa","passwort")
```

Für die Implementierung weiterer SQL-Statements in Ihrem Programm lesen Sie bitte den jetzt folgenden Windows-Abschnitt. Die dort angegebenen SQL-Anweisungen gelten ohne Einschränkungen auch für FreeTDS unter Linux.

15.4 Datenbankzugriff mit PHP unter Windows

Einfacher gestaltet sich der PHP-Zugriff auf einen MS SQL Server mit XAMPP für Windows. Auch hier ist das MS SQL-Modul in der *php.ini* bereits geladen. Weitere Einstellungen können wahlweise in der Sektion *[mssql]* vorgenommen werden.

Konfigurationsdatei: <xampp>\apache\bin\php.ini

PHP 5-Modul:<xampp>\php\ext\php_mssql.dll

Da war doch noch etwas? Ja! Wie eingangs bereits angesprochen, funktionierte der Zugriff auf den SQL Server 2005 Express nicht in der uns vorliegenden PHP-Version 5.1.2. Unproblematisch war hingegen der Zugriff auf den MS SQL 2000 (inklusive MSDE). Wir gehen hier aber einmal davon aus, dass die PHP-Programmierer in den neueren Versionen eine Schnittstelle zum SQL Server 2005 eingebaut haben. In jedem Fall sieht ein einfacher Kontakt wie folgt aus:

Listing 15.2: Das PHP-Programm mssql.php

```

<html>
<head>
<title>PHP und MS SQL</title>
</head>
<body>
<h1>PHP und MS SQL</h1>
<table border="1">
<tr>
    <th>Interpret</th>
    <th>Titel</th>
    <th>Jahr</th>
</tr>
<?php
if(@mssql_connect("localhost","sa","geheim"))
{
    mssql_select_db("cdcol") or die("Kann die Datenbank nicht öffnen!");

    $result = mssql_query("SELECT * from cds");
    while ($row = mssql_fetch_array($result))
    {
        echo "<tr>";
        echo "<td>".$row['interpret'].</td>";
        echo "<td>".$row['titel'].</td>";
        echo "<td>".$row['jahr'].</td>";
        echo "</tr>";
    }
    mssql_close();
}
else
{
    echo "<h3>Kein Connect zum SQL Server!</h3>";
}
?>
</table>
</body>
</html>

```

15.4.1 Datenbankzugriff mit Perl unter Windows

Für den Zugriff auf den MS SQL Server mit Perl sind zwei Methoden wohl am häufigsten anzutreffen: die Verbindung über *ODBC* und die Verbindung über *OLEDB/ADO*. *ODBC* (Open DataBase Connectivity) nennt Microsoft die normierte Programmierschnittstelle für den Zugriff auf alle gängigen SQL-Datenbankserver. Mit der Alternative *ADO* (*ActiveX Data Objects*) stellt Microsoft eine neuere und schnellere Schnittstelle zu SQL-Datenbanken bereit. Das *DBI::ADO*-Modul ist nur unter *Windows* brauchbar, weil nur in Windows-Systemen implementiert.

Beide Treiber basieren auf dem DBI-Modul, einem sog. »Abstraktionslayer« für die Normierung, d.h. Vereinheitlichung, von Datenbankzugriffen unter Perl. Für diesen Layer gibt es verschiedene DBD-Treiber für die unterschiedlichen Datenbankserver. Diese Treiber ermöglichen die datenbanktypische Anmeldung sowie weitere Interaktionen. Im Klartext: Der BDB-Treiber ermöglicht die Kommunikation mit dem SQL Server und das DBI-Modul vereinheitlicht diese Kommunikation. Das folgende Beispiel benutzt die zu diesem Zeitpunkt noch verbreitete DBI::ODBC-Schnittstelle für den Verbindungsaufbau zum Microsoft SQL Server. Sie benötigen für Ihren XAMPP das entsprechende *XAMPP Perl-Add-On*. In diesem Fall kam das XAMPP Perl-Add-On mit Perl 5.8.7 und *mod_perl 2.0.1* für den Apache 2.0.54 zum Einsatz. Als Testdatenbank verwendeten wir die bereits bekannte Datenbank *cdcol* mit der Tabelle *cds*, die wir bereits für den SQL-Server Express eingerichtet haben.



Abbildung 15.11: DNS-Konfiguration auf dem Rechner KVO für den SQL Server 2005

Zu Beginn richten Sie eine *ODBC-System-Datenquelle* (System-DSN) über **START > EINSTELLUNGEN > SYSTEMSTEUERUNG > VERWALTUNG > DATENQUELLEN (ODBC)** ein. Im Gegensatz zu einer *Benutzer-DSN* steht die *System-DSN* unabhängig von dem aktiven Benutzerkonto jedem jederzeit zur Verfügung. Als Treiber wählen Sie den SQL Server und richten die Datenquelle mit dem optionalen Namen *mssql2005* ein. Sie können die neue DSN natürlich auch anders benennen.

Zwei Punkte sollten Sie bei der Einrichtung beachten. Erstens muss die MS SQL Server-Authentifizierung verwendet werden. Zweitens richten Sie hier gleich eine Standarddatenbank ein. Diese DB wird bei einer erfolgreichen Anmeldung als Standard benutzt und erspart Ihnen so Zeit bei der Programmierung. Die Eingabe des Benutzers, in diesem Fall des SQL-Administrators *sa*, und des Passworts ist hier einzig für den Test der DSN notwendig. Das Passwort wird aus Sicherheitsgründen hier nicht

gespeichert. Sowohl Benutzer als auch Passwort müssen Sie damit in Ihrem Perl-Programm permanent übergeben. Für mehr Sicherheit in einem langfristigen und öffentlichen Einsatz empfehlen wir Ihnen, einen eigenen Benutzer mit beschränkten Zugriffsrechten einzurichten, der einzig auf seine Datenbank vollen Zugriff besitzt. Für das kleine Beispiel vernachlässigen wir dieses Problem; es geht uns ja um das methodische Vorgehen und nicht zuletzt um die Programmierung!

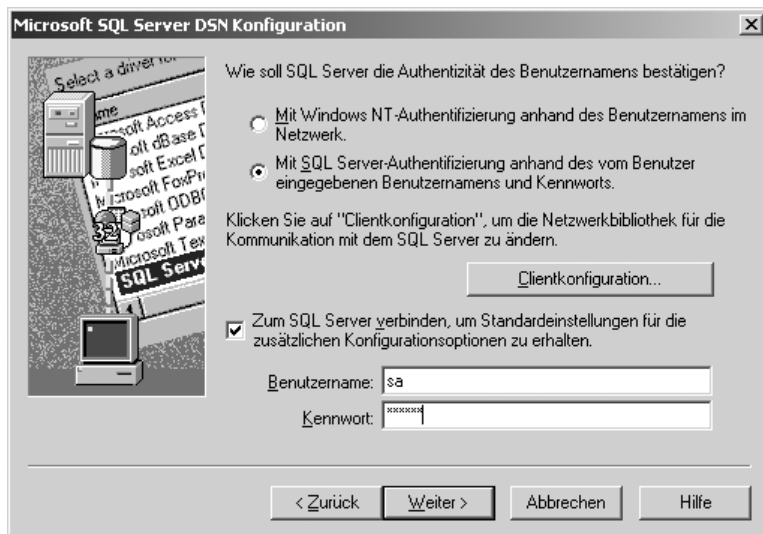


Abbildung 15.12: Die Datenquelle mit SQL Server-Authentifizierung

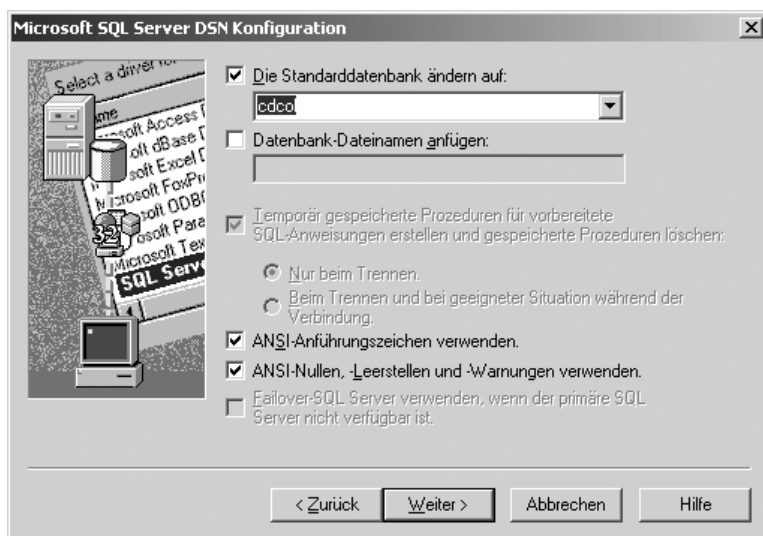


Abbildung 15.13: cdcol ist nun die Standarddatenbank

Mit dem Beispiel `mssql.pl` testen Sie zu Beginn die Verbindung. Das Skript legen Sie in dem Verzeichnis `<xampp>\htdocs\modperl` ab, da wir hier mit dem Apache-Perl-Modul arbeiten. Der Aufruf des neuen Programms erfolgt über die URL `http://localhost/perl/mssql.pl`.

```
use DBI;
print "Content-Type: text/html\n\n";
my $user = 'sa';
my $pass = 'geheim';
my $dbh = DBI->connect('dbi:ODBC:mssql2005', $user, $pass);
$dbh->disconnect();
```

Im ersten Parameter der `DBI > Connect()`-Methode wählen Sie den DBi-ODBC-Treiber und setzen den Namen der von Ihnen erstellten System-DSN: `dbi:ODBC:mssql2005` bestimmt. Der Name `mssql2005` ist hier nichts anderes als der Name der eben eingerichteten DSN. Dieser heißt bei Ihnen vielleicht anders und sollte von Ihnen selbstverständlich angepasst werden.

Listing 15.3: Das Programm `mssql.pl`

```
use DBI;
print "Content-Type: text/html\n\n";
my $user = 'sa';
my $pass = 'geheim';
my $dbh = DBI->connect('dbi:ODBC:mssql2005', $user, $pass);
my $sql = "SELECT * from cds";
my $sth = $dbh->prepare($sql)
    or die "Can't prepare statement: $DBI::errstr";
$sth->execute();
while (@row = $sth->fetchrow_array())
{
    print "<h3>CD Musik Datenbank</h3> <p>";
    print "ID: ".$row[0]."<br>";
    print "Interpret: ".$row[1]."<br>";
    print "Album: ".$row[2]."<br>";
    print "Erscheinungsjahr: ".$row[3]."<br>";
}
$dbh->disconnect();
```




16 Oracle

Während der Arbeit an diesem Buch bot *Oracle* auf seiner Homepage (<http://www.oracle.com/database/>) zwei Versionen an: zum einen die kleinere *Oracle Database 10g Express Edition* und zum anderen die *Oracle Database 10g Release 2*. Bevor Sie die Pakete herunterladen können, müssen Sie sich bei Oracle einen kostenfreien Account einrichten. Für Einsteiger ist die nicht so ressourcenhungrige Express-Version zu empfehlen. Wer jedoch die Installation der großen Version bevorzugt, sollte einen speicherstarken Rechner vorhalten.

Die Systemanforderungen von Oracle Database 10g:

- mindestens 512 RAM
- mindestens 1 Gbyte SWAP bzw. Auslagerungsspeicher
- mindestens 400 Mbyte freier Speicherplatz im temporären Ordner
- eine Pentium-kompatibler CPU mit mindestens 500 MHz
- mindestens 2,5 Gbyte freier Festplattenplatz für das Oracle-System und 1 Gbyte für die Datenbank bei einem System mit Minimalanforderungen

16.1 Installation

Der Datenbankserver von Oracle braucht also vor allem Arbeits- sowie Festplattenspeicher. Doch die Erfüllung der Minimalanforderungen ist nicht immer eine Erfolgsgarantie. Bei einem Versuch auf einem Rechner mit 512 Mbyte Arbeitsspeicher und genügend Festplattenplatz sowie einem AMD ATHLON XP 1800+ als Prozessor fror das System während der Installation ein, obwohl der Oracle-Installer zuvor meldete, dass alle Systemanforderungen erfüllt seien. Ähnliche Qualen erleiden auch Linux-Benutzer bei der Installation. Nehmen Sie also ein Scheitern nicht so schwer. Sie wären nicht der Erste.

In diesem Kapitel verwenden wir die Oracle Express Edition. Wie schon erwähnt, ist sie viel kleiner und »leichter« als die große Version und auch viel bequemer zu installieren. Wenn Sie nicht unbedingt die große Version brauchen, dann sollten Sie wirklich die kleine nehmen. Sie wird Ihnen viel Arbeit ersparen.

Laden Sie also zunächst die Version für Linux oder Windows von der Oracle-Website unter <http://www.oracle.com/technology/products/database/xe/index.html> herunter. Sie sollten dazu auf jeden Fall DSL haben: Die Linux-Version ist etwa 220 Mbyte groß und die Windows-Version etwa 170 Mbyte.

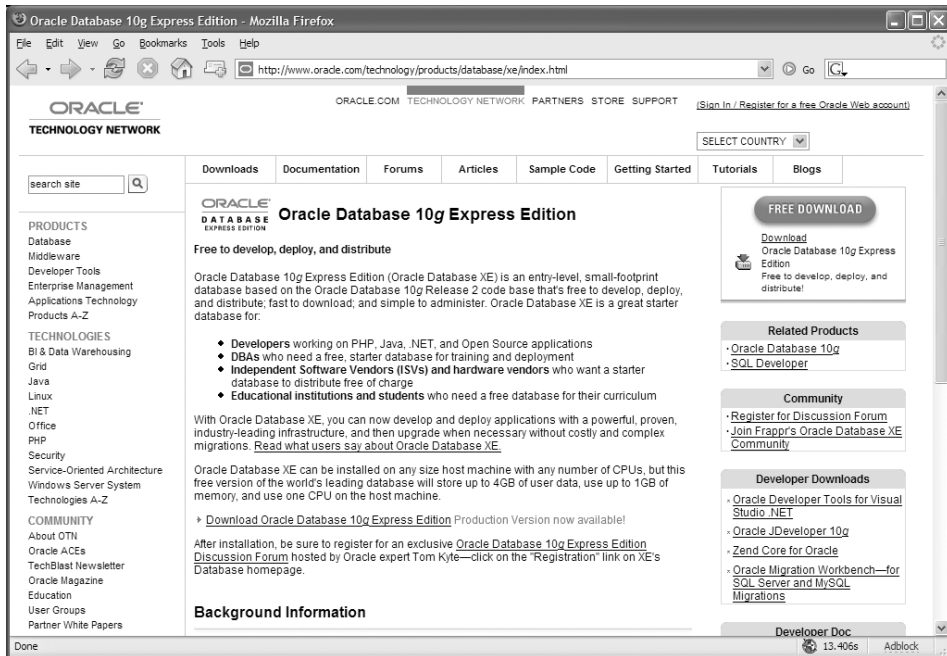


Abbildung 16.1: Die Express Edition von Oracle

16.1.1 Windows

Wie schon eingangs erwähnt, ist die Installation von Oracle Express relativ einfach und vom Aufwand her gar kein Vergleich zur Installation einer vollständigen Oracle-Version:

1. Zur Installation benötigen Sie unbedingt Administrationsrechte. Melden Sie sich also als Hauptbenutzer oder Administrator auf Ihrem System an.
2. Nach dem Start der `OracleXE.exe` werden Sie zunächst begrüßt und um Ihre Zustimmung zu den Lizenzbedingungen gebeten.
3. Im nächsten Schritt werden Sie nach dem Installationsziel gefragt. Vorgeschlagen wird das Verzeichnis `C:\oracle\xe\`. Wenn Sie hier genügend Festplattenplatz haben – Oracle Express benötigt etwa 166 Mbyte –, dann sollten Sie es bei dieser Wahl belassen.

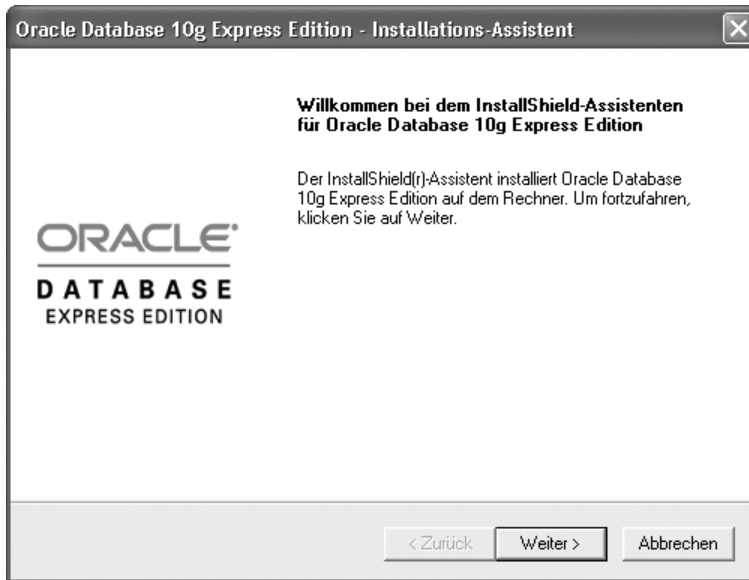


Abbildung 16.2: Die Installation von Oracle Express ist unter Windows recht komfortabel.

4. Als Nächstes werden Sie nach dem Passwort für die beiden Administrations-Datenbank-Benutzer *SYS* und *SYSTEM* gefragt. In den folgenden Beispielen gehen wir davon aus, dass hier das Passwort *geheim* gewählt wurde. Wählen Sie ein Passwort, das Ihnen recht ist.
5. Jetzt erhalten Sie eine Zusammenfassung aller Einstellungen. Mit einem Klick auf *WEITER>* starten Sie die Installation.

Nach Abschluss der Installation bietet Ihnen der Installer an, die »Datenbank-Homepage zu starten«. Lassen Sie das Kästchen aktiviert und klicken Sie auf *FERTIG*. Nun können Sie mit dem Einrichten einer Oracle-Beispieldatenbank fortfahren.

16.1.2 Linux

Verglichen mit der Installation der großen Oracle-Version ist die Installation von Oracle Express unter Linux – ähnlich wie bei Windows – sehr, sehr einfach. Zwar kommt sie nicht mit einem so schönen grafischen Installer, aber wer will das auf einem Serversystem auch schon.

1. Melden Sie sich als *root* auf Ihrem System an und wechseln Sie in das Verzeichnis, in dem das heruntergeladene RPM von Oracle liegt.
2. Rufen Sie nun den *rpm*-Befehl auf, um das Paket zu installieren:

```
rpm -ivh oracle-xe-10.2.0.1-0.1.i386.rpm
```

Eventuell erhalten Sie die folgende Fehlermeldung:

error: Failed dependencies:

libaio >= 0.3.96 is needed by oracle-xe-10.2.0.1-0.1.i386

Installieren Sie in diesem Fall das Paket *libaio* mit dem Paket-Manager Ihrer Distribution nach und starten Sie den obigen `rpm`-Befehl noch einmal.

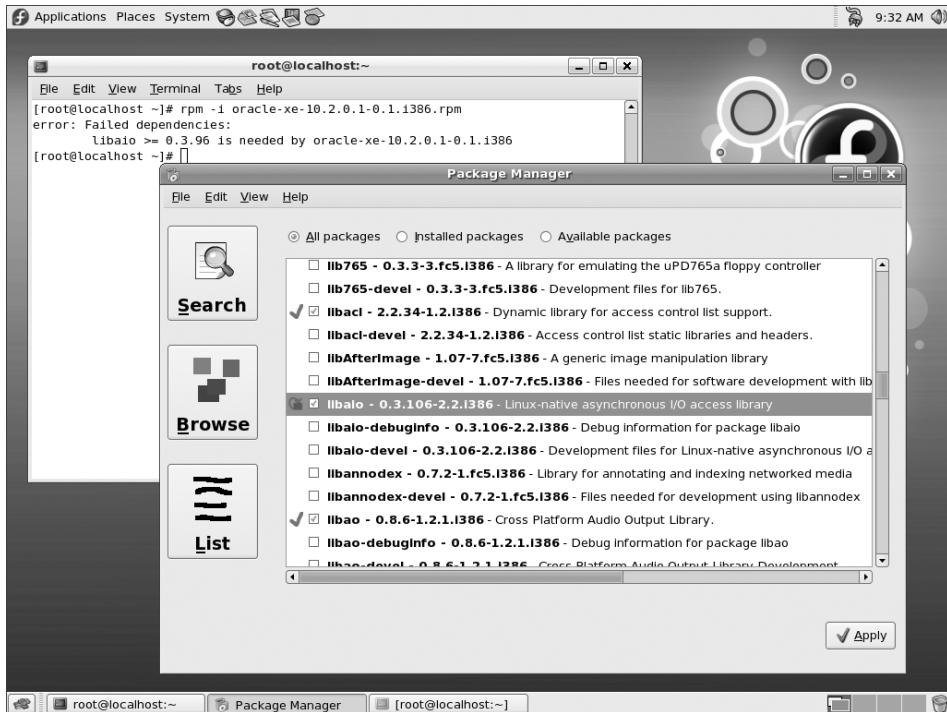


Abbildung 16.3: Oracle Express benötigt das Paket *libaio*.

3. Starten Sie nun das Setup-Tool von Oracle Express:

```
/etc/init.d/oracle-xe configure
```

4. Auf dem Bildschirm erscheint nun ein kleiner Erklärungstext und Sie werden nach der Portnummer gefragt, unter der die webbasierte Administrationsoberfläche (genannt »Datenbank-Hompage«) erreichbar sein soll:

Oracle Database 10g Express Edition Configuration

This will configure on-boot properties of Oracle Database 10g Express Edition. ↵
 The following questions will determine whether the database should be starting ↵
 upon system boot, the ports it will use, and the passwords that will be used ↵
 for database accounts. Press <Enter> to accept the defaults. Ctrl-C will abort.

Specify the HTTP port that will be used for HTML DB [8080]:

Wenn Sie keine besonderen Gründe haben, von diesem Vorschlag abzuweichen, dann drücken Sie hier einfach nur die -Taste und bestätigen so die vorgeschlagene Einstellung von Port 8080.

5. Im nächsten Schritt werden Sie nach dem Port gefragt, unter dem die Oracle-Datenbank erreichbar sein soll:

Specify a port that will be used for the database listener [1521]:

Sie sollten hier in jedem Fall einfach nur die -Taste drücken und die Vorgabe von Port 1521 akzeptieren.

6. Als Nächstes werden Sie nach dem Passwort für die Administrationsbenutzer der Oracle-Datenbank gefragt. Oracle macht es Ihnen einfach und Sie müssen sich nur ein Passwort ausdenken und dieses wird dann für alle Benutzer gesetzt.

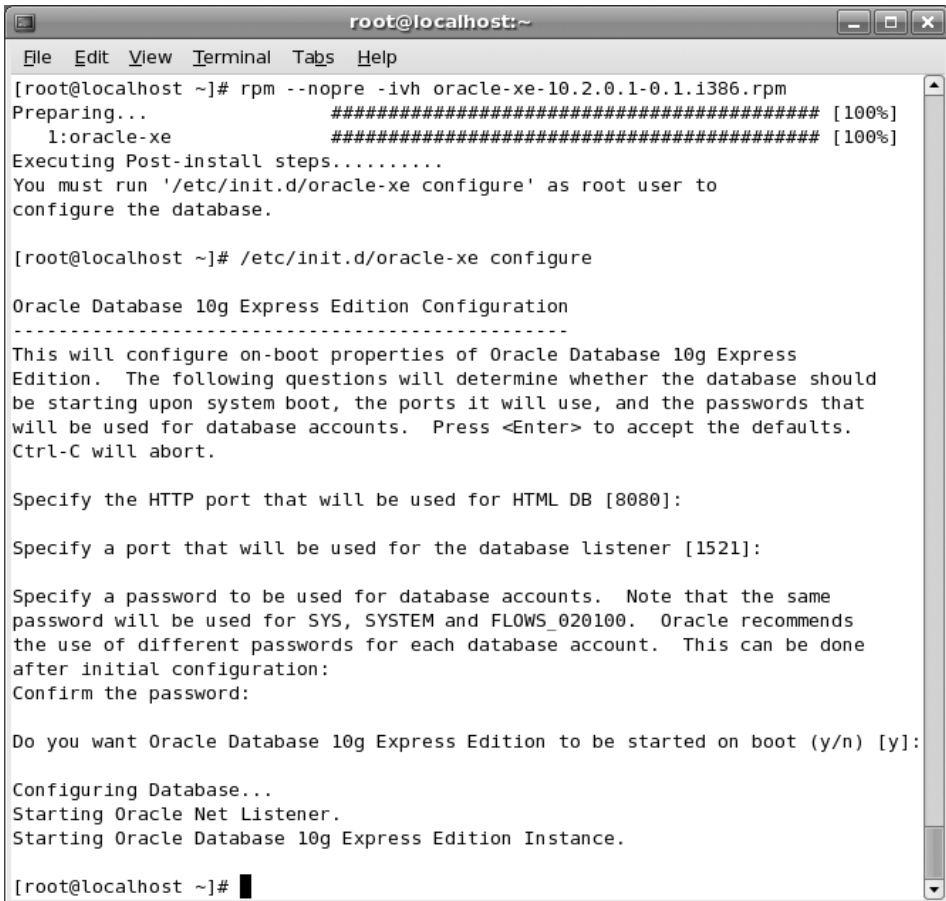
Specify a password to be used for database accounts. Note that the same ↵
 password will be used for SYS, SYSTEM and FLOWS_020100. Oracle recommends the ↵
 use of different passwords for each database account. This can be done after ↵
 initial configuration:
 Confirm the password:

7. Im letzten Schritt werden Sie noch gefragt, ob Sie wünschen, dass Oracle auch beim nächsten Neustart des Systems mitgestartet werden soll:

Do you want Oracle Database 10g Express Edition to be started on boot (y/n) [y]:

Je nachdem, was Ihnen lieber ist, geben Sie hier einfach »y« oder »n« ein.

Damit ist die initiale Konfiguration von Oracle Express abgeschlossen und Oracle Express wird automatisch gestartet. Nun können Sie mit dem Einrichten einer Oracle-Beispieldatenbank fortfahren.



```
root@localhost:~  
File Edit View Terminal Tabs Help  
[root@localhost ~]# rpm --noprogress -ivh oracle-xe-10.2.0.1-0.1.i386.rpm  
Preparing... ##### [100%]  
1:oracle-xe ##### [100%]  
Executing Post-install steps.....  
You must run '/etc/init.d/oracle-xe configure' as root user to  
configure the database.  
  
[root@localhost ~]# /etc/init.d/oracle-xe configure  
  
Oracle Database 10g Express Edition Configuration  
-----  
This will configure on-boot properties of Oracle Database 10g Express  
Edition. The following questions will determine whether the database should  
be starting upon system boot, the ports it will use, and the passwords that  
will be used for database accounts. Press <Enter> to accept the defaults.  
Ctrl-C will abort.  
  
Specify the HTTP port that will be used for HTML DB [8080]:  
  
Specify a port that will be used for the database listener [1521]:  
  
Specify a password to be used for database accounts. Note that the same  
password will be used for SYS, SYSTEM and FLOWS_0020100. Oracle recommends  
the use of different passwords for each database account. This can be done  
after initial configuration:  
Confirm the password:  
  
Do you want Oracle Database 10g Express Edition to be started on boot (y/n) [y]:  
  
Configuring Database...  
Starting Oracle Net Listener.  
Starting Oracle Database 10g Express Edition Instance.  
  
[root@localhost ~]#
```

Abbildung 16.4: Die Installation von Oracle Express passt auf eine Terminalseite.

16.2 Beispielbenutzer und -datenbank einrichten

Wenn Sie gerade die Installation unter Windows durchgeführt haben, werden Sie schon die »Datenbank-Homepage« auf Ihrem Bildschirm vor sich haben. Sollten Sie sie nicht sehen, rufen Sie bitte in Ihrem Browser die Adresse <http://127.0.0.1:8080/apex/> auf. Sollte Oracle Express nicht auf Ihrem lokalen Rechner laufen, dann verwenden Sie den Namen oder die IP-Adresse des Rechners, auf dem Sie Oracle installiert haben.

Unter Windows können Sie die »Datenbank-Homepage« auch über das Startmenü aufrufen: ALLE PROGRAMME > ORACLE DATABASE 10G EXPRESS EDITION > GEHE ZU DATENBANK-HOMEPAGE.



Hinweis

Eigentlich ist der Titel dieses Abschnitts irreführend: Es wird nur ein Beispielbenutzer aktiviert und eine Beispieldatenbank eingerichtet. Das tatsächliche Einrichten eines neuen Benutzers und einer Datenbank wäre zu komplex und würde für dieses Buch zu weit in Oracle-Details gehen. Wie schon analog an anderen Stellen erwähnt: Dies ist ein XAMPP-Buch und kein Oracle-Buch.

Um die Symmetrie mit den anderen Datenbank-Kapiteln zu behalten, haben wir den Titel so gelassen – auch wenn er ein wenig irreführend ist. Was hier zählt, ist das Resultat, nämlich ein funktionierender Benutzer und eine funktionierende Datenbank.

16.2.1 Beispielbenutzer aktivieren

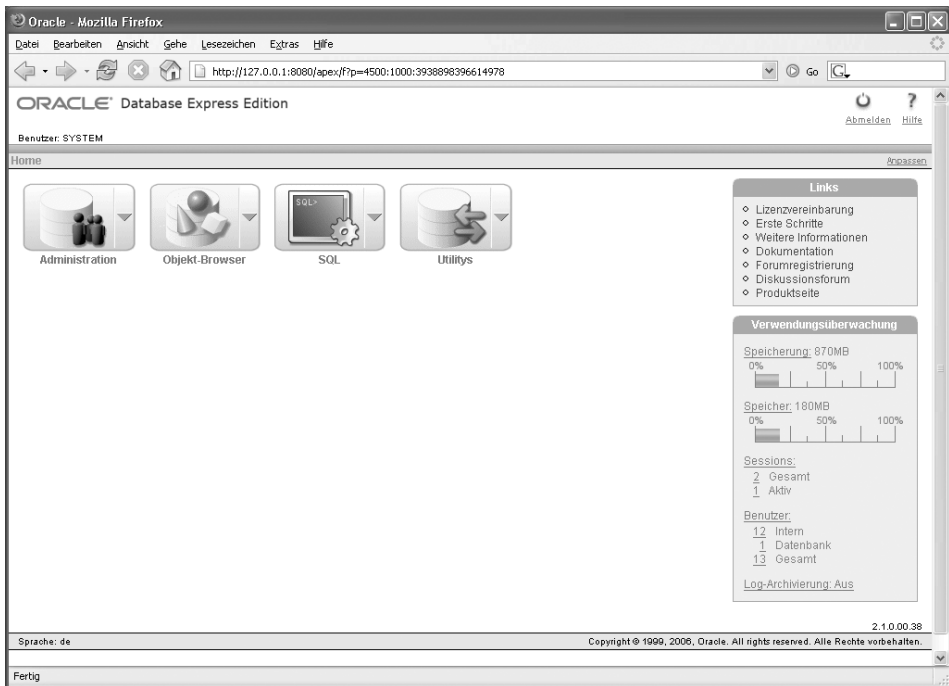


Abbildung 16.5: Die sog. »Datenbank-Homepage« ist die webbasierte Administrationssoftware von Oracle Express

1. Melden Sie sich mit der Benutzerkennung *SYSTEM* und dem Passwort, das Sie während der Installation ausgewählt haben, an.

Es erscheint das Hauptmenü der Administrationsoberfläche (siehe Abbildung 16.5).

2. Klicken Sie auf **ADMINISTRATION** und danach auf **BENUTZERVERWALTUNG**.
3. Wählen Sie hier den Beispielbenutzer *HR* aus. Es sollte nur dieser eine Benutzer angezeigt werden.

Datenbankbenutzer verwalten Abbrechen Löschen Benutzer ändern

Benutzername **HR**

Kennwort

Kennwort bestätigen

Kennwort abgelaufen

Account-Status: Gesperrt

Default Tablespace: **USERS**

Temporärer Tablespace: **TEMP**

Abbildung 16.6: Hier können Sie den Beispielbenutzer *HR* aktivieren.

4. Aktivieren Sie nun den Beispielbenutzer *HR*, indem Sie den **ACCOUNT-STATUS** von *Gesperrt* auf *Freigegeben* setzen und dem Benutzer ein Passwort zuweisen. Für die folgenden Beispiele haben wir hier das Passwort *geheim* gewählt.
5. Klicken Sie auf **BENUTZER ÄNDERN**, um die Änderungen zu speichern.
6. Melden Sie sich nun über den **ABMELDEN**-Knopf oben rechts ab. Es erscheint die bestätigende Meldung *Sie sind abgemeldet*.
Darunter finden Sie einen Link zurück zur Anmeldeoberfläche. Klicken Sie auf diesen Link, um zum nächsten Dialog zu gelangen.

16.2.2 Beispieltabelle einrichten

1. Melden Sie sich nun mit der Benutzerkennung *HR* und dem Passwort, das Sie diesem Benutzer zugewiesen haben, an.
Es erscheint wieder das Hauptmenü der Administrationsoberfläche.
2. Wählen Sie nun den Menüpunkt **SQL** und danach den Menüpunkt **SQL-BEFEHLE**.
3. Hier können Sie nun recht bequem SQL-Befehle ausführen. Geben Sie der Reihe nach die folgenden SQL-Befehle ein, um die Beispieltabelle einzurichten und mit Beispieldaten zu füllen.
Klicken Sie nach jedem SQL-Befehl oben rechts auf **AUSFÜHREN**.

```
CREATE TABLE cds ( id number, titel varchar(200), interpret varchar(200), jahr
number, PRIMARY KEY (id));
INSERT INTO cds VALUES (1, 'Beauty', 'Ryuichi Sakamoto', 1990);
INSERT INTO cds VALUES (2, 'Goodbye Country (Hello Nightclub)', 'Groove Armada', 2001);
INSERT INTO cds VALUES (3, 'Glee', 'Bran Van 3000', 1997);
```

So, das war's. Oracle Express ist nun so weit vorbereitet, dass Sie im nächsten Schritt von PHP oder Perl auf diese Datenbank zugreifen können.

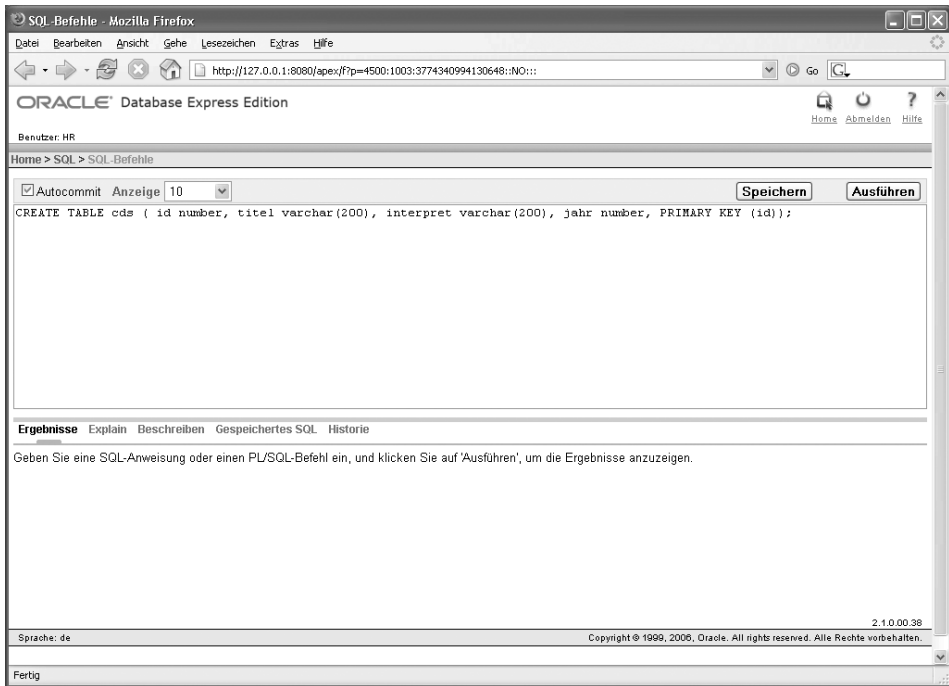


Abbildung 16.7: Über SQL-BEFEHL können Sie auf der Weboberfläche bequem SQL-Befehle in Oracle Express ausführen.

16.3 Aktivierung der PHP-Oracle-Unterstützung unter Linux

In XAMPP für Linux ist die Unterstützung für Oracle-Datenbanken nicht standardmäßig aktiv und muss also zunächst aktiviert werden. Damit sie aktiviert werden kann, muss entweder ein Oracle-Server oder der *Oracle Instant Client* installiert sein.

Mit folgenden Schritten können Sie die OCI8-Unterstützung in PHP aktivieren. Sollten Sie bereits Oracle oder den Instant Client installiert haben, dann können Sie die ersten beiden Schritte überspringen. Sie benötigen dann lediglich noch die Information, wo auf Ihrem System sich Oracle oder der Instant Client befindet.

1. Laden Sie den Instant Client für Linux von der Oracle-Website <http://www.oracle.com/technology/software/index.html> herunter.
2. Installieren Sie den Instant Client mit folgendem Befehl:

```
rpm -i oracle-instantclient-basic-10.1.0.2-1.i386.rpm
```

3. Nach der Installation des Instant Clients kann die Oracle-Unterstützung in XAMPP aktiviert werden. Rufen Sie dazu das XAMPP-Startskript mit dem Parameter `oci8` auf. Ein kleines dialogbasiertes Skript wird gestartet und fragt nach dem Verzeichnis der Oracle- oder Instant Client-Installation (Benutzereingaben fett hervorgehoben):

```
straylight /# /opt/lampp/lampp oci8
Please enter the path to your Oracle or Instant Client installation:
[/opt/oracle] /usr/lib/oracle/10.1.0.2/client/lib/
installing symlinks...
patching php.ini...
OCI8 add-on activation likely successful.
straylight /#
```

Nach dem letzten Schritt sollte die PHP-OCI8-Erweiterung aktiviert sein. Zur Sicherheit sollten Sie nun mit `phpinfo()` überprüfen, ob die Aktivierung wirklich erfolgreich war.

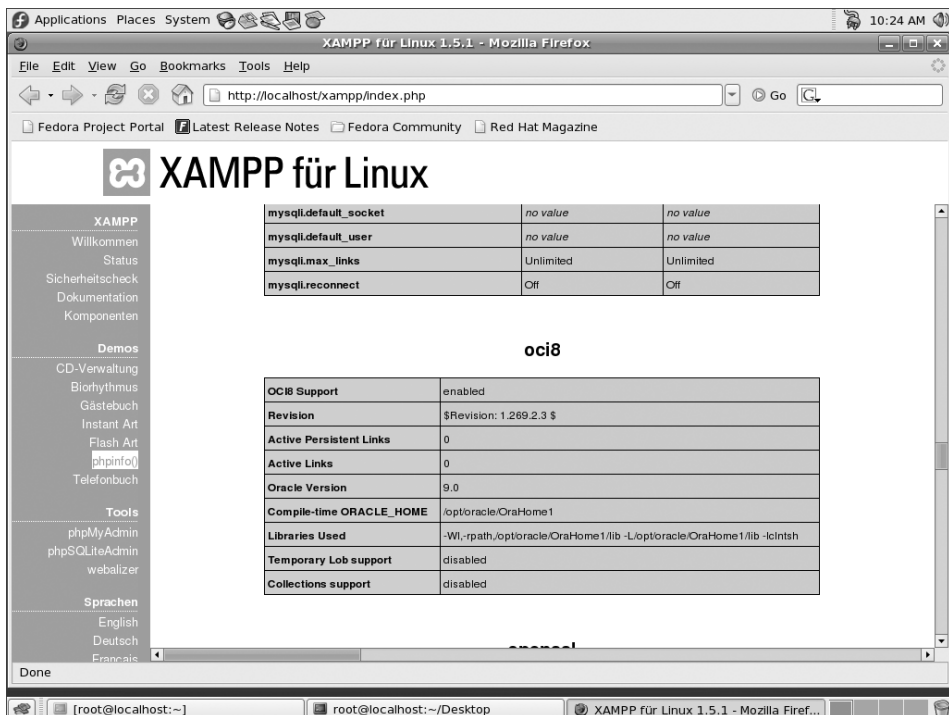


Abbildung 16.8: Über `phpinfo()` sollte überprüft werden, ob die OCI8-Erweiterung erfolgreich aktiviert wurde.

**Hinweis**

Erst seit XAMPP für Linux 1.4.15 ist auch die Verwendung des Instant Clients so einfach möglich. In älteren Versionen muss XAMPP noch manuell an den Instant Client angepasst werden.

16.4 Aktivierung der PHP-Oracle-Unterstützung unter Windows

In XAMPP für Windows muss je nach Version noch das entsprechende Modul in der `php.ini` aktiviert werden.

Konfigurationsdatei: `<xampp>\apache\bin\php.ini`

PHP4- und PHP5-Modul: `extension=php_oci8.dll`

16.5 Datenbankzugriff mit PHP

Ist die OCI8-Erweiterung aktiviert, kann von PHP aus auf eine Oracle-Datenbank zugegriffen werden.

Das gesamte Skript befindet sich noch einmal am Ende dieses Abschnitts und in der XAMPP für Linux-Installation unter `/opt/lampp/htdocs/xampp/contrib/oracle.php`. Windows-Anwender benutzen das Script `<xampp>\htdocs\contrib\oracle.php`.

Verbinden mit der Datenbank

Als Allererstes muss – wie bei jeder anderen Datenbank auch – eine Verbindung von PHP zur Datenbank hergestellt werden. Im Falle von Oracle geschieht das mit dem Befehl `ocilogon()`:

```
$db="(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)
(HOST=localhost) (PORT=1521)))
(CONNECT_DATA=(SERVICE_NAME=x)))"

$c=ocilogon("hr", "geheim", $db);
```

Dem `ocilogon()`-Befehl werden drei Parameter übergeben:

- der Benutzernamen, unter dem man sich mit der Datenbank verbinden möchte,
- das Passwort dieses Benutzers,
- der so genannte *Connect Descriptor* (oder auch *Connect String*)

Mit dem Connect Descriptor wird die Datenbank identifiziert, mit der sich PHP verbinden soll. Er ist im Falle von Oracle recht umständlich, obwohl er eigentlich nur folgende Informationen enthält:

- Rechnername oder IP-Adresse (HOST =)
- Portnummer der Oracle-Datenbank (in der Regel: PORT = 1521)
- Name der Datenbank (SID =)

Hinweis



Seit den Oracle 10g Client Libraries kann anstelle des komplizierten Connect Descriptors alternativ die sog. *Easy Connect*-Syntax verwendet werden:

```
[//]host[:port][/]service_name]
```

Aus dem herkömmlichen Connect Descriptor

```
$db="(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)
(HOST=localhost) (PORT=1521)))
(CONNECT_DATA=(SERVICE_NAME=xe)))"
```

wird dadurch ein wesentlich kürzeres

```
$db="//localhost/xe";
```

SQL-Abfrage ausführen

Nachdem die Verbindung zur Datenbank hergestellt wurde, kann mit der Abfrage der Daten aus der Datenbank begonnen werden.

Zunächst muss die SQL-Abfrage mit `ociparse()` vorbereitet werden:

```
$s = ociparse($c, "SELECT * FROM cds");
```

Mit diesem Befehl wird die Gültigkeit des SQL-Befehls überprüft und der Befehl für die Ausführung vorbereitet. Dieser Schritt ist bei Oracle unbedingt notwendig. Und erst nachdem dieser Schritt erfolgt ist, kann die SQL-Abfrage tatsächlich ausgeführt werden:

```
if(ociexecute($s))
```

Der `ociexecute()`-Befehl sendet den SQL-Befehl an die Oracle-Datenbank. Je nachdem, ob der SQL-Befehl erfolgreich war oder nicht, gibt der Befehl entweder TRUE oder FALSE zurück. Ist der Befehl erfolgreich, kann im nächsten Schritt die Antwort des Datenbankservers gelesen werden.

```

{
    while (ocifetch($s))
    {
        echo "<tr>";
        echo "<td>".ociresult($s, "INTERPRET")."</td>";
        echo "<td>".ociresult($s, "TITEL")."</td>";
        echo "<td>".ociresult($s, "JAHR")."</td>";
        echo "</tr>";
    }
}

```

Mit `ocifetch()` wird nun jede einzelne Ergebniszeile der SQL-Abfrage geholt und mit dem `ociresult()`-Befehl ausgelesen.

War der Befehl hingegen nicht erfolgreich, wird eine entsprechende Fehlermeldung ausgegeben:

```

else
{
    $e = ocierror($s);
    echo htmlentities($e['message']);
}

?>
</table>
</body>
</html>

```

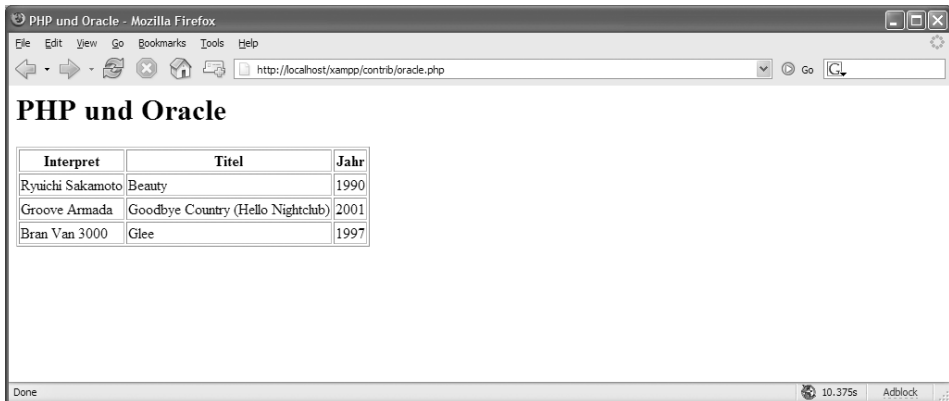


Abbildung 16.9: Der Testlauf von `oracle.php` mit der Demodatenbank

Hier das gesamte Listing im Zusammenhang:

Listing 16.1: Mit PHP auf eine Oracle-Datenbank zugreifen: oracle.php

```
<html>
<head>
<title>PHP und Oracle</title>
</head>
<body>
<h1>PHP und Oracle</h1>
<table border="1">
<tr>
    <th>Interpret</th>
    <th>Titel</th>
    <th>Jahr</th>
</tr>
<?php
    $db="(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)
        (HOST=localhost) (PORT=1521)))
        (CONNECT_DATA=(SERVICE_NAME=xe)))"

    // Oracle 10g
    //$db="//localhost/xe";

    $c=ocilogon("hr", "geheim", $db);

    $s = ociparse($c, "SELECT * FROM cds");

    if(ociexecute($s))
    {
        while (ocifetch($s))
        {
            echo "<tr>";
            echo "<td>".ociresult($s, "INTERPRET")."</td>";
            echo "<td>".ociresult($s, "TITEL")."</td>";
            echo "<td>".ociresult($s, "JAHR")."</td>";
            echo "</tr>";
        }
    }
    else
    {
        $e = ocierror($s);
        echo htmlentities($e['message']);
    }
?>
</table>
</body>
</html>
```



Achtung

In PHP 5 wurden sämtliche Oracle-Befehle um einen Unterstrich `_` erweitert. Also wurde aus `ociparse()` `oci_parse()`, aus `ocireresult()` wurde `oci_result()` und so weiter. Eine Ausnahme bildet `ocilogon()`, das nicht nur um den Unterstrich erweitert, sondern zusätzlich in `oci_connect()` umbenannt wurde.

Das vorangegangene Skript würde in PHP 5 also wie folgt aussehen:

```
$c=oci_connect("hr", "geheim", $db);

$s = oci_parse($c, "select * from cds");

oci_execute($s, OCI_DEFAULT);

while (oci_fetch($s))
{
    echo "<tr>";
    echo "<td>".oci_result($s, "INTERPRET")."</td>";
    echo "<td>".oci_result($s, "TITEL")."</td>";
    echo "<td>".oci_result($s, "JAHR")."</td>";
    echo "</tr>";
}
```

Allerdings funktionieren unter PHP 5 auch noch die alten PHP 4-Befehle. Wenn man also seine Skripte auch noch unter PHP 4 laufen lassen muss, dann sollte man die alte Schreibweise verwenden. Trotzdem gilt die alte Schreibweise heute als *deprecated* – veraltet.

16.6 Datenbankzugriff mit Perl

Für einen Zugriff auf eine Oracle-Instanz nutzt XAMPP wieder einmal das mächtige *Database Interface* (DBI). Im *XAMPP Perl-Add-On* für Windows ist der entsprechende DBD-Treiber schon enthalten. Hier können Sie gleich loslegen.



Achtung

Im Linux-XAMPP gibt es zurzeit leider keine Unterstützung der Oracle-Datenbank unter Perl. Wenn sich das in Zukunft ändert, wird der Zugriff aber genau so funktionieren, wie er hier für Windows beschrieben wird.

Analog zu dem Beispiel in PHP kontaktieren Sie die lokale Oracle-Datenbank als Benutzer *hr* mit dem Passwort *geheim*.

Listing 16.2: Zugriff von Perl auf Oracle

```
#!/opt/lampp/bin/perl
use DBI;

print "Content-Type: text/html\n\n";

my $dsn="dbi:Oracle:host=localhost;sid=xe";
my $dbh=DBI->connect($dsn,"hr","geheim") or
    die "Kann Datenbank nicht erreichen!";

print "<html>";
print "<head>";
print "<title>Perl und Oracle</title>";
print "</head>";
print "<body>";
print "<h1>Perl und Oracle</h1>";
print "<table border=\"1\">";
print "<tr>";
print "<th>Interpret</th>";
print "<th>Titel</th>";
print "<th>Jahr</th>";
print "</tr>";

my $query="SELECT * FROM cds";

my $prep_sql=$dbh->prepare($query) or die print "Can't prepare";
$prep_sql->execute() or die print "Can't execute";

while (my @row = $prep_sql->fetchrow_array())
{
    print "<tr>";
    print "<td>".$row[2]."</td>";
    print "<td>".$row[1]."</td>";
    print "<td>".$row[3]."</td>";
    print "</tr>";
}

$prep_sql->finish();
$dbh->disconnect();

print "</table>";
print "</body>";
print "</html>";
```

16.7 Typische Fehlermeldungen und ihre Ursachen

Die von Oracle produzierten Fehlermeldungen sind oft nicht direkt verständlich. Aus diesem Grund folgt hier eine kleine Liste typischer Fehlermeldungen einschließlich Erläuterung.

- `ORA-12560: TNS:protocol adapter error`

`ORA-12170: TNS:Connect timeout occurred`

PHP konnte die Oracle-Datenbank nicht erreichen, weil ein Netzwerkproblem vorliegt. Entweder ist tatsächlich etwas mit der Netzwerkhardware nicht in Ordnung oder irgendwo funkt eine Firewall dazwischen. Dieser Fehler kann aber auch auftreten, wenn eine falsche IP-Adresse im Connect Descriptor angegeben wurde.

- `ORA-12541: TNS:no listener`

Der Oracle-Server läuft nicht. Zumindest nicht auf dem Rechner, der im Connect Descriptor angegeben wurde. In Unix übersetzt heißt dieser Fehler: *Connection refused*.

- `ORA-12514: TNS:listener does not currently know of service requested in connect descriptor`

Der im Connect Descriptor angegebene Datenbankname (SID) ist falsch. Im Connect Descriptor wird er bei `SERVICE_NAME=` angegeben. Benutzt man die kurze Easy Connect-Syntax, dann steht diese Angabe hinter dem letzten Slash.



17 Firebird

Firebird ist die Open-Source-Variante der Interbase-Datenbank von Borland. (Wir wissen, dass es *Datenbankmanagementsystem* heißen müsste. Zur besseren Lesbarkeit verwenden wir hier aber den umgangssprachlichen Begriff »Datenbank«.) Firebird entstand im Jahr 2000, als Borland Interbase für kurze Zeit unter einer Open-Source-Lizenz veröffentlichte. Borland machte diesen Schritt schnell rückgängig und führte Interbase als Closed-Source-Produkt weiter. Die Open-Source-Variante Firebird war aber geboren und hat bis heute überlebt.

17.1 Installation

17.1.1 Windows

Der Firebird Server ist nicht integraler Bestandteil von XAMPP für Windows. Es ist also notwendig, dass Sie sich ihn separat unter der URL <http://firebird.sourceforge.net> besorgen. Der Firebird Datenbank Server für Windows liegt als ZIP und mit Installationssystem vor. Beim Setup der Installer-Version werden automatisch die Dienste *Firebird Server* und *Firebird Guardian* initiiert.

Allerdings ist auch für die Installer-Version von Firebird eine kleine, aber notwendige Anpassung erforderlich. Ansonsten werden Sie später nicht mit PHP auf die Firebird-Datenbank zugreifen können:

Öffnen Sie die Datei `C:\WINDOWS\system32\drivers\etc\services` in Ihrem Lieblingseditor und fügen Sie die folgende Zeile dieser Datei hinzu:

```
gds_db      3050/tcp      #Interbase/Firebird
```

Der fehlende Eintrag in der Datei `services` ist ein häufiger Fehler, der einen Zugriff über TCP/IP auf den Firebird unmöglich macht.

Damit besitzen Sie schon eine gute Basis für den Zugriff auf den Firebird z. B. mit den Sprachen PHP, Perl und Java. Doch zum Schluss noch ein paar Hinweise. In Firebird ist der Systemadministrator *SYSDBA* mit dem Passwort *masterkey* bereits voreingestellt. Um das Passwort zu ändern, benutzen Sie die Datei `gsec.exe` im `bin`-Ordner des Firebird-Stammverzeichnisses.

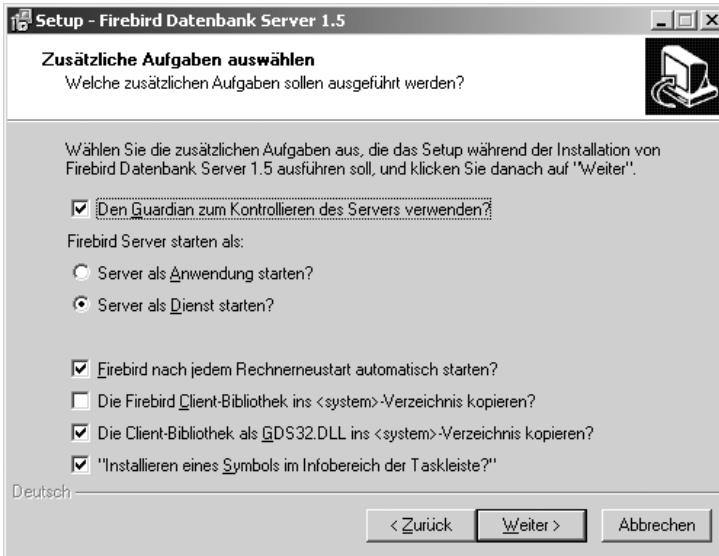


Abbildung 17.1: Die Firebird-Installation über den Setup-Assistenten



Abbildung 17.2: Die angepasste Datei services im Editor

```
C:\Programme\Firebird\Firebird_1_5\bin\gsec -user sysdba -password masterkey ←
-modify sysdba -pw geheim
```

Fertig! Firebird ist nun installiert, gestartet und einsatzbereit.

17.1.2 Linux

Die Linux-Version von XAMPP unterstützt bisher lediglich Firebird 1.0 und nicht die aktuelle Version 1.5.

1. Laden Sie bitte Firebird von der Firebird-Homepage <http://firebird.sourceforge.net/> herunter. Für die folgenden Schritte haben wir den Firebird Super Server in der Version 1.0.3.972 als tar.gz-Datei verwendet.
2. Entpacken Sie das heruntergeladene Archiv:

```
tar xvfz FirebirdSS-1.0.3.972-0.tar.gz
```

3. Für den nächsten Schritt müssen Sie *root*-Rechte besitzen. Melden Sie sich als *root* an – zum Beispiel mit dem *su*-Befehl – und wechseln Sie in das beim Entpacken des Archivs entstandene Verzeichnis:

```
cd FirebirdSS-1.0.3.972-0/
```

4. Hier rufen Sie den Firebird-Installer auf:

```
./install.sh
```

Auf dem Bildschirm erscheinen nun folgende Zeilen:

```
Firebird Classic 1.0.3.972-0 Installation
```

```
Press Enter to start installation or ^C to abort
```

Bestätigen Sie den Installationswunsch, indem Sie auf die Enter-Taste drücken. Jetzt beginnt die eigentliche Installation von Firebird.

```
Extracting install data
```

```
...
```

```
...
```

```
Install completed
```

Eventuell angezeigte Warnungen können Sie in der Regel ignorieren.

5. Nach der Installation können Sie den Firebird Server starten:

```
/etc/init.d/firebird start
```

Wenn Sie wünschen, dass Firebird auch nach dem nächsten Neustart Ihres Systems mit gestartet wird, dann sollten Sie dies nun mit dem Runlevel Manager Ihrer Linux-Distribution einstellen.

6. Zum Schluss sollten Sie noch das Passwort für den Datenbankadministrator *SYS-DBA* setzen, da es in der Standardinstallation von Firebird auf *masterkey* voreingestellt ist.

```
/opt/interbase/bin/gsec -user sysdba -password masterkey -modify sysdba -pw geheim
```

Fertig! Firebird ist nun installiert, gestartet und einsatzbereit.

17.2 Beispielbenutzer und -datenbank einrichten

Auch für dieses Beispiel kommt wieder die CD-Verwaltungsdemo zum Einsatz. Nachdem Sie einen neuen Benutzer für die Datenbank eingerichtet haben, können Sie mit dem *isql*-Tool die Datenbank *cdcol* und die *cds*-Tabelle anlegen und mit Inhalten füllen.

1. Richten Sie mit dem `gsec`-Befehl den Benutzer *oswald* mit dem Passwort *geheim* ein.
Unter Windows rufen Sie dazu auf:

```
C:\Programme\Firebird\Firebird_1_5\bin\gsec -user sysdba -password geheim -add oswald -pw geheim
```

Unter Linux verwenden Sie den folgenden Aufruf:

```
/opt/interbase/bin/gsec -user sysdba -password geheim -add oswald -pw geheim
```

2. Erstellen Sie nun ein Verzeichnis, in dem Sie später die Datenbank-Dateien ablegen werden. Wo dieses Verzeichnis genau liegt und wie es heißt, ist Ihnen überlassen.

Unter Windows:

```
mkdir C:\Programme\Firebird\Firebird_1_5\db
```

Unter Linux:

```
mkdir /opt/lampp/var/firebird
```

Hinweis



Firebird 1.0 läuft standardmäßig unter der Benutzerkennung *root*. Wenn Sie dies geändert haben, sollten Sie unbedingt darauf achten, dass dieses Verzeichnis auch für den Firebird-Benutzer schreibbar ist. Am einfachsten erreichen Sie das, indem Sie dieses Verzeichnis dem Firebird-Benutzer überschreiben:

```
chown firebird /opt/lampp/var/firebird
```

3. Rufen Sie den `isql`-Befehl auf, um eine Datenbank anzulegen und diese mit Inhalten zu füllen.

Unter Windows:

```
C:\Programme\Firebird\Firebird_1_5\bin\isql
```

Unter Linux:

```
/opt/interbase/bin/isql
```

Sie werden von dem *isql*-Tool begrüßt und es erscheint der SQL-Prompt:

```
Use CONNECT or CREATE DATABASE to specify a database
SQL>
```

4. Erstellen Sie eine neue Datenbank. Verwenden Sie dazu die Benutzerinformationen des neu eingerichteten Benutzers:

Unter Windows:

```
CREATE DATABASE "C:\Programme\Firebird\Firebird_1_5\db\cdcol.gdb" USER 'oswald'
PASSWORD 'geheim';
```

Unter Linux:

```
CREATE DATABASE "/opt/lampp/var/firebird/cdcol.gdb" USER 'oswald' PASSWORD
'geheim';
```

Nach diesem Befehl verlassen Sie das *isql*-Programm mit folgendem Befehl:

```
QUIT;
```

5. Rufen Sie erneut den *isql*-Befehl auf und verbinden Sie sich mit dem neuen Benutzer mit der neuen Datenbank:

Unter Windows:

```
C:\Programme\Firebird\Firebird_1_5\bin\isql
C:\Programme\Firebird\Firebird_1_5\db\cdcol.gdb -u oswald -p geheim
```

Unter Linux:

```
/opt/interbase/bin/isql /opt/lampp/var/firebird/cdcol.gdb -u oswald -p geheim
```

6. Füllen Sie die Datenbank mit einer neuen Tabelle und deren Inhalten:

```
CREATE TABLE cds (id INT NOT NULL, titel VARCHAR(200), interpret VARCHAR(200),
jahr INT, PRIMARY KEY (id));
INSERT INTO cds VALUES (1,'Beauty','Ryuichi Sakamoto',1990);
INSERT INTO cds VALUES (2,'Goodbye Country (Hello Nightclub)','Groovy
e Armada',2001);
INSERT INTO cds VALUES (3,'Glee','Bran Van 3000',1997);
```

7. Schließen Sie die Eingabe der neuen Inhalte mit den beiden folgenden Befehlen ab:

```
COMMIT;
QUIT;
```

Das war's. Die Beispieldatenbank ist nun eingerichtet und mit Beispieldaten gefüllt. Im nächsten Abschnitt geht es um den Zugriff auf diese Daten.

17.3 Datenbankzugriff mit PHP

17.3.1 Interbase-Erweiterung aktivieren

Der Zugriff auf Firebird oder Interbase erfolgt unter Windows über das PHP-Modul `php_interbase.dll` und unter Linux über das Modul `interbase.so`. Beide sind allerdings in XAMPP nicht standardmäßig aktiviert und müssen zunächst von Ihnen eingeschaltet werden.

1. Öffnen Sie die Datei `<xampp>\apache\bin\php.ini` unter Windows oder die Datei `/opt/lampp/etc/php.ini`, wenn Sie Linux verwenden.
2. Suchen Sie die folgende Zeile, wenn Sie unter Windows arbeiten:

```
;extension=php_interbase.dll
```

Oder im Fall von Linux:

```
;extension="interbase.so"
```

Entfernen Sie das Semikolon am Anfang der Zeile.

Achtung



Die Firebird/Interbase-Unterstützung in XAMPP für Linux geht davon aus, dass Interbase oder Firebird unter `/opt/interbase/` installiert ist.

3. Für die Aktivierung des neuen Moduls starten Sie nun den Apache-Server neu. Unter der URL `http://localhost/xampp/phpinfo.php` können Sie kontrollieren, ob die Interbase-Erweiterung verfügbar ist.

17.3.2 Das Beispiel

Danach folgt endlich das Beispiel, das Sie auch in XAMPP als `interbase.php` im `contrib`-Verzeichnis wiederfinden:

Listing 17.1: Die `interbase.php` von XAMPP für Linux als Beispiel für den Zugriff von PHP auf Interbase/Firebird

```
<html>
<head>
<title>PHP und Interbase</title>
```

```

</head>
<body>
<h1>PHP und Interbase</h1>
<table border="1">
<tr>
    <th>Interpret</th>
    <th>Titel</th>
    <th>Jahr</th>
</tr>
<?php
    $db = ibase_pconnect(
        "localhost:/opt/lampp/var/firebird/cdcol.gdb",
        "oswald", "geheim");

    $query = "SELECT * FROM cds";
    $result = ibase_query($query);
    if ($result) {
        while ($row = ibase_fetch_assoc($result))
        {
            echo "<tr>";
            echo "<td>".$row['INTERPRET']. "</td>";
            echo "<td>".$row['TITEL']. "</td>";
            echo "<td>".$row['JAHR']. "</td>";
            echo "</tr>";
        }
    }
    else
    {
        echo ibase_errmsg();
    }

?>
</table>
</body>
</html>

```

Die Windows-Version dieser Datei unterscheidet sich lediglich im Verbindungsaufbau:

```

$db = ibase_pconnect(
    "localhost:C:/Programme/Firebird/Firebird_1_5/db/cdcol.gdb",
    "oswald", "geheim");

```

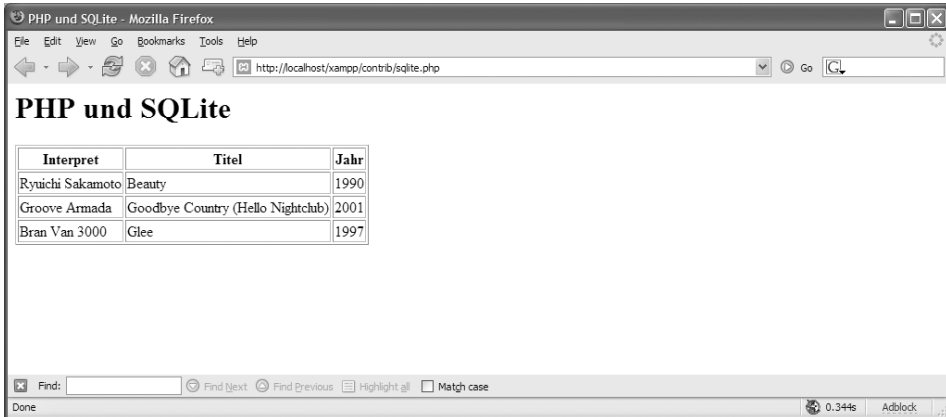


Abbildung 17.3: Es funktioniert: der Zugriff von PHP auf die Firebird-Datenbank

17.4 Datenbankzugriff mit Perl

Für die Verbindung mit Firebird unter Perl bietet sich wieder einmal ein entsprechender *DBD*-Treiber an. Tatsächlich existiert ein *DBD::Interbase*-Modul, erhältlich unter der URL <http://dbi-interbase.sourceforge.net>. Im *XAMPP Perl-Add-On* für Windows gehört der *DBD::Interbase*-Treiber für DBI allerdings nicht zur Standardausstattung. Denn leider besitzt das Paket doch so manche Tücken bei der Einbindung auf Windows-Systemen.

Unter Linux funktioniert der Interbase-DBD-Treiber allerdings wunderbar.

17.4.1 Das Beispiel unter Windows

Die hier vorgestellte Methode beruht auf dem *DBD::ODBC* Treiber, der im *XAMPP Perl-Add-On* enthalten ist. Sie benötigen also nur den *ODBC-Treiber* von der Firebird-Webseite: <http://firebird.sourceforge.net>.

1. Installieren Sie den ODBC-Treiber (*Windows Full Install*) und richten Sie danach mit dem Firebird/Interbase-Treiber eine neue *System-DSN* (*Data Source Name*) beispielsweise mit dem Namen *firebirdexample* ein.
In diesem Test benutzen wir die Beispieldatenbank *employee.fdb* und greifen mit dem *SYSDBA*-Administrationsbenutzer darauf zu.
2. Legen Sie das neue Perl-Programm mit der ODBC-Verbindung in das Verzeichnis `<xampp>\htdocs\modperl` ab.

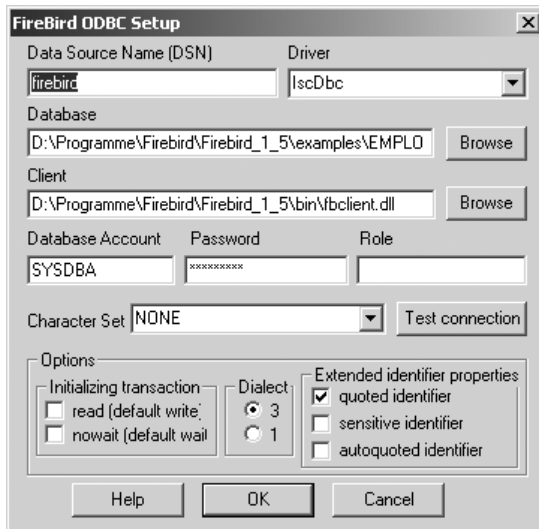


Abbildung 17.4: Konfiguration von Firebird mit DBI und ODBC

3. Rufen Sie dieses Perl-Programm über die URL <http://localhost/modperl/firebird-odbc.pl> auf.

Listing 17.2: Die `firebird-odbc.pl` für Windows

```
use DBI;
print "Content-Type: text/html\n\n";
my $user = 'SYSDBA';
my $pass = 'masterkey';
my $dsn = 'dbi:ODBC:firebirdexample';
my $dbh = DBI->connect($dsn, $user, $pass);
# SELECT
my $sql = "SELECT * from country";
my $sth = $dbh->prepare($sql)
or die "Can't prepare statement: $DBI::errstr";
$sth->execute();
while (@row = $sth->fetchrow_array())
{
    print "Land: ".$row[0]."<br>";
    print "Währung: ".$row[1]."<br><br>";
}
$dbh->disconnect();
```

17.4.2 Das Beispiel unter Linux

Unter Linux existiert die Interbase/Firebird-Unterstützung in Perl erst seit XAMPP 1.5.2. Wenn Sie eine ältere Version besitzen, können Sie versuchen, den Interbase-Treiber nachzuinstallieren:

```
/opt/lampp/bin/cpan
cpan shell -- CPAN exploration and modules installation (v1.7601)
cpan> install DBD::Interbase
```

Sie benötigen dazu allerdings auf dem gleichen System eine vollständige Interbase- oder Firebird-Installation.

Im folgenden Listing finden Sie beispielhaft den Zugriff auf die `cdcol`-Datenbank. Wenn Sie dieses Listing mit den Perl-Listings aus den anderen Datenbank-Kapiteln vergleichen, werden Sie eine recht große Ähnlichkeit feststellen können.

Es ist quasi immer wieder das gleiche Skript und der einzige Unterschied ist in der Variablen `$dsn` zu finden: Das ist eben der große Vorteil, wenn man eine Datenbankabstraktionsschicht wie DBI verwendet.

Listing 17.3: CD-Verwaltung mit Perl DBD::Interbase

```
#!/opt/lampp/bin/perl
use DBI;

print "Content-Type: text/html\n\n";

my $dsn="DBI:Interbase:db=localhost:/opt/lampp/var/firebird/cdcol.gdb";
my $dbh=DBI->connect("$dsn","oswald","geheim") or
    die "Kann Datenbank nicht erreichen!";

print "<html>";
print "<head>";
print "<title>Perl und Interbase/Firebird</title>";
print "</head>";
print "<body>";
print "<h1>Perl und Interbase/Firebird</h1>";
print "<table border='1'>";
print "<tr>";
print "<th>Interpret</th>";
print "<th>Titel</th>";
print "<th>Jahr</th>";
print "</tr>";

my $query="SELECT * FROM cds";

my $prep_sql=$dbh->prepare($query) or die print "Can't prepare";
$prep_sql->execute() or die print "Can't execute";
```

```

while (my @row = $prep_sql->fetchrow_array())
{
    print "<tr>";
    print "<td>".$row[2]."</td>";
    print "<td>".$row[1]."</td>";
    print "<td>".$row[3]."</td>";
    print "</tr>";
}

$prep_sql->finish();
$dbh->disconnect();

print "</table>";
print "</body>";
print "</html>";

```

Auch hier gilt wieder: Verwenden Sie anstelle von *oswald* und *geheim* den Benutzer und das Passwort, die Sie bei der Einrichtung der Datenbank eingesetzt haben.



Abbildung 17.5: Perl und Firebird in Aktion

17.5 Tools

Eine Übersicht über die zahlreichen freien oder auch kommerziellen Programme, die Ihr Leben mit dem Firebird vereinfachen oder verbessern sollen, erhalten Sie im Download-Bereich der Webseite <http://www.ibphoenix.com>. Ein Tool für Windows wäre beispielsweise Marathon, eine universelle Administrationsoberfläche für Interbase/Firebird. Die Entscheidung für oder gegen ein bestimmtes Tool hängt aber letztendlich vom konkreten Einsatzgebiet ab.

17.5.1 ibWebAdmin

Das Tool *ibWebAdmin* ist ein PHP-Frontend für Firebird ähnlich wie *phpMyAdmin* für MySQL, wenngleich nicht ganz so umfangreich. Für die Installation unter Windows besorgen Sie sich das aktuelle ZIP-Paket des Programms unter der URL <http://www.ibwebadmin.net>. Extrahieren Sie das ZIP-Paket direkt in das `htdocs`-Verzeichnis in XAMPP. Es entsteht der Ordner:

```
<xampp>\htdocs\ibWebAdmin_x.y
```

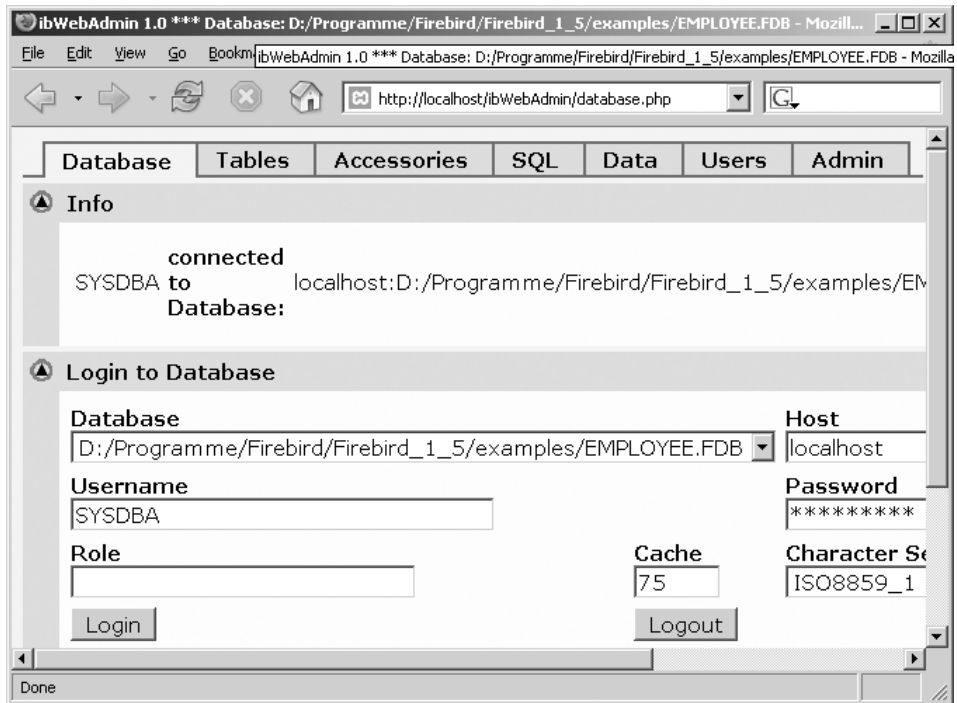


Abbildung 17.6: Das Firebird-Tool *ibWebAdmin*

Am besten benennen Sie gleich den neuen Ordner in *ibWebAdmin* um, damit Sie eine bequemere URL erhalten: <http://localhost/ibWebAdmin>.

Doch noch ist es nicht so weit. Zuvor müssen Sie die Konfiguration in der `configuration.inc.php` anpassen. Sie finden die Datei im `inc`-Ordner von *ibWebAdmin*. Hier passen Sie alle Pfade an Ihre aktuelle Windows-Firebird-Installation an. Aber Achtung: Für alle Pfadangaben wird hier der Slash anstelle des Backslashes verwendet. In diesem Beispiel ist der Firebird-Stammordner unter `D:/Programme/Firebird/Firebird_1_5/` zu finden. Ändern Sie die folgenden Werte in der `configuration.inc.php` nach diesen Schemata:

Listing 17.4: Auszug einer configuration.inc.php für Firebird (für Windows)

```

define('BINPATH', 'D:/Programme/Firebird/Firebird_1_5/bin/');
define('SECURITY_DB', 'D:/Programme/Firebird/Firebird_1_5/security.fdb');
define('TMP_PATH', 'D:/Programme/xampp/tmp/');
define('DEFAULT_USER', 'SYSDBA');
define('DEFAULT_DB', 'D:/Programme/Firebird/Firebird_1_5/examples/employee.fdb');
define('DEFAULT_PATH', 'D:/Programme/Firebird/Firebird_1_5/');
define('DEFAULT_HOST', 'localhost');
define('DEFAULT_ROLE', '');
define('DEFAULT_CACHE', 75);
define('DEFAULT_CHARSET', 'ISO8859_1');
define('DEFAULT_DIALECT', 3);
define('DEFAULT_SERVER', 'FB_1.5');
define('PROTOCOL', 'http');
$ALLOWED_DIRS = array ('D:/Programme/Firebird/Firebird_1_5/',
'D:/Programme/Firebird/Firebird_1_5/examples/', 'D:/Programme/xampp/tmp/');

```

Das methodische Vorgehen unter Linux ist selbstverständlich ähnlich. Besorgen Sie sich das aktuelle Release und extrahieren Sie es in das `\opt\lamm\htdocs`-Verzeichnis, beispielsweise mit

```
tar xvfz ibWebAdmin_1.0.tar.gz -C /opt/lamm/htdocs
```

Wechseln Sie danach die Benutzerrechte:

```
chown nobody.root -R /opt/lamm/htdocs/ibWebAdmin_1.0
```

Und abschließend passen Sie die `configuration.inc.php` im `inc`-Verzeichnis von *ibWebAdmin* mit den aktuellen Pfaden an.



18 PostgreSQL

PostgreSQL ist eine Datenbank, deren Bekanntheitsgrad bisher leider nicht sehr hoch ist. In einer Umfrage, die wir im Juni 2005 durchführten, hatte sie bei der Frage nach der Wichtigkeit von verschiedenen Datenbanken immerhin den vierten Platz belegt (hinter MySQL, Oracle, SQLite und knapp vor MS SQL) (Quelle: <http://www.apachefriends.org/files/surveys/af-umfrage-2005-uninterpretiert.pdf>).

Aufgrund dieser Umfrage entschieden wir, PostgreSQL auch in XAMPP zu unterstützen. *Unterstützen* bedeutet in diesem Fall, dass der Zugriff von XAMPP auf eine PostgreSQL-Datenbank möglich ist. Es bedeutet aber nicht, dass XAMPP selbst die PostgreSQL-Datenbanksoftware mitbringt. Diese müssen Sie selbst noch installieren. Im Hinblick auf Datenbanken ist MySQL nach wie vor unsere erste Wahl und es sieht auch nicht so aus, als ob sich das bald ändern würde.

Hinweis



Ausgesprochen wird PostgreSQL übrigens »post gres Q L« oder einfach nur »postgres«. Letztere Aussprache ist historisch bedingt, da PostgreSQL aus einer Software entstanden ist, die nur Postgres hieß.

Eine Besonderheit von PostgreSQL ist, dass es dabei nicht einfach eine relationale Datenbank, sondern eine so genannte *objektrelationale Datenbank* ist (kurz *ORDB* oder richtiger ausgedrückt: *ORDBMS*). Das bedeutet ganz knapp gesagt: In dieser Datenbank können Eigenschaften von Tabellen weitervererbt werden.

Hinweis



Auch die Lizenz, unter der PostgreSQL veröffentlicht wurde, ist besonders:

Permission to use, copy, modify, and distribute this software and its documentation for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies.

(Zitat aus der Copyright-Datei der Software)

18.1 Installation

18.1.1 Windows

Seit der Version 8 gibt es PostgreSQL als Microsoft Installer Paket (.msi), was die Installation unter Windows recht einfach gestaltet. Sie finden das Installer-Paket auf der Buch-CD-ROM oder im Internet auf der Homepage von PostgreSQL: <http://www.postgresql.org>.

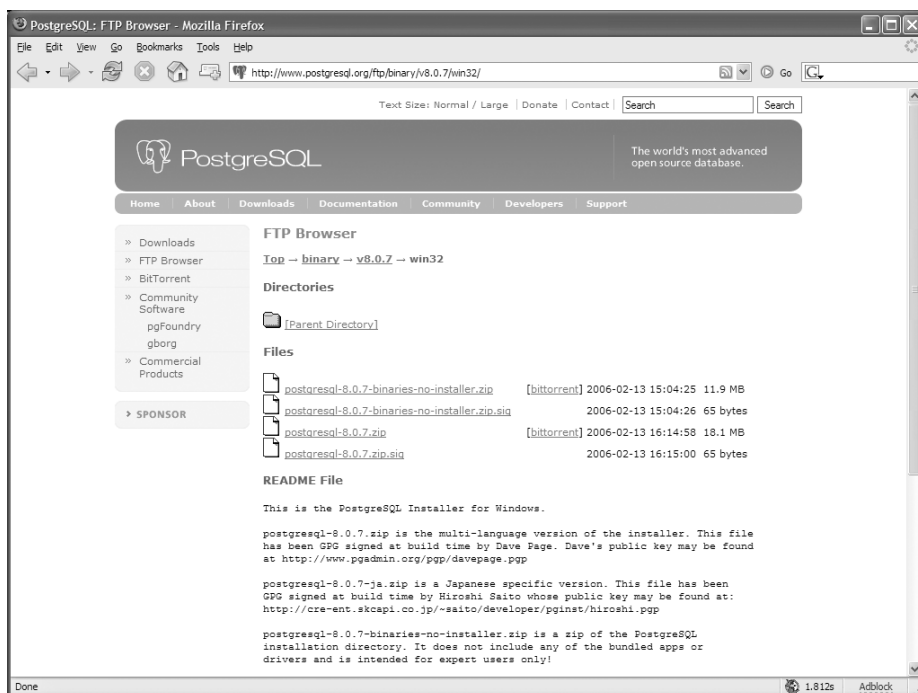


Abbildung 18.1: Laden Sie die `postgresql-8.0.7.zip`-Datei herunter: Sie enthält die Microsoft Windows-Installer-Version

Zu Beginn der Installation haben Sie die Möglichkeit, einzustellen, ob Sie PostgreSQL als Dienst laufen lassen wollen. Wenn Sie das möchten, dann wird der Installer automatisch einen neuen User *postgres* für diesen Dienst anlegen. Dabei können Sie ein Passwort für diesen Benutzer angeben. Ist dieses Passwort zu einfach, schlägt Ihnen der Installer ein schwierigeres vor: Nehmen Sie ruhig dieses schwierigere Passwort. In der Regel werden Sie es nie benötigen. Daher brauchen Sie es sich auch gar nicht erst zu merken.

Erst im folgenden Schritt werden Sie nach der Einrichtung des PostgreSQL-Administrators gefragt. Und jetzt sollten Sie sich das Passwort merken bzw. notieren, um den PostgreSQL-Server mit neuen Datenbanken und Benutzern einrichten zu können.



Abbildung 18.2: Wenn Sie über das Netzwerk auf die PostgreSQL-Datenbank zugreifen wollen, sollten Sie die Checkbox bei Adressen aktivieren. Wählen Sie unter Locale »German, Germany« und unter Encoding »LATIN1« aus. Benutzername und Kennwort sollten Sie sich unbedingt merken

Nach der Installation wird der PostgreSQL-Dienst automatisch gestartet. Wie alle anderen Dienste auch können Sie diesen unter SYSTEMSTEUERUNG > VERWALTUNG > DIENSTE später noch starten oder stoppen.

Wenn Sie die Eingabeaufforderung vorziehen, können Sie auch folgende zwei Befehle verwenden:

■ Starten des Dienstes:

```
net start "PostgreSQL Database Server 8.0"
```

■ Stoppen des Dienstes:

```
net stop "PostgreSQL Database Server 8.0"
```

Nach der Installation finden Sie im Verzeichnis `C:\Programme\PostgreSQL\8.0\bin` sämtliche Haupt- und Hilfsprogramme zu PostgreSQL. Bei den dort enthaltenen EXE-Dateien handelt es sich allesamt um Windows-Konsolenprogramme. Wer hingegen ein grafisches Administrationswerkzeug sucht, greift am besten gleich zu *pgAdmin III*. Dieses und die wichtigsten Dienstprogramme finden Sie im Startmenü unter **START > PROGRAMME > POSTGRESQL 8.0**.

18.1.2 Linux

Die Installation von PostgreSQL unter Linux ist recht einfach, da eigentlich alle Linux-Distributionen bereits PostgreSQL mitbringen und es lediglich über den Paket-Manager installiert werden muss.

Achten Sie bei der Installation darauf, auch die PostgreSQL-Client-Tools zu installieren.

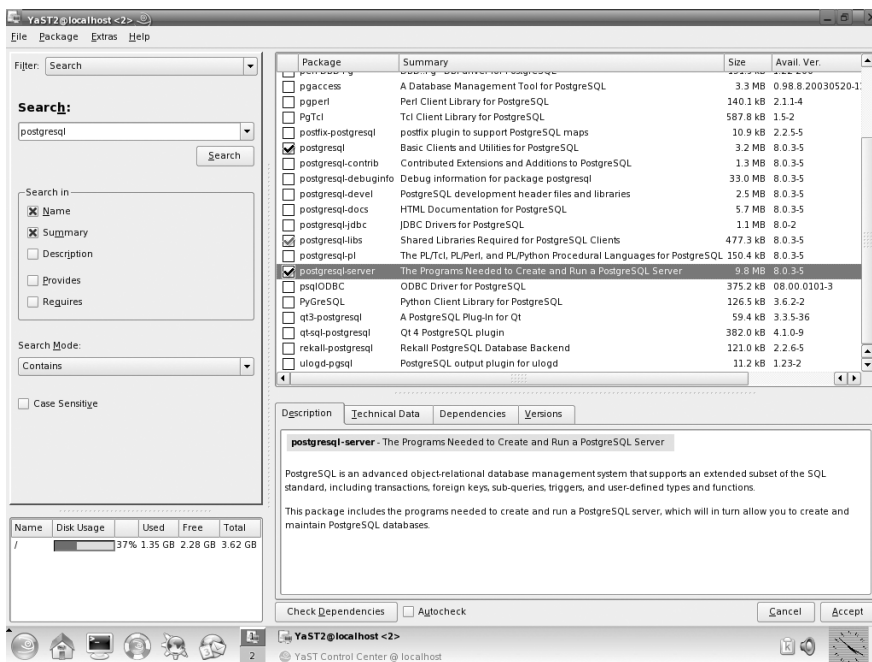


Abbildung 18.3: Die Installation von PostgreSQL unter openSUSE 10.0

Nach der Installation von PostgreSQL öffnen Sie die Datei `/var/lib/pgsql/data/pg_hba.conf` und suchen die folgende Zeile:

```
host    all             all             127.0.0.1/32      ident sameuser
```

Entfernen Sie `ident sameuser` und ersetzen Sie es durch `md5`. Die Zeile sollte nun so aussehen:

```
host    all             all             127.0.0.1/32      md5
```

Starten Sie erst nach dieser Änderung die PostgreSQL-Datenbank:

```
/etc/init.d/postgresql start
```

Sollten Sie allerdings PostgreSQL schon früher gestartet haben, dann starten Sie PostgreSQL jetzt neu, um die Änderung an der Datei `pg_hba.conf` wirksam werden zu lassen:

```
/etc/init.d/postgresql restart
```

18.2 Beispielbenutzer und -datenbank einrichten

Wie auch in den anderen Datenbank-Kapiteln haben wir das CD-Verwaltungsbeispiel an PostgreSQL angepasst. In diesem Beispiel legen Sie zunächst einen neuen Datenbankbenutzer *oswald* mit dem Passwort *geheim* an. Danach richten Sie über diesen Benutzer eine neue PostgreSQL-Datenbank *cdcol* mit der Tabelle *cds* an.

Unter Windows können Sie für diese Schritte auch sehr gut die grafische Oberfläche *pgAdmin III* verwenden.

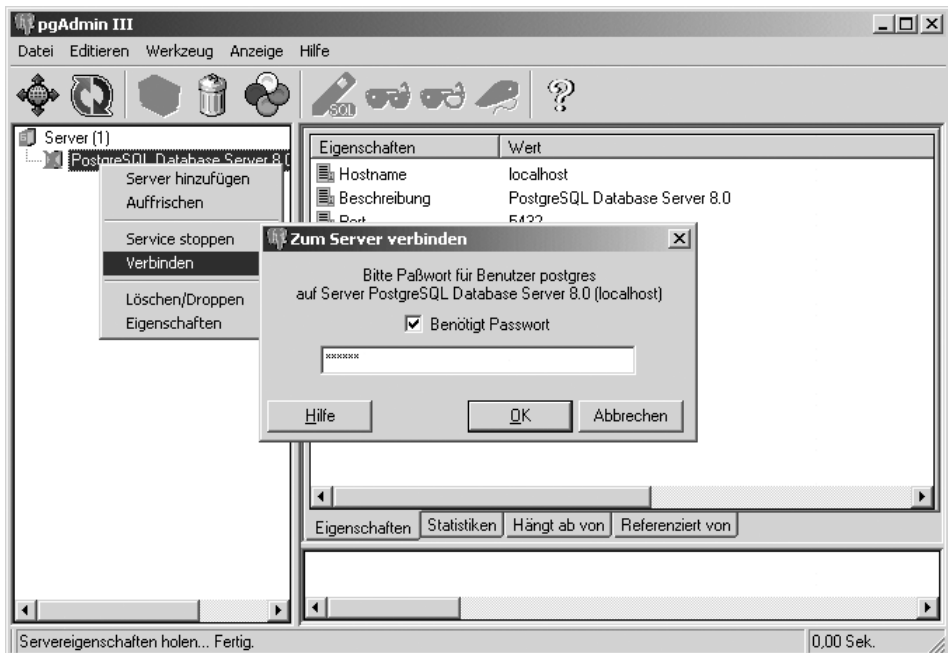


Abbildung 18.4: Der Administrator postgres meldet sich lokal auf pgAdmin III an.

Zum Einrichten von neuen Benutzern gibt es in PostgreSQL den Befehl *createuser*. Dieser Befehl funktioniert unter Linux und Windows fast identisch.

1. Wenn Sie allerdings Windows verwenden, wechseln Sie bitte in der Eingabeaufforderung als Erstes in das `bin`-Verzeichnis Ihrer PostgreSQL-Installation, ansonsten werden die benötigten Befehle nicht gefunden:

```
cd C:\Programme\PostgreSQL\8.0\bin
```

Unter Linux wechseln Sie vom Benutzer *root* zum Benutzer *postgres*:

```
su - postgres
```

2. Geben Sie nun den folgenden Befehl ein, um einen neuen PostgreSQL-Benutzer einzurichten. Achten Sie bitte auf die Großschreibung der Optionen:

```
createuser -U postgres -P
```

3. Sie werden jetzt nach dem Benutzernamen, dem Passwort und den Rechten des neuen Benutzers gefragt. Als Letztes werden Sie unter Windows nach dem Passwort des PostgreSQL-Administrators gefragt. Dieses haben Sie bei der Installation eingestellt und sich hoffentlich gut gemerkt. Unter Linux entfällt diese Passwortabfrage.

```
Enter name of user to add: oswald
Enter password for new user: *****
Enter it again: *****
Shall the new user be allowed to create databases? (y/n) y
Shall the new user be allowed to create more new users? (y/n) n
Password: ****
CREATE USER
```

4. Im nächsten Schritt erstellen Sie unter diesem Benutzer die neue Datenbank `cdcol`.

```
createdb cdcol -U oswald -h localhost
Password: *****
CREATE DATABASE
```

Wenn Sie nach dem Passwort gefragt werden, geben Sie das Passwort an, das Sie oben beim `createuser`-Befehl für den neuen Benutzer angegeben haben.

5. Richten Sie nun die Tabelle `cds` ein und füllen Sie diese mit ein paar Beispieldaten. Verwenden Sie dazu das PostgreSQL-Tool *psql*. Es ist für PostgreSQL das, was für MySQL der MySQL-Monitor ist.

```
psql cdcol -U oswald -h localhost
Willkommen bei psql 8.0.7, dem interaktiven PostgreSQL-Terminal.
```

Geben Sie ein: `\copyright` für Urheberrechtinformationen
`\h` für Hilfe über SQL-Anweisungen
`\?` für Hilfe über interne Anweisungen
`\g` oder Semikolon, um eine Anfrage auszuführen
`\q` um zu beenden

```
cdcol=# CREATE TABLE cds ( id INT, titel VARCHAR(200),
interpret VARCHAR(200), jahr INT, PRIMARY KEY (id));
HINWEIS: CREATE TABLE / PRIMARY KEY erstellt implizit einen Index »cds_pkey«
für Tabelle »cds«
CREATE TABLE
cdcol=# INSERT INTO cds VALUES (1,'Beauty','Ryui
chi Sakamoto',1990);
INSERT 17147 1
cdcol=# INSERT INTO cds VALUES (2,'Goodbye Country (
Hello Nightclub)','Groove Armada',2001);
INSERT 17148 1
cdcol=# INSERT INTO cds VALUES (3,'Glee','Bran Van 3000',1997);
INSERT 17150 1
cdcol=# \q
postgres@straylight:~>
```

Das war's. In den nächsten Abschnitten sehen Sie nun, wie Sie mittels PHP und Perl auf diese Datenbank zugreifen können.

18.3 Datenbankzugriff mit PHP

18.3.1 PostgreSQL-Erweiterung unter Linux aktivieren

Unter XAMPP für Linux ist die PostgreSQL-Unterstützung schon automatisch aktiviert.

18.3.2 PostgreSQL-Erweiterung unter Windows aktivieren

Das PHP-Modul, über das Sie sich mit Ihrem PostgreSQL-Server unter Windows verbinden, heißt `php_pgsql.dll`. Zu Beginn stellen Sie sicher, dass das Modul auch geladen ist. Suchen Sie hierfür in der Datei `<xampp>\apache\bin\php.ini` folgende Zeile:

```
extension=php_pgsql.dll
```

Sollte diese Zeile durch ein Semikolon (also `;extension=php_pgsql.dll`) deaktiviert sein, entfernen Sie dieses und starten den Apache-Server z.B. über das *XAMPP Control Panel* (`xampp-control.exe`) neu. Damit wird automatisch das PHP-Modul für den Zugriff auf PostgreSQL geladen.

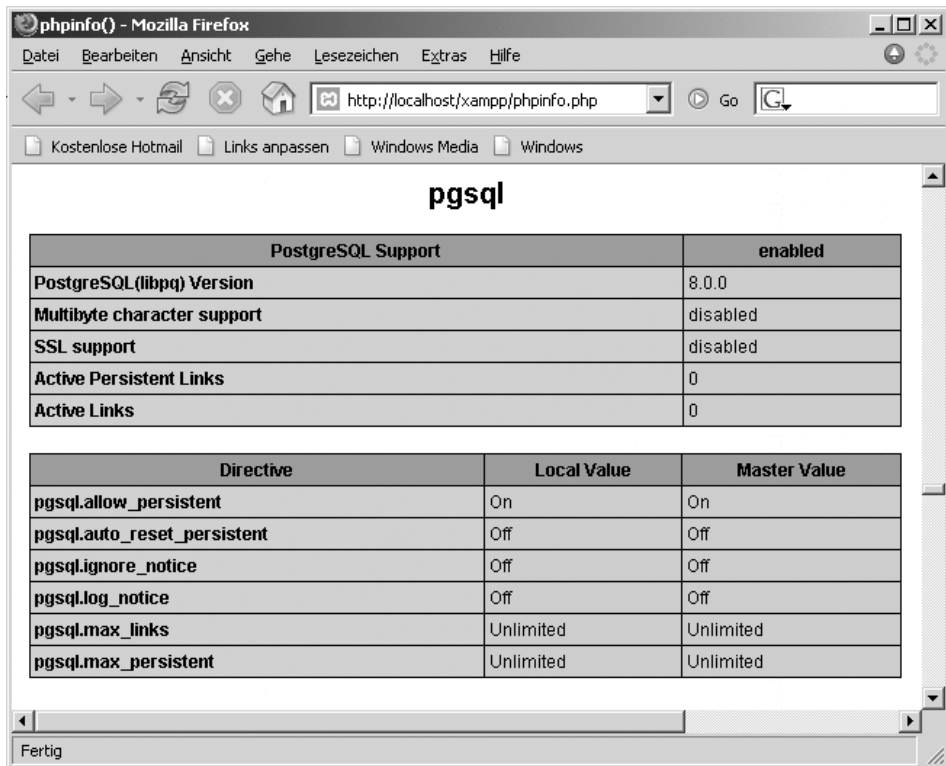


Abbildung 18.5: Die PostgreSQL-Unterstützung ist geladen.

18.3.3 Das Beispiel

Das folgende Beispielskript greift mit SELECT auf die Tabelle `cds` der Datenbank `cdcol` zu. Ab XAMPP 1.5.2 ist dieses Beispielskript Bestandteil von XAMPP. Sie finden es bei XAMPP für Windows unter `<xampp>\htdocs\contrib\postgres.php` bzw. bei XAMPP für Linux unter `/opt/lampp/htdocs/xampp/contrib/postgres.php`.

Listing 18.1: Zugriff mit PHP auf die Beispieldatenbank `cdcol`

```
<html>
<head>
<title>PHP und PostgreSQL</title>
</head>
<body>
<h1>PHP und PostgreSQL</h1>
<table border="1">
<tr>
<th>Interpret</th>
<th>Titel</th>
```

```

        <th>Jahr</th>
    </tr>
<?php

    $conn_string = "host=localhost port=5432 dbname=cdcol
                    user=oswald password=geheim";

    $db_handle = pg_connect($conn_string);
    if (!$db_handle)
        die("Kann Datenbank nicht erreichen!");

    $query = "SELECT * FROM cds";

    $result = pg_exec($db_handle, $query);
    if ($result)
    {
        for ($row = 0; $row < pg_numrows($result); $row++)
        {
            $values = pg_fetch_array($result, $row, PGSQL_ASSOC);
            echo "<tr>";
            echo "<td>".$values['interpret']. "</td>";
            echo "<td>".$values['titel']. "</td>";
            echo "<td>".$values['jahr']. "</td>";
            echo "</tr>";
        }
    }
    else
    {
        echo pg_errormessage($db_handle);
    }
?>
</table>
</body>
</html>

```

In dem Beispiel wird das Benutzer/Passwort-Paar *oswald* und *geheim* verwendet. Passen Sie dieses bitte an Ihren Benutzer und dessen Passwort an.

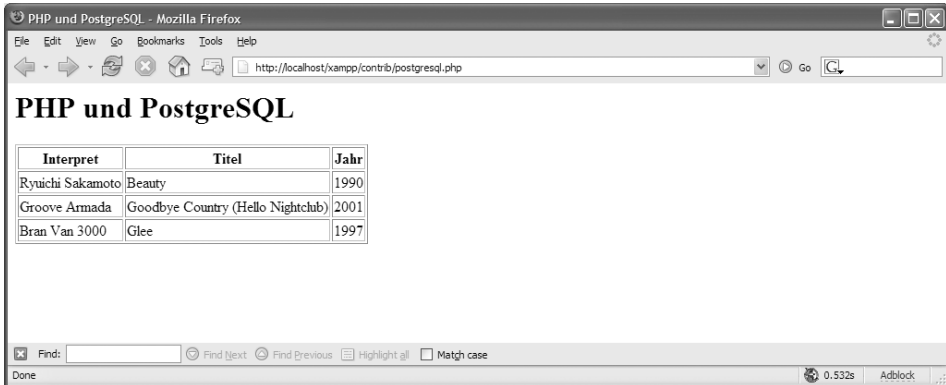


Abbildung 18.6: PostgreSQL und PHP in Aktion

18.4 Datenbankzugriff mit Perl

XAMPP verwendet den *Pure Perl PostgreSQL*-Treiber für DBI/DBD, um mit Perl auf PostgreSQL zuzugreifen. Dieser Treiber hat den großen Vorteil, dass er so ziemlich auf jeder Plattform gut und problemlos funktioniert.

Hinweis



PgPP für XAMPP für Linux 1.5.1 und früher

Unter Linux wird erst ab XAMPP 1.5.2 der Zugriff auf PostgreSQL von Perl aus unterstützt. Wenn Sie eine ältere Version von XAMPP verwenden, können Sie aber Ihr Perl auch ganz leicht über CPAN mit PgPP aufrüsten:

```
/opt/lampp/bin/cpan
cpan shell -- CPAN exploration and modules installation (v1.7601)
cpan> install DBD::PgPP
```

Der PgPP-Treiber hat den großen Vorteil, dass er komplett in Perl geschrieben wurde. D. h., er wird überall da funktionieren, wo auch Perl funktioniert, und zum Installieren benötigen Sie nicht einmal einen C-Compiler auf Ihrem System.

18.4.1 Das Beispiel

Das folgende Beispielskript greift mit SELECT auf die Tabelle `cds` der Datenbank `cdcol` zu. Ab XAMPP 1.5.2 ist dieses Beispielskript Bestandteil von XAMPP. Sie finden es bei XAMPP für Windows unter `<xampp>\htdocs\contrib\postgres.pl` bzw. bei XAMPP für Linux unter `/opt/lampp/htdocs/xampp/contrib/postgres.pl`.

Listing 18.2: CD-Verwaltung mit Perl DBD::PgPP

```
#!/opt/lampp/bin/perl
use DBI;

print "Content-Type: text/html\n\n";

my $string="dbi:PgPP:dbname=cdcol;host=localhost;port=5432";
my $dbh=DBI->connect("$string","oswald","geheim") or
    die "Kann Datenbank nicht erreichen!";

print "<html>";
print "<head>";
print "<title>Perl und PostgreSQL</title>";
print "</head>";
print "<body>";
print "<h1>Perl und PostgreSQL</h1>";
print "<table border='1'>";
print "<tr>";
print "<th>Interpret</th>";
print "<th>Titel</th>";
print "<th>Jahr</th>";
print "</tr>";

my $query="SELECT * FROM cds";

my $prep_sql=$dbh->prepare($query) or die print "Can't prepare";
$prep_sql->execute() or die print "Can't execute";

while (my @row = $prep_sql->fetchrow_array())
{
    print "<tr>";
    print "<td>".$row[2]."</td>";
    print "<td>".$row[1]."</td>";
    print "<td>".$row[3]."</td>";
    print "</tr>";
}

$prep_sql->finish();
$dbh->disconnect();

print "</table>";
print "</body>";
print "</html>";
```

Auch hier gilt wie beim PHP-Beispiel: Verwenden Sie anstelle von *oswald* und *geheim* den Benutzer und das Passwort, die Sie bei der Einrichtung der Datenbank verwendet haben.

Die Zeile am Anfang des Perl-Programms (`#!/opt/lampp/bin/perl`) hat nur unter Linux eine Bedeutung. Dort bestimmt sie, welches Perl verwendet werden soll: nämlich das Perl von XAMPP und nicht das Perl vom Linux-System.

Unter Windows ist diese Zeile bedeutungslos, da dort in der Konfiguration von `mod_perl` vordefiniert wird, dass ausschließlich das Perl von XAMPP verwendet werden soll.

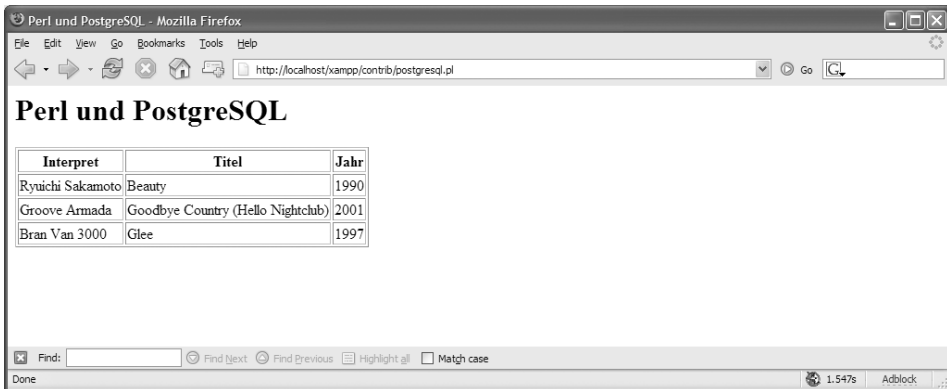


Abbildung 18.7: Perl und PostgreSQL in Aktion

18.5 Ein praktisches Werkzeug: phpPgAdmin

Wohl eines der beliebtesten Administrationstools heißt *phpPgAdmin* und ist unter der URL <http://phpPgAdmin.sourceforge.net> frei erhältlich. Wie der Name schon vermuten lässt, handelt es sich hierbei um ein webbasiertes PHP-Programm, vergleichbar mit *phpMyAdmin* für den *MySQL*-Server. Der Vorteil liegt hier wieder einmal in der Plattformunabhängigkeit.

Die Installation ist sehr einfach. Unter Windows entpacken Sie das ZIP-Archiv `phpPgAdmin-x.y.z.zip` direkt in den `htdocs`-Ordner als `[Programmpfad]\xampp\htdocs\phpPgAdmin`. Das Gleiche gilt für Linux, also:

```
tar xvfz phpPgAdmin-x.y.tar.gz -C /opt/lampp/htdocs
```

Gehen Sie danach in das Verzeichnis `phpPgAdmin\conf` (Windows) bzw. `phpPgAdmin/conf` (Linux), um die Datei `config.inc.php-dist` in `config.inc.php` umzubenennen und zu öffnen. Die `config.inc.php` ist die zentrale Konfigurationsdatei für `phpPgAdmin`. Hier passen Sie die wichtigsten Einstellungen an. Für unsere Beispielininstallation unter Windows verändern wir folgende Zeilen:

```
// Display name for the server on the login screen
$config['servers'][0]['desc'] = 'PostgreSQL';
// Hostname or IP address for server. Use '' for UNIX domain socket.
```

```

$conf['servers'][0]['host'] = 'localhost';
// Database port on server (5432 is the PostgreSQL default)
$conf['servers'][0]['port'] = 5432;
// Change the default database only if you cannot connect to template1
$conf['servers'][0]['defaultdb'] = 'template1';
// Specify the path to the database dump utilities for this server.
// You can set these to '' if no dumper is available.
$conf['servers'][0]['pg_dump_path'] = '
'C:\\Programme\\PostgreSQL\\8.0\\bin\\pg_dump.exe';
$conf['servers'][0]['pg_dumpall_path'] = '
=C:\\Programme\\PostgreSQL\\8.0\\bin\\pg_dumpall.exe';

```

Damit sich auch der »Superadministrator« *postgres* anmelden darf, ändern Sie in der `config.inc.php` noch schnell die Zeile

```
$conf['extra_login_security'] = true;
```

in

```
$conf['extra_login_security'] = false;
```

um. Ob Sie die letzte Änderung für praktisch halten oder aus Gründen der Sicherheit ablehnen, müssen Sie selbst entscheiden. Für die Demonstration in unserem Beispiel darf sich der Admin *postgres* aber anmelden. Diese hier beschriebenen Modifikationen in der Konfiguration gelten im Prinzip auch für Linux, abgesehen von den Pfaden zu den *dump*-Programmen (sog. *dumps* werden als Kopie bzw. zur Sicherung ganzer Datenbanken erstellt), die Sie für das Linux-Filesystem anpassen müssen, z.B.:

```

$conf['servers'][0]['pg_dump_path']='/usr/bin/pg_dump';
$conf['servers'][0]['pg_dumpall_path']='/usr/bin/pg_dumpall';

```

Speichern Sie die Änderungen in der Konfigurationsdatei `config.inc.php`. Das war es auch schon! Jetzt können Sie sich unter der URL <http://localhost/phpPgAdmin> anmelden.

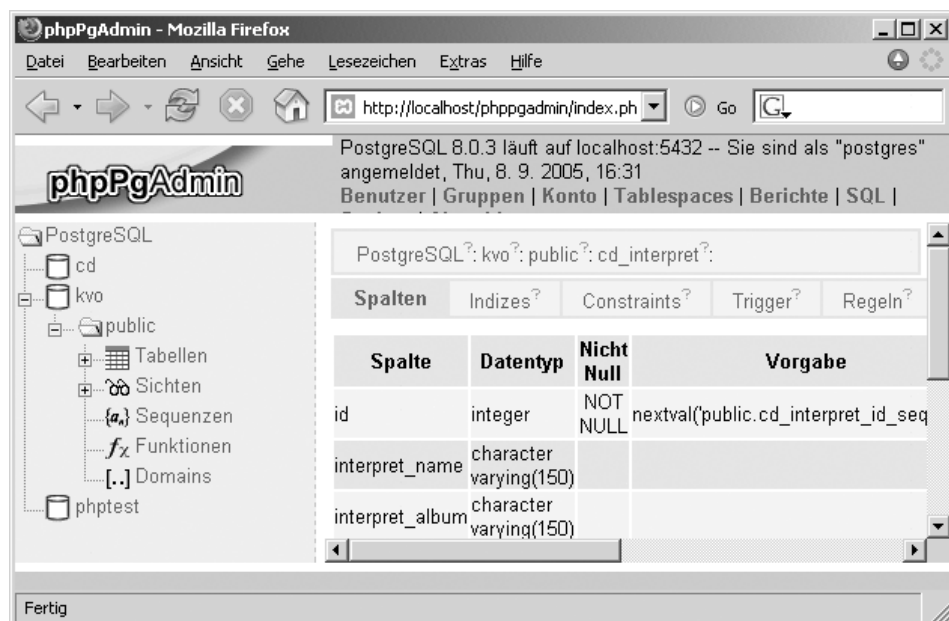


Abbildung 18.8: PostgreSQL mit phpPgAdmin

Teil IV

XAMPP erweitern und ausbauen





19 Komponenten selbst kompilieren

Wir haben XAMPP so zusammengestellt, dass es möglichst vielen Ansprüchen gerecht wird. Der eine oder andere stellt jedoch möglicherweise ganz besondere Anforderungen an das System, so dass Änderungen an den XAMPP-Komponenten unumgänglich sind.

Je nach individueller Anforderung kann es dabei notwendig sein, eine Erweiterung an einer XAMPP-Komponente vorzunehmen oder – bei sehr komplexen Wünschen – gar eine Komponente wie Apache oder MySQL komplett neu zu übersetzen. Ein weiterer Grund könnte sein, dass Sie auf eine andere Version zurückgreifen wollen, vielleicht eine ältere Version, die nicht mehr in XAMPP enthalten ist, oder eine, die noch nicht integriert ist.



Abbildung 19.1: Der zeitweise frei erhältliche C++-Compiler von Microsoft

In diesem Kapitel wird also beschrieben, wie Sie ausgewählte XAMPP-Komponenten komplett neu übersetzen können.



Tipp

Zukünftig wird – wie bereits unter Linux – auch für Windows ein *Devel-Paket* mit den Quellen sowie den Libraries der Programme erscheinen. Schauen Sie doch einfach auf der Homepage von Apache Friends nach, vielleicht ist es bereits erschienen!

19.1 Windows

Um *OpenSSL*, *MySQL* oder den *Apache* selbst zu kompilieren, benötigen Sie neben einer Perl-Installation neuerdings *MS Visual C++ 7.0* oder höher. Der *Apache 2.2.0* ließ sich beispielsweise schon nicht mehr mit *Visual C++ 6.0* übersetzen.



Tipp

Zum Zeitpunkt der Drucklegung dieses Buches hielt Microsoft den Compiler Visual C++ 2005 Express auf seinen Seiten zum Gratis-download bereit:

<http://msdn.microsoft.com/vstudio/express/visualc/>

19.1.1 OpenSSL kompilieren

In diesem Beispiel wurde *OpenSSL 0.9.8a* mit *MS Visual C++ 7.0* kompiliert. Hierfür benutzen wir den Konsolencompiler *nmake*.

1. Extrahieren Sie das Paket `openssl-0.9.8a.tar.gz`.
Hierfür können Sie beispielsweise das freie 7-zip (<http://www.7-zip.org>) verwenden. Nach dem ersten Extrahieren des `tar.gz`-Pakets entsteht das reine tar-Archiv, das erneut entpackt zu dem eigentlichen Verzeichnis `openssl-0.9.8a` mit den Quelldateien führt.
2. Wechseln Sie in das Verzeichnis `openssl-0.9.8a` und konfigurieren Sie die Quelle auf das System.

```
cd openssl-0.9.8a
perl Configure VC-WIN32
```


3. Stellen Sie die Optionen der Übersetzung ein.

```
ms\do_ms no-krb5 no-asm no-mdc2 no-rc5 bgh-nb no-idea no-ss13 VC-WIN32
```

4. Starten Sie die Kompilation mit folgendem Befehl:

```
nmake -f ms\ntdll.mak
```

Die fertig übersetzten Binärprogramme und Bibliotheken finden Sie nun im Unterordner `out32dll`.

5. Testen Sie die Kompilation. Dieser Schritt ist optional.

```
cd out32dll
..\ms\test.bat
```

```
C:\WINNT\System32\cmd.exe
D:\release\compile\openssl-0.9.8a>perl Configure UC-WIN32
Configuring for UC-WIN32
no-gmp [default] OPENSSL_NO_GMP <skip dir>
no-krb5 [krb5-flavor not specified] OPENSSL_NO_KRB5
no-mdc2 [default] OPENSSL_NO_MDC2 <skip dir>
no-rc5 [default] OPENSSL_NO_RC5 <skip dir>
no-shared [default]
no-zlib [default]
no-zlib-dynamic [default]
IsMK1MF=1
CC = cl
CFLAG = -DOPENSSL_THREADS -DDSO_WIN32
EX_LIBS =
CPUID_OBJ =
BN_ASM = bn_asm.o
DES_ENC = des_enc.o fcrypt_b.o
AES_ASM_OBJ = aes_core.o aes_cbc.o
BF_ENC = bf_enc.o
CAST_ENC = c_enc.o
RC4_ENC = rc4_enc.o
RC5_ENC = rc5_enc.o
MD5_OBJ_ASM =
SHA1_OBJ_ASM =
RMD160_OBJ_ASM =
PROCESSOR =
RANLIB = true
ARFLAGS =
PERL = perl
THIRTY_TWO_BIT mode
BN_LLONG mode
RC4_INDEX mode
RC4_CHUNK is undefined
Configured for UC-WIN32.
D:\release\compile\openssl-0.9.8a>
```

Abbildung 19.2: Über »perl Configure« OpenSSL für Windows einrichten

19.1.2 Zlib übersetzen

Das freie *Zlib* gehört zum Standard fast jeder Applikation, wenn es um das Thema Komprimierung geht. Wir wollen das Paket nach der Kompilation für das Modul `mod_deflate` im Apache nutzen.

Den Quellcode der Zlib könne Sie unter <http://www.zlib.net> herunterladen.

1. Kompilation

Extrahieren Sie das ZIP-Archiv mit dem Quellcode und öffnen Sie den VC-Arbeitsbereich.

```
\zlib123\projects\visualc6\zlib.dsw
```

Stellen Sie innerhalb von Visual VC++ unter ERSTELLEN > KONFIGURATIONSMANAGER *DLL Release* ein und erstellen Sie eine Projektmappe. Die fertigen Programm-bibliotheken finden Sie danach unter:

```
\zlib123\projects\visualc6\Win32_DLL_Release
```

2. Binärdateien für den Apache anpassen

Noch haben die binären Dateien die falschen Dateinamen. Benennen Sie alle `zlib1.*`-Dateien (wie `zlib1.dll`) in `zlib.*` (wie `zlib.dll`) um. Danach kopieren Sie alle `zlib.*`-Dateien in den Stammordner von `zlib`, in unserem Fall wäre dies das Verzeichnis `zlib123`.

19.1.3 Apache-Quellen übersetzen

Der Apache kann sowohl über *nmake* als auch über eine VC-Projektmappe (`apache.dsw`) erstellt werden. Als Sourcecode lag uns der *Apache 2.2.0* (`httpd-2.2.0-win32-src-rev2.zip`) vor, den wir mit *nmake* übersetzten.

Achtung



Eine zusätzliche Voraussetzung ist die Existenz des Programms *awk* in Ihrer *PATH*-Umgebung. Sie erhalten eine Windows-Version von *awk* beispielsweise auf den Seiten von Brian Kernighan: <http://cm.bell-labs.com/cm/cs/who/bwk/>

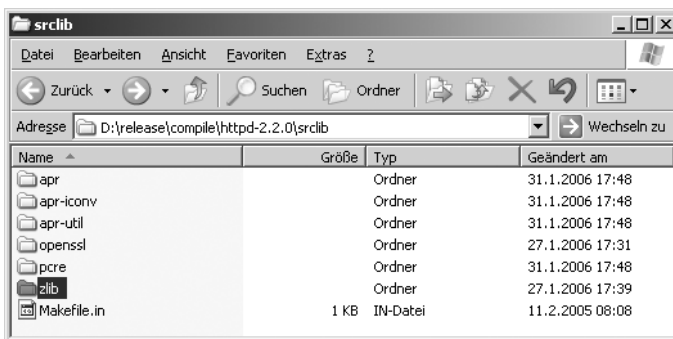


Abbildung 19.3: Die Extensions für den Apache im Ordner `src\lib`

Weitere Informationen zur Kompilierung von Apache unter Windows finden Sie unter der URL:

http://httpd.apache.org/docs/2.2/platform/win_compiling.html



Achtung

Für die Übersetzung des Quellcodes Apache 2.2.x wird MS VC 7.0 oder höher benötigt!

1. OpenSSL- und Zlib-Erweiterungen anpassen

Nachdem Sie die Quellen an einen Ort Ihrer Wahl extrahiert haben, beginnen Sie mit der Anpassung der Quellen für die Unterstützung von *OpenSSL* (*mod_ssl*) und *Zlib* (*mod_deflate*). Hierfür kopieren Sie die Verzeichnisse mit den zuvor übersetzten Quellen von OpenSSL (in diesem Beispiel den Ordner *openssl-0.9.8a*) und Zlib (in diesem Beispiel den Ordner *zlib123*) komplett in das *src/lib*-Verzeichnis von Apache. Danach benennen Sie das OpenSSL-Verzeichnis in *openssl* und das Zlib-Verzeichnis in *zlib* um:

```
\httpd-2.2.0\src\lib\openssl
\httpd-2.2.0\src\lib\zlib
```

```
C:\WINNT\System32\cmd.exe
D:\release\compile\httpd-2.2.0>nmake /f Makefile.win _apacher

Microsoft (R) Program Maintenance Utility, Version 7.10.3077
Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.

INSTDIR      = \Apache2
SERVERNAME   = localhost
PORT         = 80

To change these options use 'nmake /f Makefile.win [option=value]'
Example: nmake /f Makefile.win PORT=8080

Microsoft (R) Program Maintenance Utility, Version 7.10.3077
Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.

    echo Building Win32 Release targets (R suffixes)
Building Win32 Release targets (R suffixes)
    cd src\lib\apr
    nmake -f apr.mak                CFG="apr - Win32 Release" RECURSE=0

Microsoft (R) Program Maintenance Utility, Version 7.10.3077
Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.
```

Abbildung 19.4: Die Apache-Quelle mit nmake kompilieren

2. Jetzt können Sie den Apache kompilieren. Wechseln Sie dazu in das Verzeichnis `httpd-2.2.0`:

```
cd httpd-2.2.0
```

3. Und rufen Sie dann die beiden folgenden `nmake`-Befehle nacheinander auf:

```
nmake /f Makefile.win _apacher
nmake /f Makefile.win installr INSTDIR=c:\apache2
```

Nach erfolgreicher Übersetzung liegt das Binärpaket fix und fertig unter `c:\apache2` vor.

19.1.4 MySQL kompilieren

Für diese Übersetzung verwendeten wir das Quellpaket `mysql-5.0.18-win-src.zip` von der Website <http://www.mysql.com>.

Eine umfangreiche MySQL-Referenz finden Sie unter folgender URL:

<http://dev.mysql.com/doc/refman/5.0/en/windows-source-build.html>



Abbildung 19.5: Das MySQL-Projekt möchte nicht direkt mit VC 7.0 geladen werden!

1. Quellpaket extrahieren und `mysql.sln` mit Visual C++ öffnen

In der uns vorliegenden Version 5.0.18 war weder der Arbeitsbereich `mysql.dsw` noch die Projektdatei `mysql.dsp` kompatibel mit VC++ 7.0! Einzig das sog. *Microsoft Visual Studio Solution Object* mit dem Namen `mysql.sln` lud den gesamten Arbeitsbereich vollständig und fehlerfrei.

2. Release einstellen

Die aktive Konfiguration stellen Sie in Visual C++ unter **ERSTELLEN > KONFIGURATIONSMANAGER** auf *Release* ein. Hierfür sollte die gesamte Projektmappe MySQL ausgewählt sein.

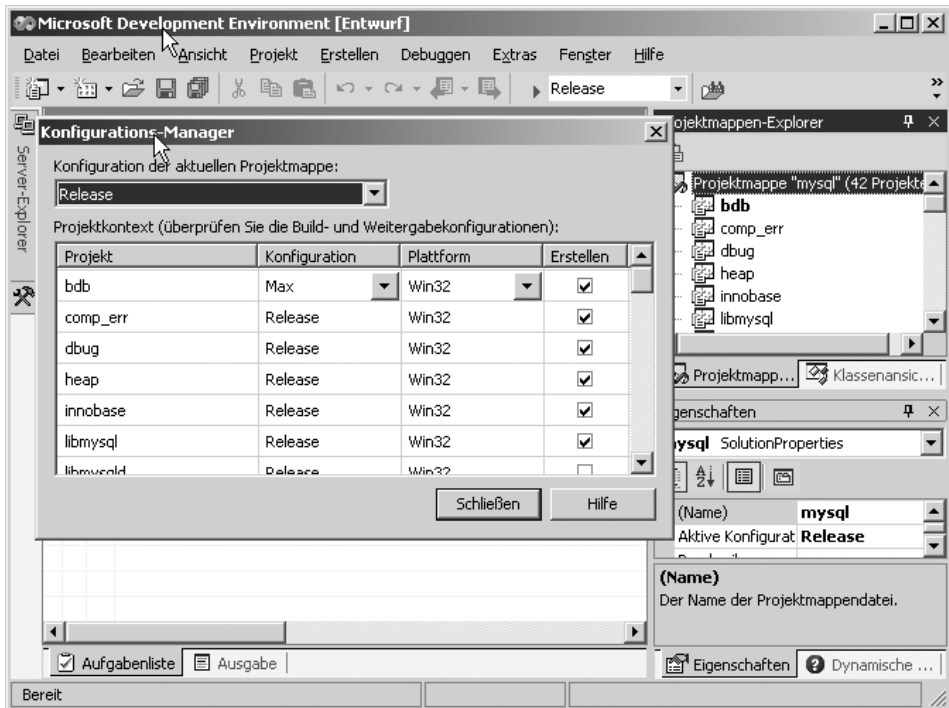


Abbildung 19.6: Die Release-Version in der `mysql.sln` auswählen

3. Kompilieren

Jetzt erstellen Sie die komplette Arbeitsmappe mit **ERSTELLEN > PROJEKTMAPPE NEU ERSTELLEN** neu. Nach der Übersetzung finden Sie alle fertigen Programmdateien im Verzeichnis wie folgt vor:

Ausführbar: `\mysql-x.x.x\client_release`

Library: `\mysql-x.x.x\lib_release`

Include: `\mysql-x.x.x\include`

4. MySQL-Paket zusammenstellen

Stellen Sie zum Schluss ein fertiges MySQL-Binärpaket zusammen.

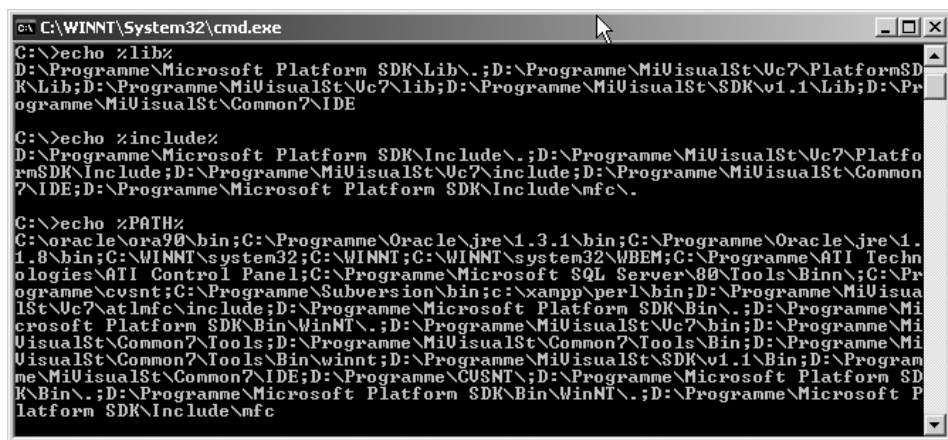
```
\mysql-x.x.x\client_release -> \mysql\bin
\mysql-x.x.x\data           -> \mysql\data
\mysql-x.x.x\share          -> \mysql\share
\mysql-x.x.x\scripts        -> \mysql\scripts
```

Um MySQL erfolgreich zu starten, benötigen Sie noch eine gültige Konfigurationsdatei (standardmäßig `my.cnf`). Sie können sich dabei von der Konfiguration der `my.cnf` in XAMPP inspirieren lassen:

```
<xampp>\mysql\bin\my.cnf
```

19.1.5 PHP übersetzen

Um PHP kompilieren zu können, benötigen Sie neben der *Visual C++* ab Version 7 auch so manche *Includes* von dem Entwicklerpaket *Microsoft Platform SDK*. Zusätzlich brauchen Sie selbstverständlich den *PHP 5-Quellcode* sowie die Pakete *bindlib_w32.zip* und *win32build.zip* von <http://www.php.net>. Von Vorteil wäre auch der Bestand der *GNU Tools* (<http://unxutils.sourceforge.net>) für Windows in Ihrer *Path-Umgebung*. Überprüfen Sie zudem bereits im Vorfeld, ob Ihre Umgebungsvariablen `%PATH%`, `%include%` und `%lib%` Ihres Arbeitsumfelds (VC ++ und MS SDK) korrekt eingerichtet sind.



```

C:\WINNT\System32\cmd.exe
C:\>echo %lib%
D:\Programme\Microsoft Platform SDK\Lib\.;D:\Programme\MiVisualSt\Uc7\PlatformSD
K\Lib;D:\Programme\MiVisualSt\Uc7\lib;D:\Programme\MiVisualSt\SDK\vl.1\Lib;D:\Pr
ogramme\MiVisualSt\Common7\IDE

C:\>echo %include%
D:\Programme\Microsoft Platform SDK\Include\.;D:\Programme\MiVisualSt\Uc7\Platfo
rmsDK\Include;D:\Programme\MiVisualSt\Uc7\include;D:\Programme\MiVisualSt\Comm
on7\IDE;D:\Programme\Microsoft Platform SDK\Include\mfc\

C:\>echo %PATH%
C:\oracle\ora90\bin;C:\Programme\Oracle\jre\1.3.1\bin;C:\Programme\Oracle\jre\1.
1.9\bin;C:\WINNT\system32;C:\WINNT;C:\WINNT\system32\WBEM;C:\Programme\ATI Techn
ologies\ATI Control Panel;C:\Programme\Microsoft SQL Server\80\Tools\Binn\;C:\Pr
ogramme\cusnt;C:\Programme\Subversion\bin;c:\xampp\perl\bin;D:\Programme\MiVisua
lSt\Uc7\atlmfc\include;D:\Programme\Microsoft Platform SDK\Bin\.;D:\Programme\Mi
crosoft Platform SDK\Bin\WinNT\.;D:\Programme\MiVisualSt\Uc7\bin;D:\Programme\Mi
VisualSt\Common7\Tools\Bin\winnt;D:\Programme\MiVisualSt\SDK\vl.1\Bin;D:\Program
me\MiVisualSt\Common7\IDE;D:\Programme\CUSNT\.;D:\Programme\Microsoft Platform SD
K\Bin\.;D:\Programme\Microsoft Platform SDK\Bin\WinNT\.;D:\Programme\Microsoft P
latform SDK\Include\mfc
  
```

Abbildung 19.7: Die Überprüfung der Umgebungsvariablen auf der Kommandozeile

Tipp



PHP unter Windows zu kompilieren ist nicht einfach. Denn meistens fehlt irgendetwas. Das eine Modul benötigt eine zusätzliche Library. Ein anderes Modul fordert die Quellen eines weiteren Programms. Damit ist das hier geschilderte Muster nur ein einführender Leitfaden, aber eben kein Ticket für garantierten Erfolg. Haben Sie aber schließlich eine Umgebung erstellt, mit der auch PHP glücklich ist, dann halten Sie sie am besten in Ehren!

Weitere Informationen rund um PHP finden Sie unter:

- PHP: <http://www.php.net/manual/de/install.windows.building.php>
- bindlib_w32: http://www.php.net/extra/bindlib_w32.zip
- win32build: <http://www.php.net/extra/win32build.zip>

In diesem Beispiel soll PHP 5 via `nmake` auf der Konsole mit MySQL-Unterstützung und dem Apache2-Modul kompiliert werden. Wir verwendeten hierfür *MySQL* in der Version 5.0.19, den *Apache 2.2.0* sowie das Quellpaket `php-5.1.2.tar.gz`.

1. Ein Arbeitsverzeichnis für PHP 5 erstellen

Erstellen Sie wie gewohnt ein Verzeichnis für PHP 5:

```
c:\work
```

2. Die Pakete *PHP-5.1.2*, *win32build* und *bindlib_w32* in das Arbeitsverzeichnis extrahieren

Nachdem Sie die Pakete extrahiert haben, sollte Ihre Verzeichnisstruktur mit dem folgenden Aufbau vergleichbar sein:

```
c:\work\bindlib_w32
c:\work\php-5.1.2
c:\work\win32build
```

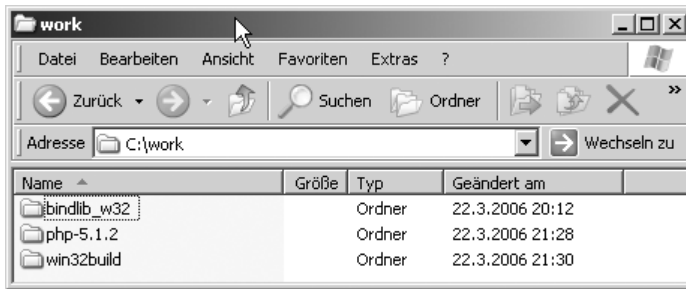


Abbildung 19.8: Der Inhalt des *work*-Verzeichnisses

3. Erstellen einer neuen `resolv.lib` über den Arbeitsbereich `bindlib.dsw` (`C:\work\bindlib_w32`)

Sie erzeugen `resolv.lib` ganz normal über die GUI Ihrer VC-Entwicklungsumgebung z.B. mit **PROJEKTMAPPE NEU ERSTELLEN**. Sie benötigen in jedem Fall eine neue `resolv.lib`, um alle Bindungen auf Ihrem System neu zu erfassen. Danach kopieren Sie den gesamten Inhalt von `C:\work\bindlib_w32\Release` nach `C:\work\win32build\lib`.

```
cp C:\work\bindlib_w32\Release\* C:\work\win32build\lib
```

4. Weitere notwendige GnuWin32-Bibliotheken hinzufügen

Leider verfügt der PHP-Quellcode noch nicht über alle für eine Übersetzung nötigen Includes und Libraries. Häufig benötigen Sie noch die folgenden Pakete:

- *libiconv*
- *Jpeg*
- *libxml*
- *freetype*

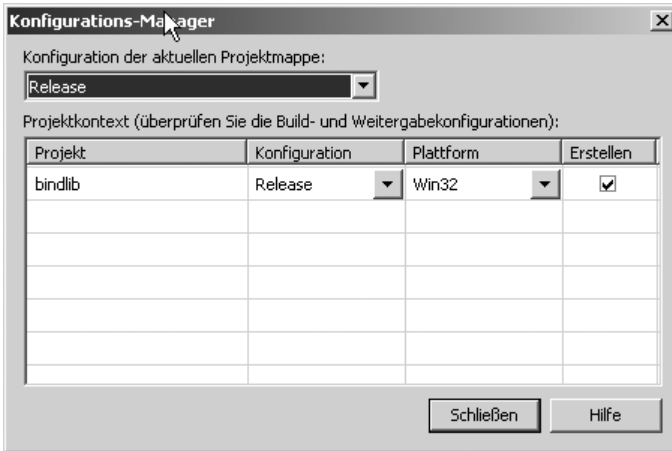


Abbildung 19.9: Die BINDLIB als Release in der VC++-Oberfläche

Um nicht alle zusätzlichen Pakete selbst kompilieren zu müssen, können Sie alternativ die doch sehr aktuellen Übersetzungen auf der Seite von <http://sourceforge.net/projects/gnuwin32/> verwenden. Wichtig sind hier die Lib- sowie Include-Ordner, die Sie für jedes Paket erneut unter `C:\work\win32build\` bereitstellen. Ein Beispiel für das `iconv`-Paket:

```
cp C:\release\compile\libiconv-1.9.1\lib\* C:\work\win32build\lib
cp C:\release\compile\libiconv-1.9.1\bin\*.dll C:\work\win32build\lib\dlls
cp C:\release\compile\libiconv-1.9.1\includes\* C:\work\win32build\includes
```

5. MySQL-Unterstützung vorbereiten

Auch hier benötigen Sie die Include- und Lib-Dateien Ihrer aktuellen MySQL-Übersetzung. Optional können Sie auch hier wieder das Zip-Binary von www.mysql.com benutzen, wo ebenfalls die Dateien enthalten sind. Im Folgenden das Beispiel einer Vorbereitung bei einer eigenen MySQL-Kompilation:

```
cp C:\release\compile\mysql-5.0.19\lib_release\*.lib C:\work\win32build\lib
cp C:\release\compile\mysql-5.0.19\lib_release\*.dll C:\work\win32build\lib\dlls
cp C:\release\compile\mysql-5.0.19\include\* C:\work\win32build\include
```

6. Apache2-Unterstützung vorbereiten

Hierfür benötigen Sie erneut die Lib- und Include-Dateien Ihrer Apache2-Kompilation für das Verzeichnis `C:\work\win32build`. Beispiel:

```
cp C:\apache2\lib\* C:\work\win32build\lib
cp C:\apache2\include\* C:\work\win32build\include
```

Und das war es schon mit den Vorbereitungen. Nein, nicht ganz!

7. Include-Pfade korrigieren

Abhängig von dem von Ihnen eingesetzten Microsoft-Compiler und der zu übersetzenden PHP-Version kommt es leider recht häufig zu Übersetzungsfehlern aufgrund fehlender Einbindungen von Include-Dateien. Der eine mögliche Grund hierfür ist recht einfach: Der Linker Ihres Compilers findet die einzubindende Datei nicht, weil sie am gesuchten Ort tatsächlich nicht existiert. Die zweite mögliche Ursache ist dagegen schon ärgerlicher: Die per Include einzubindende Datei existiert zwar, aber der Include-Pfad wird vom Compiler falsch interpretiert. Ein Beispiel hierfür ist die Einbindung der `timelib_config.h` in die `timelib_structs.h`: mit der Syntax `#include <timelib_config.h>` findet unser VC++-Compiler die Datei nicht. Und das, obwohl sie im selben Ordner liegt! Richtig wäre in diesem Fall `#include "timelib_config.h"`. Wenn das Problem auch bei Ihrem Compiler auftritt, hilft es nichts: Sie müssen alle Includes in der `timelib_structs.h` entsprechend von Hand korrigieren.

Datei: `C:\work\php-5.1.2\ext\date\lib\timelib_structs.h`

8. Über *buildconf* die *configure.js* erstellen

Aber jetzt geht es los! Rufen Sie die Eingabeaufforderung auf, wechseln Sie in Ihr PHP-Arbeitsverzeichnis und erstellen Sie im ersten Schritt mit *buildconf* die *configure.js*, mit der Sie – wie unter Linux – Ihre PHP-Kompilation später anpassen können:

```
cd c:\work\php-5.1.2
buildconf
```

9. *Configure* für Apache2- und MySQL-Unterstützung

In diesem Beispiel wählen Sie nun gezielt die Apache2- und MySQL-Unterstützung für PHP 5.1.2 aus. Mit der Anweisung `cscript /nologo configure.js`, gepaart mit jeweils optionalen Anweisungen, initiiert und überprüft das *Configure*-Programm nun diese Auswahl.

```
cscript /nologo configure.js --enable-prefix=D:\php5 --disable-debug ↵
--with-mysql=shared --with-mysqli=shared --enable-apache2handler=shared
```

Mit der Option `--enable-prefix` kennzeichnen Sie das Verzeichnis für die fertigen PHP-Binärdateien. Die Anweisung `--disable-debug` weist die Übersetzung als Release aus.

Danach folgen die MySQL- und Apache2-Abschnitte. Sie werden mit `shared` gekennzeichnet.

Damit erwartet PHP alle Lib- und Include-Dateien im Arbeitsbereich `C:\work\win32build`. Wenn nun alles gut läuft, erhalten Sie einen Output wie den in Abbildung 19.10:

```

C:\WINNT\System32\cmd.exe
Checking for library iconv_a.lib;iconv.lib ... ..\win32build\lib\iconv_a.lib
Checking for libxml/parser.h ... ..\win32build\include
Enabling extension ext\libxml
Checking for libxml/xmlwriter.h ... ..\win32build\include
Enabling extension ext\xmlwriter
Enabling extension ext\xmlreader
Enabling extension ext\xml
Enabling extension ext\wddx
Checking for timelib.config.h ... ..\date\lib
Enabling extension ext\simplexml
Enabling extension ext\dom

Creating build dirs...
Generating files...
Generating Makefile
Generating main/internal_functions.c
[content unchanged; skipping]
Generating main/config.w32.h
Done.

Type 'nmake' to build PHP
C:\work\php-5.1.2>

```

Abbildung 19.10: Das configure unter Windows auf der Kommandozeile

Mit der Option `--enable-prefix` kennzeichnen Sie das Verzeichnis für die zukünftigen PHP-Binaries. Die Anweisung `--disable-debug` weist die Übersetzung als Release aus. Danach folgen die MySQL- und Apache2-Abschnitte. Sie werden mit `shared` gekennzeichnet. Damit erwartet PHP alle Lib- und Include-Dateien im Arbeitsbereich `C:\work\win32build`. Wenn nun alles gut verläuft, erhalten Sie einen Output wie diesen:

Listing 19.1: Configure prüft die Systemvoraussetzungen für die gewünschte PHP5-Übersetzung

```

Saving configure options to config.nice.bat
Checking for cl.exe ... <in default path>
Checking for cl.exe ... <in default path>
Detected MS compiler version 13
Checking for link.exe ... D:\Programme\MiVisualSt\Vc7\bin
Checking for nmake.exe ... <in default path>
Checking for lib.exe ... <in default path>
Checking for bison.exe ... <in default path>
Checking for flex.exe ... <in default path>
Checking for re2c.exe ... <in default path>
Checking for zip.exe ... <in default path>
Checking for lemon.exe ... <not found>
Checking for mc.exe ... D:\Programme\Microsoft Platform SDK\Bin\
Checking for arpa\nameser.h ... ..\win32build\include
Checking for library resolv.lib ... ..\win32build\lib\resolv.lib

Build dir: Release_TS
PHP Core: php5ts.dll and php5ts.lib

Checking for wsapi.h ... <in default path>
Enabling IPv6 support

```

```

Checking for NewAPIs.h ... <in default path>
Enabling SAPI sapi\cli
Enabling SAPI sapi\cgi
Checking for httpd.h ... ..\win32build\include
Checking for library libhttpd.lib ... ..\win32build\lib\libhttpd.lib
Checking for library libapr.lib ... ..\win32build\lib\libapr.lib
Checking for library libaprutil.lib ... ..\win32build\lib\libaprutil.lib
Enabling SAPI sapi\apache2handler
Enabling extension ext\zlib
Checking for library zlib.lib ... ..\win32build\lib\zlib.lib
Checking for zlib.h ... ..\win32build\include
Enabling extension ext\tokenizer
Checking for timelib_config.h ... ext/date/lib
Enabling extension ext\standard
Enabling extension ext\spl
Enabling extension ext\session
Enabling extension ext\reflection
Enabling extension ext\pcre
Enabling extension ext\odbc
Checking for library libmysql.lib ... ..\win32build\lib\libmysql.lib
Checking for mysql.h ... ..\win32build\include
Enabling extension ext\mysql [shared]
Checking for library libmysql.lib ... ..\win32build\lib\libmysql.lib
Checking for mysql.h ... ..\win32build\include
Enabling extension ext\mysql [shared]
Checking for iconv.h ... ..\win32build\include
Enabling extension ext\iconv
Checking for library libjpeg.lib ... ..\win32build\lib\libjpeg.lib
Checking for library freetype2.lib ... ..\win32build\lib\libfreetype2.lib
Checking for library libpng.lib ... <not found>
Enabling extension ext\ftp
Enabling extension ext\date
Enabling extension ext\ctype
Checking for library oleaut32.lib ... <in LIB path> OleAut32.Lib
Enabling extension ext\com_dotnet
Checking for mscoree.h ... <not found>
Checking for mscoree.h ... <not found>
Enabling extension ext\calendar
Enabling extension ext\bcmath
Checking for library libxml2_a.lib;libxml2.lib ... ..\win32build\lib\libxml2_a.lib
Checking for library iconv_a.lib;iconv.lib ... ..\win32build\lib\iconv_a.lib
Checking for libxml/parser.h ... ..\win32build\include
Enabling extension ext\libxml
Checking for libxml/xmlwriter.h ... ..\win32build\include
Enabling extension ext\xmlwriter
Enabling extension ext\xmlreader
Enabling extension ext\xml
Enabling extension ext\wddx
Checking for timelib_config.h ... ext/date/lib

```

```
Enabling extension ext\simplexml
Enabling extension ext\dom

Creating build dirs...
Generating files...
Generating Makefile
Generating main/internal_functions.c
    [content unchanged; skipping]
Generating main/config.w32.h
Done.
```

Type 'nmake' to build PHP

Das Skript moniert die fehlende GD-Library. Außerdem fehlt beispielsweise die `mscoree.h`. Aber für das konkrete Beispiel unserer Übersetzung besitzen wir alles. Also kompilieren (`nmake`) und installieren Sie danach die Binärdateien (`nmake install`) in diesem Beispiel unter `D:\php5`. Das Übersetzen könnte etwas dauern. Gönnen Sie sich nun also eine kleine Erfrischung.

```
nmake
nmake install
```

Hinweis



Die Datei `winres.h` fehlt? Sie finden eine entsprechende Datei im `include`-Ordner in Ihrem Microsoft Platform SDK unter dem Namen `WinResrc.h`. Kopieren Sie die `WinResrc.h` nach `c:\work\php-5.1.2\win32\build` und benennen Sie diese danach um in `winres.h`.

Wenn Sie nach der erfolgreichen Übersetzung mit diesem Paket weiterarbeiten wollen, geben Sie erst folgende Anweisung ein:

```
nmake clean
```

Nun müssen Sie erneut die Anweisung `cscript /nologo configure.js` mit den von Ihnen gewünschten Optionen durchführen, um eine veränderte Konfiguration für eine neue Übersetzung von PHP einzurichten.

**Tipp**

Sie haben hier die *shared*-Methode verwendet, wobei Sie im Vorfeld alle Include- und Library-Dateien in das zentrale Arbeitsverzeichnis `C:\work\win32build` kopiert haben. Alternativ können Sie die ganze »Kopiererei« z.B. für den Apache 2 sein lassen und die extra *Libs* und *Includes* über das `configure` angeben. Das folgende Beispiel erstellt PHP 5 (nur) mit dem *apache2*-Modul:

```
cscript /nologo configure.js --enable-prefix=D:\php5 --disable-debug
--with-extra-includes="C:\apache2\include" --with-extra-
libs="C:\apache2\lib" --enable-apache2handler
```

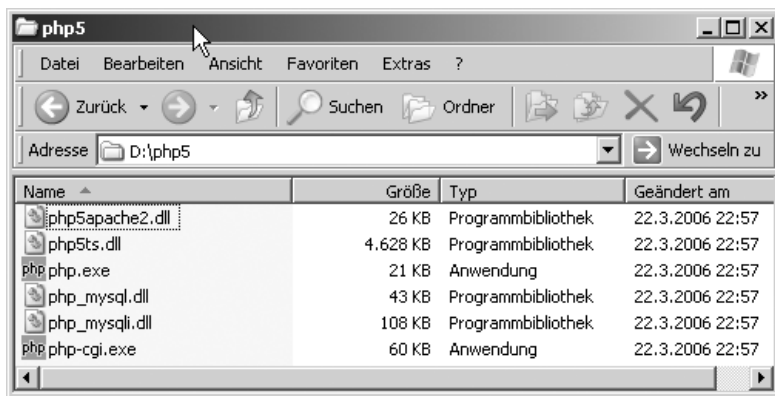


Abbildung 19.11: Nach *nmake install* ist am Zielort alles vorhanden.

Und wie geht es weiter?

Nun haben Sie hoffentlich die neuen Binärdateien. Denken Sie aber daran, dass Sie das gesamte XAMPP-PHP mit Ihren neuen PHP-Dateien ersetzen (überschreiben) müssen, um mit Ihren Modulen und anderen Erweiterungen arbeiten zu können. Das trifft insbesondere neben der `php.exe` auf die zentrale `php5ts.dll` zu. Aber was Ihnen vielleicht wirklich auf dem Herzen liegt, ist die Frage nach den weiteren *Extensions*. Auch diese können Sie nun (versuchen zu) kompilieren. Zuvor brauchen Sie unter Umständen aber wieder zusätzliche GNU-Pakete. Für die GD-Erweiterung benötigen Sie mindestens das Paket *GD* und ggf. auch das Paket *lib*. Sie etablieren die *Includes* und *Libraries* nach dem hier vorgestellten Schema unter `win32build`. Und danach können Sie es probieren.

```
cscript /nologo configure.js --enable-prefix=D:\php5 --disable-debug --with-
mysql=shared --with-mysqli=shared --enable-apache2handler=shared --with-gd=shared
```

Sie merken es vielleicht schon: Um PHP mit allen Erweiterungen zu übersetzen, bedarf es Zeit, Kreativität und starker Nerven. Damit bewegen Sie sich auf einem Niveau, auf dem wir Ihnen viel Glück wünschen, aber jetzt allein lassen müssen.

**Tipp**

Noch ein Tipp zum Schluss: Alle Optionen für `configure` erhalten Sie über:

```
cscrip /nologo configure.js --help
```

19.2 Linux

XAMPP für Linux besteht aus etwa 30 bis 40 einzelnen Komponenten, die wir für XAMPP jeweils übersetzt haben. Nun wäre es sicherlich für dieses Buch viel zu umfangreich und auch zu langweilig, auf das Übersetzen jeder einzelnen Komponente einzugehen, und so werden wir uns mit einem Kompromiss begnügen.

In den nächsten Abschnitten werden wir generell beschreiben, wie das Übersetzen einer Komponente von XAMPP unter Linux funktioniert. Mit dieser Beschreibung sind Sie in der Lage, die meisten, wenn nicht sogar alle Komponenten von XAMPP zu übersetzen und zu installieren.

In den darauf folgenden Abschnitten zeigen wir dann noch in allen Einzelheiten, wie Sie unter Linux Apache, PHP und Perl übersetzen.

19.2.1 Vorbereitungen

Um selbst Komponenten für XAMPP für Linux zu kompilieren, benötigen Sie das sog. Entwicklungs-Paket. Sie können es unter der URL <http://www.apachefriends.org/de/xampp-linux.html> herunterladen, unter der Sie auch XAMPP für Linux herunterladen können (siehe Abbildung 19.12).

Das Entwicklungs-Paket enthält neben Bibliotheken und Header-Dateien, die für das Kompilieren notwendig sind, auch die mitgelieferten Dokumentationen der einzelnen Softwarepakete.

Das Entwicklungs-Paket enthält aber nicht den Quellcode der XAMPP-Komponenten. Das würde den Download einfach zu umfangreich machen.

Um das Entwicklungs-Paket zu installieren, gehen Sie bitte folgendermaßen vor:

1. Laden Sie das Entwicklungs-Paket für die Linux-Version von XAMPP von der Apache Friends-Website herunter.
2. Melden Sie sich als Systemadministrator *root* in einer Shell an und wechseln Sie mit dem `cd`-Befehl in das Verzeichnis, in dem sich das heruntergeladene Download-Paket befindet.
3. Installieren Sie das Entwicklungs-Paket mit folgendem Befehl:

```
tar xvfz xampp-linux-devel-1.5.3.tar.gz -C /opt
```

Nachdem das Entwicklungs-Paket installiert ist, können Sie selbst Komponenten für XAMPP übersetzen.

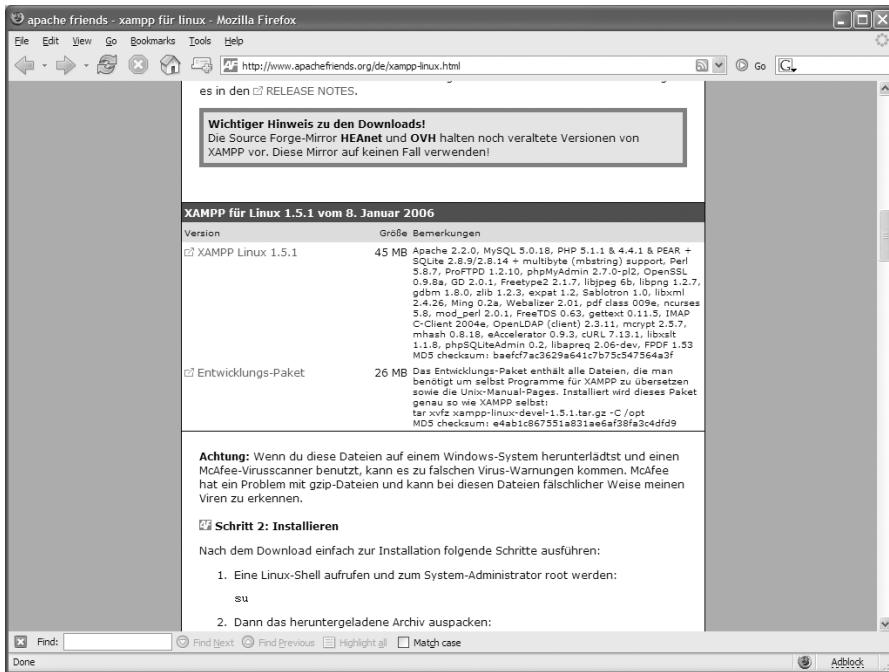


Abbildung 19.12: Download des Entwicklungs-Pakets von XAMPP

19.2.2 Generelle Vorgehensweise

Das Selbstkompilieren von Softwarepaketen gilt bei Einsteigern und auch bei jüngeren Profis als besonders ehrenhafte Tätigkeit. Ob das nun wirklich gerechtfertigt ist oder nicht, sei dahingestellt, aber auf alle Fälle ist das Selbstkompilieren nicht so schwierig, wie man weitläufig meint.

Grundsätzlich kann man den Vorgang in fünf einfache Schritte unterteilen:

1. Das Herunterladen des Quellcodes

Um etwas kompilieren zu können, muss es erst einmal da sein. Also besteht der erste Schritt im Herunterladen des Quellcodes. In der Regel findet man diesen auf der Website des jeweiligen Softwarepakets.

Zum Beispiel:

<http://httpd.apache.org/download.cgi>

<http://www.php.net/downloads.php>

<http://dev.mysql.com/downloads/mysql/>

2. Quellcode-Archiv auspacken

Der Quellcode wird in der Regel in `.tar.gz`-Dateien angeboten. Diese Dateien sind mit *gzip* komprimierte tar-Archive. Nach dem Herunterladen müssen sie zunächst ausgepackt werden.

Dazu rufen Sie folgenden Befehl auf (der Dateiname ist ein Platzhalter für die von Ihnen heruntergeladene Datei):

```
tar xvfz supersoft-1.2.3.tar.gz
```

Manchmal liegen Softwarepakete aber auch in `.tar.bz`-Dateien vor. Diese Dateien sind mit *bzip2* komprimiert und meist etwas kleiner als die mit *gzip* komprimierten. Diese Dateien können Sie mit folgendem Befehl entpacken:

```
tar xvfj supersoft-1.2.3.tar.bz
```

Nach diesem Befehl befinden sich die Dateien aus dem Archiv in einem Unterverzeichnis des aktuellen Verzeichnisses. Meist heißt dieses Verzeichnis so ähnlich wie das Archiv selbst. Im obigen Beispiel würde das Verzeichnis `supersoft-1.2.3` heißen.

3. Anpassen des Quellcodes

Der Quellcode sollte so geschrieben sein, dass er unabhängig von dem Betriebssystem ist, für das er eingesetzt werden soll. Daher muss er in diesem Schritt zunächst an das Zielbetriebssystem – in diesem Fall Linux – angepasst werden.

Diese Anpassung übernimmt in der Regel der `configure`-Befehl. Wechseln Sie aber zunächst in das im letzten Schritt ausgepackte Verzeichnis:

```
cd supersoft-1.2.3
```

Nun rufen Sie den `configure`-Befehl auf:

```
./configure --prefix=/opt/lampp
```

Mit diesem Befehl starten Sie die Anpassung des Quellcodes an Ihr Linux-System. Mit `--prefix=/opt/lampp` weisen Sie den `configure`-Befehl an, den Quellcode so anzupassen, dass die Software später in das Verzeichnis `/opt/lampp` installiert werden soll.

Dieser Vorgang kann – je nach System – einige Minuten in Anspruch nehmen. In dieser Zeit werden Sie folgende Meldungen auf Ihrem Bildschirm beobachten können:

```
checking build system type... i686-pc-linux-gnu
checking host system type... i686-pc-linux-gnu
checking target system type... i686-pc-linux-gnu
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for sigthreadmask... no
checking for gawk... yes
```


Der `configure`-Befehl überprüft, ob Ihr System bestimmten Anforderungen entspricht oder nicht. So stellt er oben zum Beispiel fest, dass das System einen `gawk` besitzt, aber kein `sigthreadmask`. Was das genau ist, muss Sie aber nicht interessieren: Solange der `configure`-Befehl weiterläuft, ist es nicht schlimm, wenn mal etwas fehlt. Es ist sogar die Regel, dass einige Sachen fehlen, dazu ist dieser Anpassungsprozess auch da: damit er die Software an Ihr System anpasst.

Erst wenn `configure` abbricht wie in der folgenden Meldung, müssen Sie aktiv werden:

```
checking for OpenSSL... Could not find OpenSSL
Looks like you've forgotten to install OpenSSL development RPM
```

In diesem Fall findet `configure` nicht die Entwicklungsdateien von OpenSSL, die offenbar von diesem Softwarepaket benötigt werden.

Im Falle von XAMPP haben Sie vergessen, das XAMPP-Entwicklungs-Paket zu installieren oder dem `configure`-Befehl nicht gesagt, wo er das OpenSSL von XAMPP findet. Zum letzten Punkt finden Sie mehr Informationen im nächsten Abschnitt »Apache neu übersetzen«.

4. Kompilierung starten

Nachdem der Anpassungsprozess erfolgreich abgelaufen ist, kann die Kompilierung der Software gestartet werden. Dazu rufen Sie einfach den folgenden Befehl in Ihrer Shell auf:

```
make
```

Dieser Vorgang kann noch länger dauern. Besonders wenn Sie ein sehr altes System verwenden, kann die Kompilierung mehrere Stunden in Anspruch nehmen. Auf einem aktuellen Rechner sollte dieser Vorgang aber nach etwa 15 Minuten abgeschlossen sein.



Hinweis

Dass während des Kompilierens Warnungen auf dem Bildschirm erscheinen, ist durchaus normal. Lassen Sie sich davon nicht beunruhigen. Sollten Sie in diesem Schritt aber Fehlermeldungen erhalten, die den Abbruch des Kompilierungsvorgangs zur Folge haben, dann wird es knifflig.

Wenn Sie Erfahrung in C-Programmierung haben, können Sie versuchen, die Fehlermeldung zu interpretieren und den Fehler zu beseitigen. Eine Recherche über die Fehlermeldung bei Google kann auch hilfreich sein.

In den meisten Fällen jedoch dürfte Ihr Versuch, etwas selbst zu kompilieren, hier beendet sein.

5. Installation der Software

Ist die Kompilierung bis zum Ende durchgelaufen, kann die Software installiert werden. Dazu rufen Sie einfach den folgenden Befehl auf:

```
make install
```

Das war's. Durch diesen Befehl wird die Software auf Ihrem System installiert und kann verwendet werden.

Läuft alles so, wie Sie es sich vorgestellt haben, können Sie das Quellcode-Verzeichnis `supersoft-1.2.3` und das heruntergeladene Archiv `supersoft-1.2.3.tar.gz` löschen. Beides wird nun nicht mehr benötigt.

19.2.3 Beispiel: Apache neu übersetzen

Im letzten Abschnitt haben Sie gesehen, wie einfach es eigentlich ist, Software selbst zu kompilieren. Wenn es dabei noch ein Geheimnis gibt, dann steckt es im Aufruf des `configure`-Befehls. Mit ihm wird nicht nur die Software auf Ihr System angepasst, sondern Sie können damit auch die Software nach Ihren Wünschen konfigurieren.

Im Falle vom Apache können Sie mit `configure` zum Beispiel bestimmen, welche Apache-Module Sie benutzen möchten oder welche MPMs der Apache verwenden soll. Je nach Ihren Anforderungen kann so ein `configure`-Befehl entweder recht übersichtlich ausfallen (wie im vorherigen Abschnitt) oder einen nahezu monströsen Umfang annehmen.

So sieht beispielsweise der `configure`-Befehl, mit dem wir den Apache für XAMPP übersetzt haben, aus:

Listing 19.2: Der configure-Aufruf für den XAMPP-Apache

```
./configure --prefix=/opt/lampp --enable-nonportable-atomics --enable-so ➤
--enable-cgid --sysconfdir=/opt/lampp/etc --enable-auth-anon ➤
--enable-auth-dbm --enable-auth-digest --enable-file-cache ➤
--enable-echo --enable-charset-lite --enable-cache --enable-disk-cache ➤
--enable-mem-cache --enable-example --enable-ext-filter --enable-case-filter ➤
--enable-case-filter-in --enable-deflate --enable-mime-magic ➤
--enable-cern-meta --enable-expires --enable-headers --enable-usertrack ➤
--enable-unique-id --enable-proxy --enable-proxy-connect --enable-proxy-ftp ➤
--enable-proxy-http --enable-bucketeer --enable-http --enable-info ➤
--enable-suexec --enable-cgid --enable-vhost-alias --enable-speling ➤
--enable-rewrite --enable-so --with-z=/opt/lampp --with-ssl=/opt/lampp ➤
--with-expat=/opt/lampp --enable-dav --enable-dav-fs --enable-ssl=/opt/lampp ➤
--enable-mods-shared=most --with-mpm=prefork --with-suexec-caller=nobody ➤
--with-suexec-docroot=/opt/lampp/htdocs --without-berkeley-db --enable-ldap ➤
--with-ldap --enable-auth-ldap --disable-ipv6
```

In diesen `configure`-Befehlen liegt quasi das große Geheimnis des Kompilierens. Damit Sie unsere Kompilierung nachvollziehen können, haben wir bei XAMPP für Linux alle `configure`-Befehle, die wir zum Kompilieren benutzten, mit ins XAMPP gepackt. Dort finden Sie in der Datei `/opt/lampp/share/lampp/configures.tar.gz` ein `tar.gz`-Archiv mit allen unseren `configure`-Befehlen.

Dieses Archiv enthält für jede XAMPP-Komponente ein Verzeichnis, in dem jeweils eine Datei mit dem jeweiligen `configure`-Befehl liegt:

```
apache/configure-oswald
mysql/configure-oswald
openldap/configure-oswald
openssl/configure-oswald
proftpd/configure-oswald
php/configure-php4-oswald
php/configure-php5-oswald
...
```

Mit Hilfe dieser Dateien ist es sehr einfach, Komponenten so zu übersetzen, dass sie sich auch problemlos ins XAMPP integrieren. Am Beispiel von Apache zeigen wir Ihnen, wie Sie dabei am besten vorgehen:

1. Herunterladen des Quellcodes

Den Quellcode von Apache können Sie unter der Adresse <http://httpd.apache.org/download.cgi> herunterladen. Speichern Sie das Archiv in einem Verzeichnis namens `apache` irgendwo auf Ihrem Linux-System.

2. Quellcode-Archiv auspacken

Entpacken Sie das Archiv in diesem Verzeichnis:

```
tar xvfz httpd-2.2.0.tar.gz
```

Während der Arbeit an diesem Buch setzten wir die aktuelle Version 2.2 von Apache ein. In Ihrem Fall wird es sehr wahrscheinlich eine andere Version sein.

3. Kopieren der `configure`-Vorlage

Kopieren Sie die Datei `apache/configure-oswald` aus dem Archiv `configures.tar.gz` (siehe oben) in dieses Verzeichnis. Ihr Verzeichnis sollte jetzt etwa so aussehen:

```
-rw-r--r--  1 root    root      1663 Dec  1 12:19 configure-oswald
drwxr-xr-x 11 root    root       1240 Jan  7 10:19 httpd-2.2.0
-rw-rw-r--  1 root    root    6224116 Nov 30 21:21 httpd-2.2.0.tar.gz
```

4. Anpassung und Kompilierung starten und Software installieren

Wechseln Sie nun in das ausgepackte Quellcode-Verzeichnis:

```
cd httpd-2.2.0
```

Statt des `configure`-Befehls rufen Sie nun folgenden Befehl auf:

```
sh ../configure-oswald
```

Durch diesen Aufruf werden folgende Schritte automatisch ausgeführt:

1. Intern wird der `configure`-Befehl aufgerufen, wie wir ihn zum Kompilieren des Apache verwendet haben.
2. Ist der `configure`-Befehl erfolgreich abgelaufen, wird automatisch der `make`-Befehl aufgerufen und das eigentliche Kompilieren gestartet.
3. War die Kompilierung erfolgreich, wird automatisch die Software installiert.

Und das war's. Der neue Apache ist übersetzt und automatisch ins XAMPP integriert.

Genau auf diese Weise übersetzen wir eine Komponente für XAMPP.

19.2.4 Besonderheiten bei PHP

Grundsätzlich ist der Übersetzungsvorgang bei PHP der gleiche wie beim Apache. Allerdings gibt es zwei Besonderheiten zu berücksichtigen, wenn Sie die Übersetzung von PHP planen:

Zwei configure-Vorlagen

Für PHP finden Sie zwei `configure`-Vorlagen in der `/opt/lampp/share/lampp/configures.tar.gz`-Vorlagen-Datei:

```
php/configure-php4-oswald
php/configure-php5-oswald
```

Die Namen der Dateien deutet es schon an: Verwenden Sie die erste Datei, wenn Sie PHP 4 übersetzen wollen, und die zweite Datei, wenn es um PHP 5 geht.

Oracle, Interbase/Firebird, PostgreSQL

Damit PHP genau so übersetzt werden kann, wie es im XAMPP vorliegt, müssen diese drei Datenbanken (zumindest in der Clientform) auf dem Linux-Rechner installiert sein. Ansonsten schlägt der `configure`-Befehl mit einer entsprechenden Meldung fehl.

Wenn Sie keine dieser Datenbanken benutzen möchten, dann löschen Sie einfach die in dem folgenden Listing hervorgehobenen Stellen aus der `configure-php4-oswald` bzw. `configure-php5-oswald`.

Listing 19.3: Der `configure`-Aufruf für das XAMPP-PHP

```
env PATH="/opt/lampp/bin:$PATH" ORACLE_HOME="/opt/oracle/OraHome1" ❶
CFLAGS="-O6
-I$ORACLE_HOME/network/public -I/opt/lampp/include/libpng❷
-I/opt/lampp/include/ncurses -I/opt/lampp/include -L/opt/lampp/lib" ./configur❸
e --prefix=/opt/lampp --with-apxs2=/opt/lampp/bin/apxs --with-config❹
-file-path=/opt/lampp/etc --with-mysql=/opt/lampp --disable-debug --enable-b❺
cmath --enable-calendar --enable-ctype --enable-dbase --enable-discard-path --en❻
```

```

able-exif --enable-filepro --enable-force-cgi-redirect --enable-ftp --enable-gd-
imgstrttf --enable-gd-native-ttf --with-ttf --enable-magic-quotes --enable-m
emory-limit --enable-shmop --enable-sigchild --enable-sysvsem --enable-sysvshm
--enable-track-vars --enable-trans-sid --enable-wddx --enable-yp
--with-ftp --with-gdbm=/opt/lampp --with-jpeg-dir=/opt/lampp --with-png-di
r=/opt/lampp --with-freetype-dir=/opt/lampp --without-xpm --with-zlib=yes --with
-zlib-dir=/opt/lampp --with-openssl=/opt/lampp --with-expat-dir=/opt/lampp --e
nable-xslt=/opt/lampp --with-xsl=/opt/lampp --with-ming=/opt/lampp --w
ith-dom=/opt/lampp --with-ldap=/opt/lampp --with-ncurses=/opt/lampp --with-gd -
with-imap-dir=/opt/lampp --with-ldap=/opt/lampp --with-imap-ssl --with-imap=/opt/lampp --wi
th-gettext=/opt/lampp --with-mssql=/opt/lampp --with-interbase=sha
red,/opt/interbase --with-mysql-sock=/opt/lampp/var/mysql/mys
ql.sock --with-oci8=shared --with-mcrypt=/opt/lampp --with-mhash=/opt/
lampp --en
able-sockets --enable-mbstring=all --with-curl=/opt/lampp --enable-mbregex --
enable-zend-multibyte --enable-exif --with-bz2=/opt/lampp --with-sqlite=shared,/
opt/lampp --with-libxml-dir=/opt/lampp --enable-soap --enable-pcntl --enable-
dbx --with-mysqli=/opt/lampp/bin/mysql_config --with-mime-magic --wit
h-pgsql=shared,/opt/lampp/postgresql --with-iconv --enable-dio
--with-pdo-mysql=/opt/lampp
--with-pdo-pgsql=/opt/lampp/postgresql --with-pdo-sqlite

```

Sollten Sie allerdings eine Datenbankunterstützung in PHP benötigen, dann passen Sie die Pfadangaben in dieser Datei so an, dass sie der Datenbankinstallation auf Ihrem System entsprechen.

Mit diesen beiden Hinweisen sollten Sie auch leicht in der Lage sein, PHP für Ihr XAMPP-System neu zu übersetzen.

19.2.5 Perl neu übersetzen

Im Prinzip funktioniert das Übersetzen und Installieren von Perl genau so wie bei den anderen Komponenten von XAMPP – allerdings mit zwei kleinen Unterschieden:

1. Zum einen ist hier der Übersetzungs- und Installationsvorgang nicht ganz so automatisiert, d. h., nachdem Sie die `configure`-Vorlage aufgerufen haben, werden `make` und `make install` nicht automatisch aufgerufen, sondern dies muss manuell erfolgen.
2. Nach der Übersetzung einer neuen Perl-Version ist Perl relativ nackt. D. h., wenn Sie zum Beispiel mit Perl auf eine Datenbank zugreifen wollen, wird das überhaupt nicht funktionieren, da die entsprechenden Perl-Module mit der Datenbankunterstützung fehlen und noch nachträglich installiert werden müssen. Mehr Informationen hierzu finden Sie in Kapitel 22, »Perl erweitern«.



20 PHP erweitern

Es gibt zwei grundlegende Möglichkeiten, PHP zu erweitern: zum einen über *PEAR* und zum anderen über *PECL*.

PEAR steht für *PHP Extension and Application Repository* und erweitert PHP um zusätzliche PHP-Klassen. Vereinfacht kann man sich PEAR als eine große Sammlung vieler PHP-Dateien vorstellen, auf die – um sie zu installieren, zu deinstallieren, upzudaten – über eine genormte Schnittstelle einfach zugegriffen werden kann.

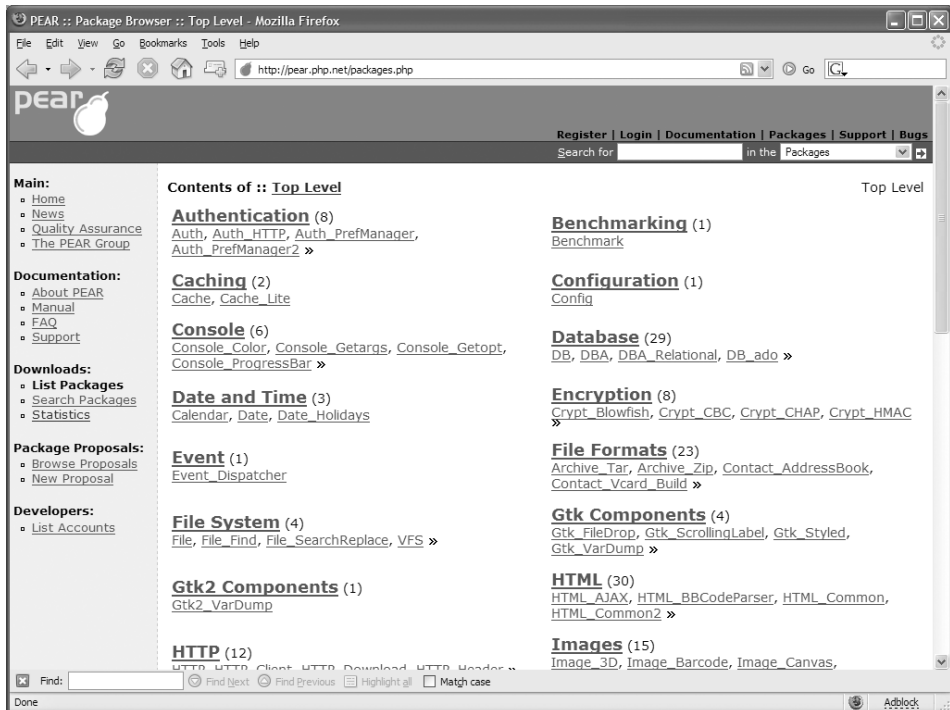


Abbildung 20.1: Über die Homepages <http://pear.php.net> und <http://pecl.php.net> kann in den Paketbeständen gestöbert werden. Als dieses Buch geschrieben wurde, enthielt PEAR 349 Pakete und PECL 142 Pakete.

Die Abkürzung PECL steht für *PHP Extension Community Library* und besteht aus zusätzlichen PHP-Erweiterungen. Gegenüber PEAR bestehen PECL-Pakete nicht aus PHP-Dateien, sondern sind meist in C geschrieben und erweitern PHP auf unterster Ebene um zusätzliche Fähigkeiten. Das bedeutet zum Beispiel, dass zum Installieren zusätzlicher PECL-Pakete in der Regel ein C-Compiler notwendig ist, der auf dem System installiert sein muss.

20.1 PEAR

Die Installation von PEAR-Paketen ist recht komfortabel und einfach für den Anwender gelöst. Sämtliche Aktionen PEAR betreffend werden durch ein einzelnes Skript (unter Linux `/opt/lampp/bin/pear` und unter Windows `<xampp>\php\pear.bat`) erledigt.

In den folgenden Abschnitten bekommen Sie einen Überblick über die wichtigsten Aufgaben, die man mit dem PEAR-Skript durchführen kann.

20.1.1 list

Mit dem `list`-Kommando können Sie sich alle PEAR-Pakete, die auf Ihrem System installiert sind, anzeigen lassen.

Zum Beispiel ergibt diese Eingabe:

```
/opt/lampp/bin/pear list
```

die folgende Ausgabe (gekürzt):

```
INSTALLED PACKAGES, CHANNEL PEAR.PHP.NET:
```

```
=====
PACKAGE          VERSION  STATE
Archive_Tar      1.3.1   stable
Auth              1.2.3   stable
Auth_HTTP        2.1.6   stable
Auth_PrefManager 1.1.4   stable
Auth_RADIUS      1.0.1   stable
Auth_SASL        1.0.1   stable
Benchmark        1.2.3   stable
...
```

20.1.2 list-all

Das `list-all`-Kommando führt neben den installierten Paketen auch die Pakete, die sich zwar im PEAR befinden, aber nicht auf Ihrem System installiert sind, auf:

Zum Beispiel ergibt diese Eingabe:

```
/opt/lampp/bin/pear list-all
```


die folgende Ausgabe (gekürzt):

```
ALL PACKAGES:
=====
PACKAGE                                LATEST      LOCAL
pear/Auth_HTTP                         2.1.6        2.1.6
pear/Auth                             1.3.0r3      1.2.3
pear/Auth_SASL                         1.0.1        1.0.1
pear/LiveUser                          0.16.8
pear/Auth_PrefManager                  1.1.4        1.1.4
pear/Auth_RADIUS                       1.0.4        1.0.1
pear/Auth_PrefManager2                  2.0.0 dev1
pear/LiveUser_Admin                     0.3.6
...
```

In der Spalte **LATEST** steht die Versionsnummer der im PEAR aktuell verfügbaren Version. Unter **LOCAL** finden Sie die Versionsnummer der lokal installierten Version, sofern dieses PEAR-Paket überhaupt installiert ist. Ist es nicht installiert, dann ist hier die Spalte leer.

20.1.3 list-upgrades

Mit dem `list-upgrades`-Kommando erhalten Sie eine Übersicht über alle verfügbaren Upgrades für die auf Ihrem System installierten PEAR-Pakete.

Zum Beispiel ergibt diese Eingabe:

```
/opt/lampp/bin/pear install pager
```

die folgende Ausgabe (gekürzt):

```
PEAR.PHP.NET AVAILABLE UPGRADES (STABLE):
=====
CHANNEL    PACKAGE          LOCAL          REMOTE
pear.php.net Auth_RADIUS      1.0.1 (stable) 1.0.4 (stable)
pear.php.net Benchmark        1.2.3 (stable) 1.2.4 (stable) pear.php.net
Cache_Lite  1.5.2 (stable) 1.6.0 (stable)
pear.php.net Config           1.10.4 (stable) 1.10.5 (stable)
pear.php.net Date              1.4.3 (stable) 1.4.6 (stable)
pear.php.net DB_DataObject     1.7.15 (stable) 1.8.2 (stable)
pear.php.net DB_QueryTool      1.0.0 (stable) 1.0.1 (stable)
pear.php.net File              1.2.1 (stable) 1.2.2 (stable)
...
```

20.1.4 install

Das am meisten genutzte PEAR-Kommando dürfte `install` sein. Mit diesem Kommando installieren Sie weitere PEAR-Pakete.

Zum Beispiel installiert das folgende Kommando das PEAR-Paket *pager*:

```
/opt/lampp/bin/pear install pager
```

Dies ergibt folgende Ausgabe:

```
downloading Pager-2.3.4.tgz ...
Starting to download Pager-2.3.4.tgz (26,639 bytes)
.....done: 26,639 bytes
install ok: channel://pear.php.net/Pager-2.3.4
```

20.1.5 upgrade

Mit dem `upgrade`-Kommando kann man ein einzelnes PEAR-Paket auf die aktuelle Version aktualisieren.

Um bei dem Beispiel des PEAR-Pakets *pager* zu bleiben, ergibt die Eingabe

```
/opt/lampp/bin/pear upgrade pager
```

die folgende Ausgabe:

```
downloading Pager-2.3.5.tgz ...
Starting to download Pager-2.3.5.tgz (26,952 bytes)
.....done: 26,952 bytes
upgrade ok: channel://pear.php.net/Pager-2.3.5
```

20.1.6 upgrade-all

Mit dem `upgrade-all`-Kommando können Sie bequem alle auf Ihrem System installierten PEAR-Pakete auf den aktuellen Stand bringen.

```
/opt/lampp/bin/pear upgrade-all
```

Obige Eingabe führt zum Beispiel zu folgender Ausgabe (gekürzt):

```
Will upgrade benchmark
Will upgrade cache_lite
Will upgrade config
Will upgrade date
...
Starting to download Benchmark-1.2.4.tgz (7,587 bytes)
.....done: 7,587 bytes
downloading Cache_Lite-1.6.0.tgz ...
Starting to download Cache_Lite-1.6.0.tgz (25,865 bytes)
...done: 25,865 bytes
downloading Config-1.10.5.tgz ...
Starting to download Config-1.10.5.tgz (24,582 bytes)
...done: 24,582 bytes
```

```

downloading Date-1.4.6.tgz ...
Starting to download Date-1.4.6.tgz (53,535 bytes)
...done: 53,535 bytes
...

```

Das `upgrade-all`-Kommando hat zunächst festgestellt, dass die Pakete `benchmark`, `cache_lite`, `config` und `date` in neuen Versionen verfügbar sind, und diese dann im zweiten Schritt automatisch aktualisiert.

20.1.7 uninstall

Und zuletzt sei noch das `uninstall`-Kommando von PEAR genannt. Mit ihm können Sie PEAR-Pakete, die Sie nicht mehr benötigen, deinstallieren.

Zum Beispiel führt die Eingabe

```
/opt/lampp/bin/pear uninstall pager
```

zur folgenden Ausgabe:

```
uninstall ok: channel://pear.php.net/Pager-2.3.3
```

20.2 PECL

PECL verwendet die gleiche Technologie zum Verwalten der Pakete wie PEAR. D. h., sämtliche Kommandos, die im vorherigen Abschnitt für PEAR beschrieben wurden, funktionieren genauso für PECL.

Da PECL-Pakete aber im Gegensatz zu den PEAR-Paketen übersetzt werden müssen, setzt PECL die Anwesenheit eines C-Compilers auf dem System voraus. Unter Linux ist es recht normal, dass ein C-Compiler installiert ist, während dies bei Windows eher seltener der Fall ist.

Aus diesem Grund behandeln wir PECL unter Linux und Windows getrennt.

20.2.1 Unter Linux

Um selbst PECL-Pakete installieren zu können, muss auf dem Linux-System ein C-Compiler (GCC) installiert sein. Eine weitere Anforderung ist die Installation des XAMPP für Linux-Entwicklungs-Pakets.

PECL-Pakete installieren

Mit dem folgenden Befehl wird beispielhaft das PECL-Paket *hash* installiert:

```
/opt/lampp/bin/pecl install hash
```

Nach dem Aufruf dieses Befehls wird das *hash*-Paket heruntergeladen und der Übersetzungsvorgang gestartet.

```

downloading hash-1.1.tgz ...
Starting to download hash-1.1.tgz (100,713 bytes)
.....done: 100,713 bytes
30 source files, building
running: phize
Configuring for:
PHP Api Version:      20041225
Zend Module Api No:   20050922
Zend Extension Api No: 220051025
building in /var/tmp/pear-build-oswald/hash-1.1
running: /tmp/tmpcTmPuq/hash-1.1/configure
checking for egrep... grep -E
checking for a sed that does not truncate output... /bin/sed
checking for gcc... gcc
...
install ok: channel://pear.php.net/hash-1.1

```

Nach dem Übersetzen wird die neue PHP-Erweiterung automatisch installiert und befindet sich nun als *.so*-Datei im Verzeichnis `/opt/lampp/lib/php/extensions/no-debug-non-zts-20050922/`:

```

-rwxr-xr-x 1 root root 325554 Jan 22 17:59
/opt/lampp/lib/php/extensions/no-debug-non-zts-20050922/hash.so

```

Damit die Erweiterung aber auch in PHP eingebunden wird, muss sie noch über die `php.ini`-Datei aktiviert werden. Dazu öffnen Sie die Datei `/opt/lampp/etc/php.ini` in einem Texteditor und suchen nach folgenden Zeilen:

```

extension="zip.so"
extension="sqlite.so"
extension="radius.so"
extension="pgsql.so"

```

Nach diesen Zeilen fügen Sie einfach die folgende Zeile an:

```
extension="hash.so"
```

Nachdem Sie den Apache neu gestartet haben, ist die neue PHP-Erweiterung in PHP verfügbar.

PECL-Pakete deinstallieren

Das Löschen von PECL-Paketen funktioniert genauso wie auch bei PEAR-Paketen mit dem `uninstall`-Kommando. Allerdings muss nach der Deinstallation noch die `extension="xxx.so"`-Zeile aus der `php.ini` entfernt werden.

Zum Beispiel führt der Aufruf

```
/opt/lampp/bin/pecl uninstall hash
```

zu der folgenden Ausgabe:

```
uninstall ok: channel://pear.php.net/hash-1.1
```

Entfernen Sie jetzt noch die Zeile `extension="hash.so"` aus der `php.ini`, starten Sie den Apache neu und die PHP-Erweiterung ist deinstalliert.

20.2.2 Unter Windows

Genauso wie unter Linux muss auch unter Windows ein C-Compiler für die Übersetzung von neuen PECL-Modulen vorhanden sein.

PECL-Pakete installieren

Im Prinzip ist die Syntax mit der unter Linux vergleichbar:

```
<xampp>\php\pecl
```

Doch der korrekte Aufruf der `pecl.bat` sowie ein C++-Compiler sind noch keine Erfolgsgarantie. Versuchen Sie bitte einmal das Linux-Beispiel: die Installation des PECL-Pakets `hash`.

```
<xampp>\php\pecl install hash
```

Die Enttäuschung ist vielleicht groß. Sie könnten nämlich so etwas wie in Abbildung 20.2 sehen.

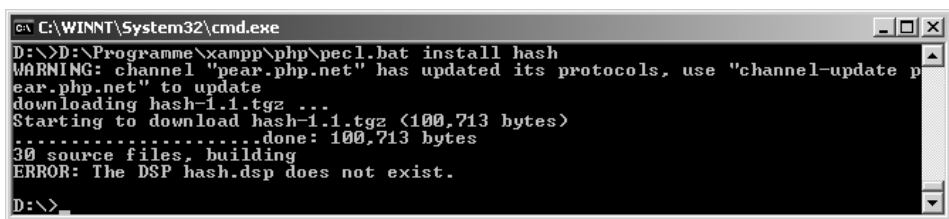


Abbildung 20.2: Eine Projektdatei für Windows ist nicht vorhanden.

Was ist passiert? Das `hash`-Paket gibt es nur für Linux. Im Paket fehlt eine C++-Projektdatei (`.dsp`). Das ist also die Crux: Den allermeisten PECL-Paketen fehlt ein entsprechendes Projekt oder der ganze Quellcode für eine Windows-Übersetzung. Sollten Sie auch mit Perl unter Windows arbeiten, kommt Ihnen das Problem vielleicht bekannt vor, Stichwort: CPAN unter Windows.

Ist also schon alles verloren? Keineswegs! Die beste Alternative zu einer eigenen Kompilation ist die Beschaffung bereits fertig kompilierter Module unter der folgenden URL:

<http://snaps.php.net/>

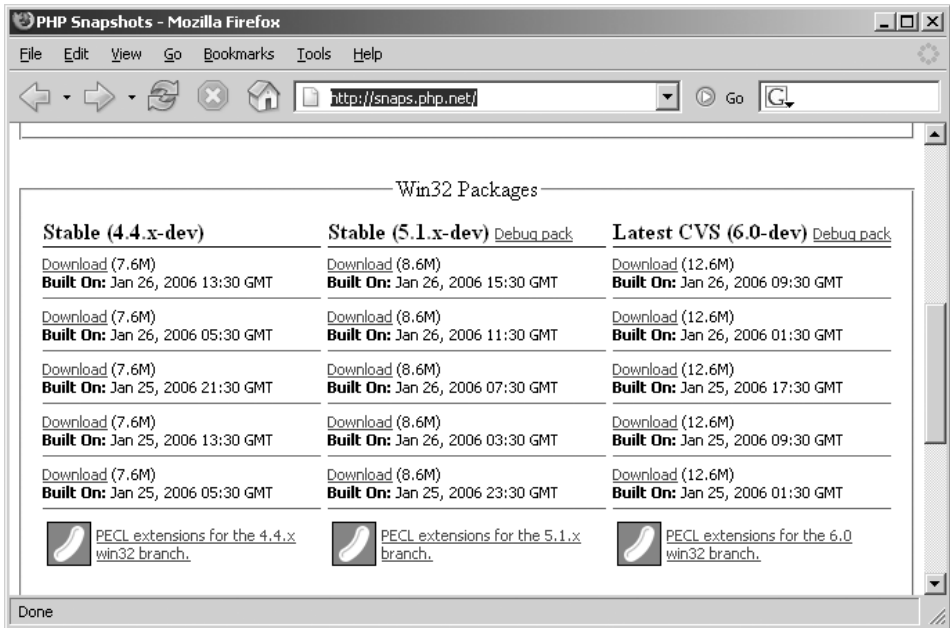


Abbildung 20.3: Vorkompilierte PECL-Erweiterungen unter snaps.php.net

Damit brauchen Sie auch keinen eigenen Compiler mehr. Öffnen Sie einfach die PECL-Seite für Ihre PHP-Version und sehen Sie nach, ob das von Ihnen gesuchte PECL-Modul vorhanden ist. In diesem Beispiel finden wir auch das *hash*-Modul mit dem Namen `php_hash.dll`. Das ausgewählte Modul speichern Sie

- für PHP 4.x.x in das Verzeichnis:

```
<xampp>\php\extensions
```

- für PHP 5.x.x in das Verzeichnis:

```
<xampp>\php\ext
```

Nun brauchen Sie nur noch das neue Modul in den *Extension*-Bereich der `php.ini` einzufügen.

Konfigurationsdatei: `<xampp>\apache\bin\php.ini`

```
extension=php_hash.dll
```

Starten Sie danach den Apache neu. Sollte es dabei zu Problemen mit dem neuen Modul kommen, deaktivieren Sie es mit einem Semikolon zu Beginn der Zeile. Oder Sie löschen die betreffende Zeile und starten den Apache erneut.



Hinweis

Vergewissern Sie sich bitte, dass Sie das passende PHP für die neuen PECL-Module besitzen. Ein PECL-Modul für PHP 5.1x funktioniert beispielsweise nicht mit PHP 5.0.x. Welche aktuelle PHP-Version in XAMPP Verwendung findet, sehen Sie am schnellsten in den *Readme*-Dateien im XAMPP-Hauptverzeichnis. Alternativ rufen Sie über die Eingabeaufforderung das PHP-Programm mit der Option `-v` (für *Version*) auf:

```
<xampp>\php\php -v
```

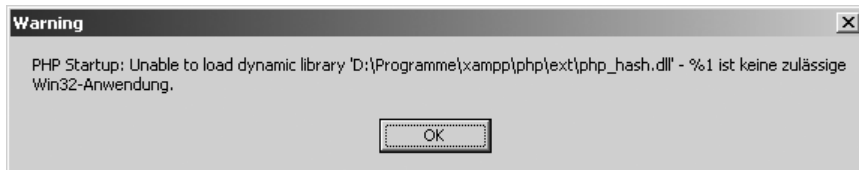


Abbildung 20.4: Das richtige PECL-Modul, aber leider die falsche PHP-Version



21 Apache erweitern

Den Apache – mit seinen ach so vielen Modulen – kann man schon als eine eierlegende Wollmilchsau bezeichnen. Wenn es um die Geschwindigkeit des Apache geht, dann wird ihm oft sogar schon vorgeworfen, dass er zu viel kann und dadurch zu träge und zu langsam geworden ist. In Kapitel 9, »Performance Tuning«, haben wir ja schon festgestellt, dass der Geschwindigkeitsgewinn relativ groß ist, wenn man nicht benötigte Module vermeidet.

Aber trotzdem können Situationen eintreten, in denen der bestehende Funktionsumfang des Apache nicht reicht und man irgendeine besondere Fähigkeit benötigt. Für diese Fälle steht eine Vielzahl zusätzlicher Module zur Erweiterung des Apache zur Verfügung.

Oft wird das Installieren nach dem gleichen Muster ablaufen. In diesem Kapitel sehen Sie beispielhaft am Modul *ModSecurity*, wie Sie eine solche Installation durchführen können.

21.1 Das Beispiel modSecurity

Das Modul *mod_security* erweitert den Apache um eine Art Content Firewall und Intrusion Detection System. Es schützt zum einen den Apache bzw. die »hinter« dem Apache liegende Webapplikation vor unerwünschten Zugriffen von außen und bietet zum anderen zusätzliche Protokollmöglichkeiten, die sonst mit dem Apache allein nicht möglich wären. Zum Beispiel kann man so Anfragen mit einem möglichen feindlichen Hintergrund protokollieren und ist damit in der Lage, auf mögliche Angriffe frühzeitig zu reagieren.

Die Homepage von *modSecurity* lautet <http://www.modsecurity.org>.

21.1.1 Die Kompilierung unter Linux

Um selbst zusätzliche Apache-Module zu übersetzen, müssen Sie das Entwicklungspaket der Linux-Version von XAMPP installiert haben. Erst danach können Sie mit der eigentlichen Übersetzung eines Apache-Moduls beginnen.

Sie finden das Entwicklungs-Paket unter der gleichen Adresse, unter der Sie auch XAMPP für Linux herunterladen können: <http://www.apachefriends.org/de/xampp->

linux.html. Die Installation erfolgt auf die gleiche Weise wie bei XAMPP selbst. Packen Sie das Archiv einfach aus:

```
tar xvfz xampp-linux-devel-1.5.3.tar.gz -C /opt
```

Damit ist das Entwicklungs-Paket installiert und Sie können mit der Installation eines neuen Apache-Moduls beginnen.

1. Laden Sie zunächst das ModSecurity-Paket `modsecurity-apache-1.9.2.tar.gz` (oder eine neuere Version) von der Seite <http://www.modsecurity.org/download/> herunter.
2. Extrahieren Sie das heruntergeladene Archiv:

```
tar xvfz modsecurity-apache-1.9.2.tar.gz
```

3. Wechseln Sie nun in das Verzeichnis `apache2` des neuen, ausgepackten Ordners:

```
cd modsecurity-apache-1.9.2/apache2
```

4. Damit das Modul im nächsten Schritt auch automatisch installiert werden kann, benötigen Sie ab nun Administrations-Rechte. Rufen Sie dazu den folgenden Befehl auf:

```
su
```

Nun folgt die eigentliche Übersetzung und Installation des Apache-Moduls mit der Hilfe des `apxs`-Befehls.

5. Mit der Option `c` weisen Sie den Befehl an, das angegebene C-Programm zu übersetzen. Die Option `i` sorgt dafür, dass das daraus entstehende Apache-Modul automatisch installiert wird. Bei Angabe der Option `a` wird `apxs` dieses Modul außerdem automatisch über die `httpd.conf` aktivieren.

```
/opt/lampp/bin/apxs -cia mod_security.c
```

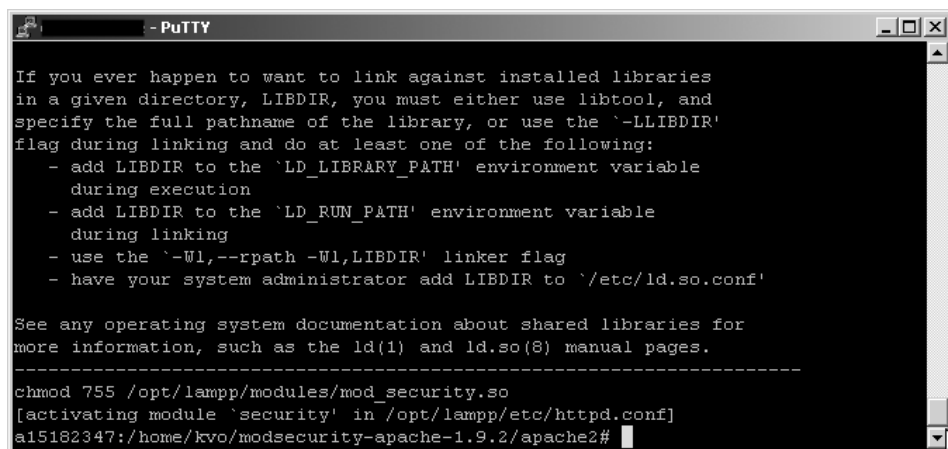


Abbildung 21.1: Ein fertiges `mod_security` auf einem Debian Remote-Rechner

Danach finden Sie das neue Modul unter `/opt/lampp/modules`. Normalerweise brauchen Sie das Modul nicht wie gewohnt in der `httpd.conf` einzubinden. Das hat das APXS-Skript bereits für Sie erledigt und Sie sollten die folgende Zeile finden:

```
LoadModule security_module modules/mod_security.so
```

Eine einfache beispielhafte Konfiguration für das *modSecurity*-Modul erhalten Sie im übernächsten Abschnitt.

21.1.2 Die Kompilierung unter Windows

Um es gleich vorwegzunehmen: Die Übersetzung über APXS (*APache eXtenSion tool*) funktionierte mit dem Apache in der Version 2.2.0 (Windows) nicht. Glücklicherweise war aber die klassische `nmake`-Methode erfolgreich. Sie benötigen hierfür das Entwicklungs-Paket (Devel) von XAMPP oder eine entsprechende Apache-Übersetzung sowie einen VC C++-Compiler.

1. Extrahieren Sie das `modsecurity-apache-1.9.2.tar.gz` z.B. mit 7-zip.
2. Wechseln Sie mit der Eingabeaufforderung im neuen `modsecurity`-Ordner in das `apache2`-Unterverzeichnis:

```
cd modsecurity-apache-1.9.2\apache2
```

3. Öffnen Sie hier `makefile.in` und passen Sie den Wert der `BASE`-Variable an den realen Ort Ihres Apache-Servers an:

```
BASE = "D:/Programme/xampp/apache/"
```

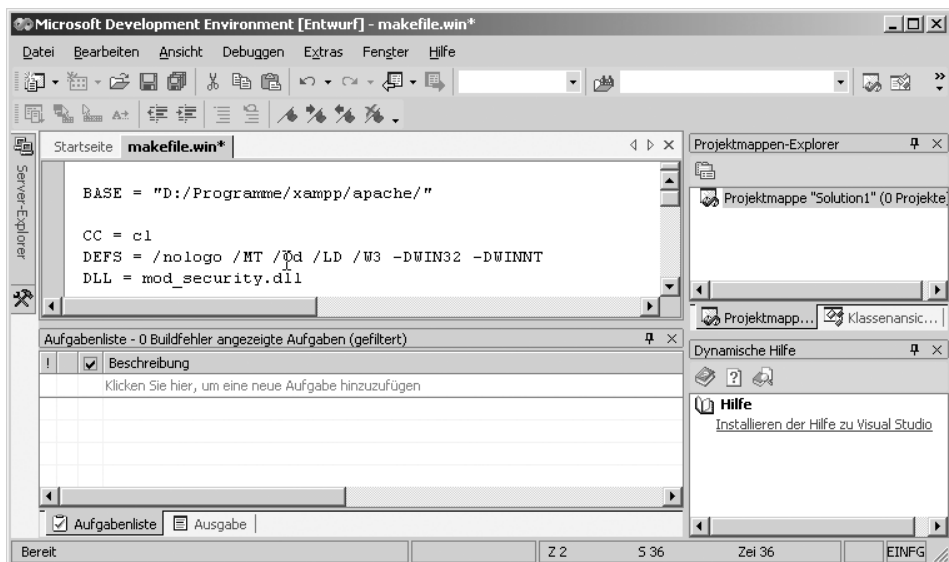


Abbildung 21.2: Wo ist der Apache2? Die Anpassung von `BASE` in der `Makefile.in`

4. Arbeiten Sie im `apache2`-Verzeichnis von `mod_security` weiter und starten Sie mit `nmake` den Übersetzungsvorgang:

```
nmake /f Makefile.win
```

5. Es entstehen nun die folgenden drei Binärdateien. Sie benötigen aber nur die Datei `mod_security.dll`, die Sie nun am besten als `mod_security.so` in dem Verzeichnis `<xampp>\apache\modules` ablegen.

Modul: `mod_security.dll`

Bibliothek: `mod_security.lib`

Objekt: `mod_security.exp`

Alles Weitere passiert in der Apache-Konfiguration. Ein Beispiel hierzu finden Sie in folgendem Abschnitt.

21.1.3 Das Konfigurationsbeispiel

Das hier angeführte Beispiel ist eine Umsetzung der von den Entwicklern von *mod-Security* beigegebenen `httpd.conf.example-minimal`. Für weiterführende Konfigurationsoptionen lesen Sie bitte die Dokumentation auf der Homepage des Projekts:

<http://www.modsecurity.org/documentation/>

<http://www.modsecurity.org/projects/rules/>

Doch kommen wir jetzt zu dem Abschnitt für Ihre `httpd.conf`, den Sie beispielsweise an das Ende der Datei eintragen können.

```
LoadModule security_module modules/mod_security.so
<IfModule mod_security.c>
    # Aktiviere ModSecurity
    SecFilterEngine On
    # Verweigere Statusanfragen mit 403
    SecFilterDefaultAction "deny,log,status:403"
    # Weitere Standards
    SecFilterScanPOST On
    SecFilterCheckURLEncoding On
    SecFilterCheckUnicodeEncoding Off
    # Akzeptiere Byte-Range
    SecFilterForceByteRange 1 255
    # Für Temp-Dateien
    SecUploadDir C:/temp # Hier Windows
    SecUploadKeepFiles Off
    # Nehme nur Relevante auf
    SecAuditEngine RelevantOnly
    SecAuditLog logs/modsec_audit.log
    # Keine Debug-Logging
    SecFilterDebugLevel 0
```

```
SecFilterDebugLog logs/modsec_debug.log
# Nur Anfragen die der Apache versteht werden toleriert
SecFilterSelective REQUEST_METHOD "!(GET|HEAD)$" chain
SecFilterSelective HTTP_Content-Type "!(^application/x-www-form-
urlencoded$|^multipart/form-data;)"
SecFilterSelective HTTP_Transfer-Encoding "!"
# Akzeptiere Nicht GET oder HEAD Anfragen im Body
SecFilterSelective REQUEST_METHOD "^(GET|HEAD)$" chain
SecFilterSelective HTTP_Content-Length "!"
# Erfordert eine Content-Length Angabe bei Post
SecFilterSelective REQUEST_METHOD "^POST$" chain
SecFilterSelective HTTP_Content-Length "^$"
</IfModule>
```

Vergessen Sie zum Schluss der Anpassung bitte nicht, den Apache neu zu starten!



22 Perl erweitern

Was für PHP PEAR und PECL sind, ist für Perl das sog. *Comprehensive Perl Archive Network* (abgekürzt CPAN). CPAN ist eine Datenbank mit allen offiziellen Perl-Modulen, auf die mit relativ einfachen und bequemen Befehlen zugegriffen werden kann.

Auf der Homepage des CPAN unter <http://www.cpan.org> finden Sie eine Übersicht über alle verfügbaren Perl-Module.

22.1 Windows

Sie brauchen in jedem Fall das *XAMPP Perl-Add-On*. Das Add-On unter Windows basiert auf *ActivePerl* von *ActiveState* (<http://www.activestate.com>). In dem Paket ist sowohl CPAN als auch PPM integriert.

22.1.1 CPAN

CPAN wird über die Eingabeaufforderung ausgeführt, und zwar mit folgender Anweisung:

```
<xampp>\perl\bin\cpan
```

Wir wollen hier nicht um den heißen Brei »herumschreiben«: CPAN-Module unter Windows zu kompilieren ist nicht so einfach. Sie benötigen in jedem Fall einen Compiler. In der Regel sollte es sogar VC++ von Microsoft sein. Und Sie brauchen noch abhängig von dem zu installierenden Paket weitere GNU-Tools. Sie erhalten das GNU-Bundle für Windows übrigens unter <http://unxutils.sourceforge.net/>.

Aber auch eine gute Entwicklerplattform ist leider noch kein Garant für ein Gelingen. Deshalb raten wir Ihnen zu der *PPM-Verwaltung* von *ActivePerl*. Das Aufsetzen der binären PPM-Pakete benötigt weder Compiler noch weiteres Zubehör und wird meist von Erfolg gekrönt sein. Alle anderen Programmierer mit viel Zeit und Muße lesen bitte den CPAN-Abschnitt für Linux. Die angewandte Syntax ist unter Linux und Windows identisch. Aber dass Sie uns bitte nicht missverstehen: Unter Linux ist CPAN eine starke und unverzichtbare Paketverwaltung für Perl!

```

C:\WINNT\System32\cmd.exe - cpan
confirmation ('ask'), or just ignore them ('ignore'). Please set your
policy to one of the three values.

Policy on building prerequisites (follow, ask or ignore)? [ask] ignore

The CPAN module will need a few external programs to work properly.
Please correct me, if I guess the wrong path for a program. Don't
panic if you do not have some of them, just press ENTER for those. To
disable the use of a download program, you can type a space followed
by ENTER.

Where is your gzip program? [D:\Programme\MiVisualSt\Uc7\bin\gzip.EXE]
Where is your tar program? [D:\Programme\MiVisualSt\Uc7\bin\tar.EXE]
Where is your unzip program? [C:\oracle\ora90\bin\unzip.EXE]
Where is your make program? [D:\Programme\Microsoft Platform SDK\Bin\nmake.EXE]

Where is your lynx program? [D:\Programme\MiVisualSt\Uc7\bin\lynx.EXE]
Where is your wget program? [D:\Programme\MiVisualSt\Uc7\bin\wget.EXE]
Where is your ncftpget program? [D:\Programme\MiVisualSt\Uc7\bin\ncftpget.EXE]
Where is your ftp program? [C:\WINNT\system32\ftp.EXE]
Where is your gpg program? [D:\Programme\MiVisualSt\Uc7\bin\gpg.EXE]
What is your favorite pager program? [D:\Programme\MiVisualSt\Uc7\bin\less.EXE]

What is your favorite shell? cmd_

```

Abbildung 22.1: Schon bei der Aktivierung fragt CPAN nach allen Entwicklerprogrammen.

```

C:\WINNT\System32\cmd.exe - cpan

C:\xampp\perl\bin\perl.exe "-MExtUtils::Command::MM" "-e" "test_harness(
0, 'blib\lib', 'blib\arch') " t\01das.t
t\01das....Can't locate Bio/Root/Root.pm in @INC (@INC contains: . ./blib/lib ..
C:\xampp\perl\bin\..cpan\build\Bio-Das-1.02\blib\lib C:\xampp\perl\bin\..cpan\buil
ld\Bio-Das-1.02\blib\arch C:\xampp\perl\lib C:\xampp\perl\site\lib) at blib/lib/
Bio/Das.pm line 7.
BEGIN failed--compilation aborted at blib/lib/Bio/Das.pm line 7.
Compilation failed in require at t\01das.t line 18.
BEGIN failed--compilation aborted at t\01das.t line 18.
t\01das....dubious
Test returned status 2 (wstat 512, 0x200)
DIED. FAILED tests 1-36
Failed 36/36 tests, 0.00% okay
Failed Test Stat Wstat Total Fail Failed
-----
t\01das.t      2    512    36    72 200.00%  1-36
Failed 1/1 test scripts, 0.00% okay. 36/36 subtests failed, 0.00% okay.
NMAKE : fatal error U1077: 'C:\xampp\perl\bin\perl.exe' : return code '0x2'
Stop.
nmake test -- NOT OK
Running make install
make test had returned bad status, won't install without force

cpan>

```

Abbildung 22.2: Bio::Das-Installation mit dem Hinweis dubious unter Windows abgebrochen.



Hinweis

Leider gab es Probleme bei dem Einsatz von CPAN und PPM mit dem XAMPP Perl-Add-On in der Version 5.8.7-2.2.0. Diese Probleme konnten schließlich mit den nachfolgenden Versionen behoben werden.

22.1.2 PPM

Das sog. *Programmer's Package Manager (PPM)* von *ActivePerl* ist unter Windows eindeutig die bequemere und erfolgreichere Verwaltung zum Laden und Installieren von zusätzlichen Perl-Modulen. Eine kurze Einführung in die Praxis vom PPM mit allen wichtigen Anweisungen folgt nun in diesem Abschnitt.

Den PPM führen Sie mit folgender Eingabe über die Eingabeaufforderung aus:

```
<xampp>\perl\bin\ppm
```

```
C:\WINNT\System32\cmd.exe - C:\xampp\perl\bin\ppm
40. Apache-Reload [0.07] Apache-Reload
41. Apache-Request-Redirect [0.05] Apache-Request-Redirect
42. Apache-Scoreboard [2.01] Apache-Scoreboard
43. Apache-SimpleReplace [0.06] Apache-SimpleReplace
44. Apache-SMTP [0.01] Apache-SMTP
45. Apache-Sybase-CTlib [1.02] Apache-Sybase-CTlib
46. Apache-Test-CookieEmulator [0.06] Apache-Test-CookieEmulator
47. Apache-TieBucketBrigade [0.03] Apache-TieBucketBrigade
48. Apache-TransLDAP [0.20] Apache-TransLDAP
49. Apache-UMonitor [1.2] Apache-UMonitor
50. Apache-WebSNMP [0.11] Apache-WebSNMP
51. ApacheDBI [0.88] ApacheDBI
52. ApacheMysql [0.3] ApacheMysql
53. Catalyst-Engine-Apache [1.06] Catalyst-Engine-Apache
54. Config-ApacheExtended [1.16] Config-ApacheExtended
55. Config-ApacheFormat [1.2] Config-ApacheFormat
56. Maypole-Plugin-Config-Apache [0.21] Maypole-Plugin-Config-Apache
ppm> install 1-25
Package 1:
=====
Install 'Apache-Admin-Config' version 0.92 in ActivePerl 5.8.7.815.
=====
Downloaded 16567 bytes.
Extracting 5/5: blib/arch/auto/Apache/Admin/Config/.exists
```

Abbildung 22.3: Über PPM die Pakete 1 bis einschließlich 25 installieren

Über folgenden Befehl lassen Sie sich alle Optionen von PPM anzeigen:

```
ppm>help
```

Folgende Eingabe listet Ihnen alle bereits installierten Module auf Ihrem System auf:

```
ppm>query *
```

Hinweis



Die query-Funktion lief in dem zur Drucklegung dieses Buches aktuellen XAMPP Perl-Add-On (5.8.7-2.2.0-pl1) nicht. Und wir wissen auch nicht, warum. Aber wir arbeiten an einer Lösung!

Mit diesem Befehl suchen Sie nach allen Paketen mit XML-Bezug auf dem *PPM-Server*:

```
ppm>search XML
```

So installieren Sie ein bestimmtes Paket, hier *libxml-perl*:

```
ppm>install libxml-perl
```

Die Pakete 1 bis 8 plus 12 werden in Bezug auf die aktuelle Ausgabe über *search* installiert:

```
ppm>install 1-8,12
```

Mit folgendem Befehl aktualisieren Sie ein vorhandenes Paket:

```
ppm>upgrade libxml-perl
```

So löschen Sie ein vorhandenes Paket:

```
ppm>remove libxml-perl
```

Den PPM beenden Sie mit dem *quit*-Befehl:

```
ppm>quit
```

22.2 Linux

Am bequemsten funktioniert die Installation von zusätzlichen Perl-Modulen unter Linux in der sog. CPAN-Shell. Bitte achten Sie darauf, dass Sie einen C-Compiler (GCC) und das *make*-Paket auf Ihrem Linux-System installiert haben. Ohne diese beiden werden Sie kaum in der Lage sein, CPAN-Module nachzuinstallieren.

Bevor Sie aber in die CPAN-Shell wechseln, stellen Sie sicher, dass Sie als *root* angemeldet sind. Rufen Sie erst danach mit folgendem Befehl die CPAN-Shell auf:

```
/opt/lampp/bin/cpan
```

Die CPAN-Shell wird sich nun mit der folgenden Meldung vorstellen und mit dem CPAN-Prompt anzeigen, dass sie auf weitere Befehle wartet:

```
cpan shell -- CPAN exploration and modules installation (v1.7601)
cpan>
```

An dieser Stelle können Sie nun die CPAN-Shell mit dem *install*-Befehl anweisen, weitere Module zu installieren. Voraussetzung ist allerdings, dass Ihr Linux-System mit dem Internet verbunden ist, um auf die CPAN-Server zugreifen zu können.

In dem folgenden Beispiel können Sie sehen, wie das CPAN-Modul `Bio::Das` mit dem `install`-Befehl installiert wird:

```
cpan> install Bio::Das
CPAN: Storable loaded ok
Going to read /root/.cpan/Metadata
  Database was generated on Sun, 27 Jun 2004 22:37:11 GMT
CPAN: Net::FTP loaded ok
Fetching with Net::FTP:
...
```

Jetzt werden bildschirmseitenweise Ausgaben an Ihnen vorbeischnitten. Sie zeigen an, dass Daten aus dem Internet heruntergeladen, C-Programme übersetzt, getestet und installiert werden. Evtl. erscheint auch folgende Meldung:

```
---- Unsatisfied dependencies detected during [L/LD/LDS/Bio-Das-1.02.tar.gz] ----
  Compress::Zlib
  Bio::Root::Root
  HTML::Parser
  LWP
Shall I follow them and prepend them to the queue
of modules we are processing right now? [yes] yes
```

Die CPAN-Shell hat festgestellt, dass zur Installation von `Bio::Das` weitere CPAN-Module notwendig sind, und fragt Sie nun, ob diese auch installiert werden sollen. Eigentlich haben Sie keine andere Wahl und darum geben Sie `yes` als Antwort an.

Danach folgen wieder bildschirmseitenweise Ausgaben, gelegentlich auch solche:

```
t/08keeperr.....ok
t/09trace.....ok
t/10examp.....ok
t/11fetch.....ok
t/14utf8.....ok
```

Hier testet sich das CPAN-Modul selbst. Ab und zu sieht man auch Folgendes:

```
t/zvpp_40profile.....skipped
```

In diesem Fall konnte ein Test aus irgendwelchen Gründen nicht durchgeführt werden. Das ist normal und stellt in der Regel kein Problem da.

Bei manchen Tests werden Sie sogar befragt:

```
Generated sub tests. go make show_tests to see available subtests
Do you want to run the BioDBGFF tests (requires a mysql database) y/n [n] n
```

Hier werden Sie gefragt, ob ein bestimmter Test, für den allerdings eine MySQL-Datenbank benötigt wird, durchgeführt werden soll. In 99 % aller Fälle können Sie hier einfach mit `n` antworten und den Test überspringen.

Irgendwann sollten Sie aber folgende Zeilen auf Ihrem Bildschirm erblicken:

```
...
Installing /opt/lampp/man/man3/Bio::Das::ProServer::SourceAdaptor::simple.3
Writing /opt/lampp/lib/perl5/site_perl/5.8.7/i686-linux/auto/Bio/Das/.packlist
Appending installation info to /opt/lampp/lib/perl5/5.8.7/i686-linux/perllocal.pod
  /usr/bin/make install -- OK
cpan>
```

Endet die Ausgabe mit diesen Zeilen, dann ist die Installation des zusätzlichen CPAN-Moduls erfolgreich abgeschlossen und Sie können die CPAN-Shell mit dem `quit`-Befehl beenden:

```
cpan> quit
```

Nun können Sie in Ihrem Perl-Programm das installierte CPAN-Modul mit `use` einbinden.



23 XAMPP und Java

23.1 Apache Tomcat

Bei *Apache Tomcat* handelt es sich um einen HTTP-Server speziell für die Ausführung von Java-basierten *Servlets* und *Java Server Pages* (JSP). Er wurde im Rahmen des Jakarta-Projekts der *Apache Software Foundation* erstellt. Ab der Version 4 erhielt der Tomcat das Synonym *Catalina*. Auch dieser Server untersteht der recht großzügigen Apache-Lizenz. Die Homepage lautet <http://tomcat.apache.org>.

Den Tomcat sollten Sie aber nicht als konkurrierendes, sondern als ergänzendes Produkt zum Apache HTTPD verstehen. Als HTTP-Server ist er der Flexibilität des großen Bruders Apache nicht gewachsen. Und das war auch niemals das Ziel. Vielmehr bestand der Zweck darin, einen stabilen und schnellen Server für die Ausführung von Java-Programmen zur Verfügung zu stellen. Sie können die Funktion des Tomcat durchaus mit der eines PHP-Parsers vergleichen, der neben der eigentlichen Ausführung den Output einer Anwendung – nennen wir es etwas salopp – »HTTP-fein macht«. Natürlich hinkt dieser Vergleich, da man Java und PHP nicht wirklich miteinander vergleichen kann.

Exkurs



Wussten Sie, dass der *Tomcat* schneller die HTTP-Anfragen bedient als sein Webserver-Bruder *Apache*? Der Grund hierfür liegt im reichhaltigen Funktionsumfang des Apache. Würde man ihn all seiner wertvollen Features beschneiden, danach für das System frisch übersetzt ins Rennen ziehen lassen, sähe das Ergebnis auch ganz anders aus.

Für eine gelungene Symbiose zwischen Apache und Tomcat erfanden die Entwickler das *Apache JServ-Protokoll*. Es dient der Kommunikation zwischen den beiden Servern und findet in dem Apache-Modul *mod_jk* seine Umsetzung. Mit diesem sog. *Tomcat Connector* ist es möglich, Anfragen auf Java-basierte Anwendungen über den Apache auf dem *Standardport 80* abzuwickeln. Dabei dient der Apache nur als Vermittler. Den eigentlichen Tomcat erreichen Sie über seinen *http-Port*, in der Regel den *Port 8080*.

Der Apache Tomcat ist nicht im XAMPP-Basispaket enthalten. Dies liegt zum einen an lizenzrechtlichen Problemen mit Java Sun und zum anderen an der Dimension des Pakets. Für Windows-Anwender gibt es aber ein Add-On mit Tomcat und *mod_jk*, zugeschnitten auf das jeweils aktuelle XAMPP-Basispaket. Für die Installation dieses Add-Ons müssen Sie zuvor Java JDK von SUN installiert haben. Aber keine Angst! Die Installation von Java, Tomcat und seinem Connector (*mod_jk*) unter Linux und Windows stellen wir Ihnen nun vor.

23.1.1 Installation unter Linux

Im Folgenden erfahren Sie, wie Sie XAMPP für Linux mit Tomcat plus Connector (*mod_jk*) aufrüsten.

Java JDK installieren

1. Laden Sie das JDK für Linux als *Linux self-extracting file*, in unserem Beispiel *jdk-1_5_0_06-nb-4_1-linux-m1.bin*, von der Sun-Website herunter:

<http://java.sun.com/j2se/1.5.0/download.jsp>

2. Starten Sie die Installation in einer Shell:

```
sh ./jdk-1_5_0_06-nb-4_1-linux-m1.bin
```

Ein grafischer Installer führt Sie nun durch die Installation. Installieren Sie die Komponenten *JDK* und *JavaBeans* in ein Verzeichnis (z.B. *java*) in ihrem Heimverzeichnis.

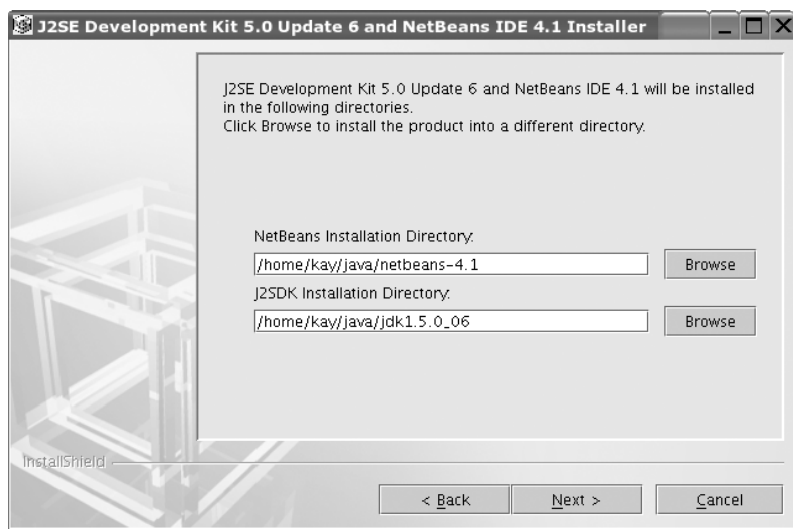


Abbildung 23.1: Der J2SE Installer von Sun Microsystems unter Linux

3. Für die folgenden Schritte brauchen Sie *root*-Rechte. Melden Sie sich entweder neu an oder verwenden Sie den *su*-Befehl.
4. Verschieben Sie danach das komplette *java*-Verzeichnis beispielsweise nach */usr*.

```
mv -R $HOME/java /usr
```

5. Setzen Sie einen symbolischen Link.
Optional können Sie auch zum *jdk*-Verzeichnis einen symbolischen Link setzen.

```
cd /usr
ln -s /usr/java/jdk1.5.0_06/ jdk
```

6. Setzen Sie neue Umgebungsvariablen.
Die SUSE-Distribution definiert diese Variablen für jeden Benutzer in der Datei *%HOME/.profile*. Unter Debian können sie auch zentral in der Datei */etc/.profile* angelegt werden. Sie ergänzen das Profil durch die folgenden Zeilen:

```
JAVA_HOME="/usr/jdk"
CLASSPATH=/usr/jdk/lib:/usr/tomcat/server/lib:/usr/tomcat/server/classes
PATH="/usr/local/bin:/usr/bin:/bin:/usr/bin/X11:/usr/games:$JAVA_HOME/bin"
export PATH JAVA_HOME CLASSPATH
```

7. Testen Sie die Installation.

```
java -version
```



Abbildung 23.2: Test der *PATH*-Variablen für Java

Tomcat installieren

Für dieses Beispiel verwenden wir den zum Zeitpunkt der Drucklegung dieses Buches aktuellen *Tomcat 5.5.15*. Das binäre Paket *apache-tomcat-5.5.15.tar.gz* bezogen wir von der Seite <http://tomcat.apache.org>. Den Quellcode wollten wir nicht kompilieren. Für die Übersetzung des Tomcat Connectors brauchen Sie nämlich den Tomcat-Quellcode nicht. Als Stammverzeichnis (*\$CATALINA_HOME*) für den Tomcat wählen wir hier optional */usr/tomcat*.

Referenz: <http://tomcat.apache.org/tomcat-5.0-doc/>

1. Extrahieren Sie die Datei als *Root*:

```
su
tar xvzf apache-tomcat-5.5.15.tar.gz -C /usr/local
```

2. Weisen Sie die Benutzerrechte analog zum Apache-Daemon zu:

```
chown -R nobody.root /usr/local/apache-tomcat-5.5.1
```

3. Richten Sie einen symbolischen Link ein:

```
cd /usr
ln -s /usr/local/apache-tomcat-5.5.1/tomcat
```

4. Rüsten Sie die Datei `catalina.sh` mit Umgebungsvariablen aus:

```
vi /usr/tomcat/bin/catalina.sh
```

5. Fügen Sie schließlich die folgenden Zeilen nach dem Kommentar ein:

```
JAVA_HOME=/usr/jdk
CATALINA_HOME=/usr/tomcat
DAEMON_HOME=/usr/tomcat
TOMCAT_USER=nobody
JRE_HOME=/usr/jdk/jre
```

6. Optional können Sie natürlich auch Umgebungsvariablen setzen:

```
export JAVA_HOME=/usr/jdk
export CATALINA_HOME=/usr/tomcat
export DAEMON_HOME=/usr/tomcat
export TOMCAT_USER=nobody
export JRE_HOME=/usr/jdk/jre
```

7. Testen Sie die Installation:

Starten: `/usr/tomcat/catalina.sh start`

Stoppen: `/usr/tomcat/catalina.sh stop`

Rufen Sie nach dem Start die lokale URL `http://localhost:8080` über den Browser auf.
Bei erfolgreicher Einrichtung sehen Sie nun die Startseite Ihres Tomcat-Servers.

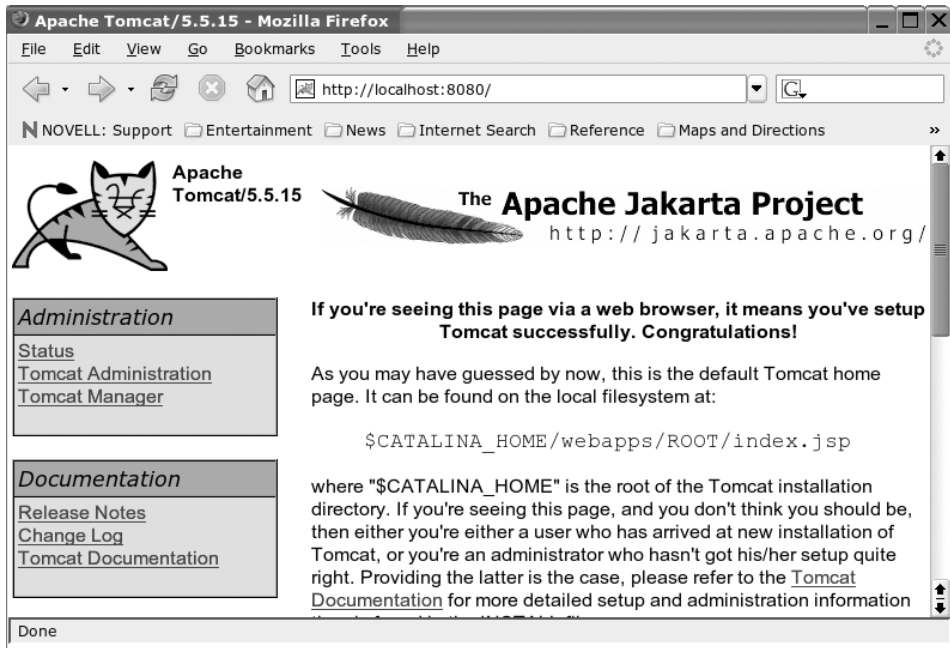


Abbildung 23.3: Es funktioniert: der Tomcat auf Port 8080.

Den Tomcat Connector übersetzen und installieren

Um den Connector unter Linux zu übersetzen, müssen Sie zuvor das XAMPP für Linux-Entwicklungs-Paket installiert haben (für nähere Einzelheiten zur Installation siehe Kapitel 21, »Apache erweitern«). Den *Apache Jakarta Tomcat Connector* selbst erhalten Sie über die Homepage:

<http://tomcat.apache.org/connectors-doc/>

1. Extrahieren Sie den Connector.

```
tar xvfz jakarta-tomcat-connectors-1.2.15-src.tar.gz
```

2. Übersetzen Sie den Connector mit make.

```
su
cd jakarta-tomcat-connectors-1.2.15-src/jk/native
./configure --with-apxs=/opt/lampp/bin/apxs --with-java-home=${JAVA_HOME} \
--with-java-platform=2 --enable-jni
make
make install
```

3. Laden Sie das Apache-Modul und fügen Sie die folgende Zeile in die `/opt/lampp/etc/httpd.conf` hinzu:

```
LoadModule jk_module modules/mod_jk.so
```

4. Erstellen Sie die Datei `workers.properties`.
In der Datei `workers.properties` wird der Connector (*ajp13*) definiert.

```
touch /opt/lampp/etc/workers.properties  
chown -R nobody.root /opt/lampp/etc/workers.properties
```

Die `workers.properties` könnte den folgenden Inhalt haben:

```
worker.list = worker1  
worker.worker1.port = 8009  
worker.worker1.host = localhost  
worker.worker1.type = ajp13
```

5. Erstellen Sie die Datei `java.conf` und binden Sie sie in die `httpd.conf` ein.
Die `java.conf` definiert dem Apache die Bedingungen für eine Weitergabe an den *ajp13*-Connector. Erstellen Sie diese Datei und weisen Sie der Konfiguration die korrekten Benutzerrechte zu.

```
touch /opt/lampp/etc/extra/java.conf  
chown -R nobody.root /opt/lampp/etc/extra/java.conf
```

In diesem Fall sollen generell der gesamte Inhalt im Verzeichnis `webapps/servlets-examples` und alle `.jsp`-Dateien im `webapps/jsp-examples` über *ajp13* ausgeführt werden. Der Inhalt der `java.conf` sähe in diesem Falle so aus:

```
<IfModule mod_jk.c>  
JkWorkersFile /opt/lampp/etc/workers.properties  
JkMount /servlets-examples/* worker1  
JkMount /jsp-examples/*.jsp worker1  
</IfModule>
```

Die Einbindung am Ende der `/opt/lampp/etc/httpd.conf` geschieht zum Abschluss ganz klassisch über die folgende Zeile:

```
Include conf/extra/java.conf
```

6. Starten Sie den Apache neu, um die Änderungen zu laden.
Ein weiteres Beispiel, wie Sie den Apache und den Connector konfigurieren können, finden Sie übrigens weiter unten im Abschnitt »Tomcat und Connector individualisieren«.

23.1.2 Installation unter Windows

Java JDK installieren

Den entsprechenden Java-Installer erhalten Sie auf der Sun Microsystems-Homepage:

<http://java.sun.com/j2se/1.5.0/download.jsp>

Hierbei ist zu beachten, dass Sie das richtige Paket herunterladen. Sie benötigen das *J2SE Development Kit (JDK)*. Für dieses Beispiel verwenden wir die `jdk-1_5_0_06-nb-4_1-win-ml.exe` (Java 5) als Basis.

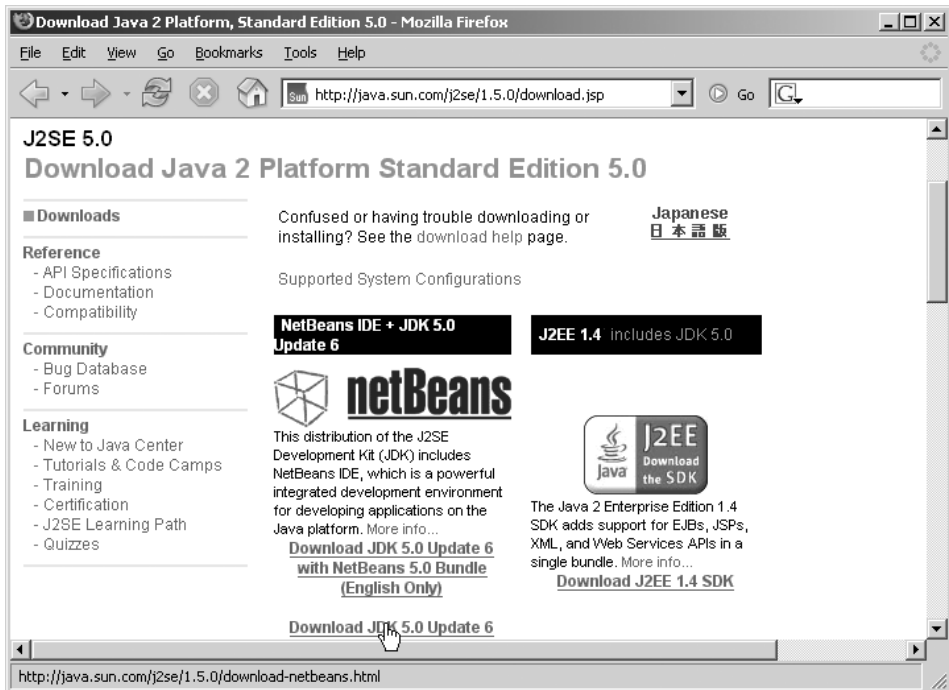


Abbildung 23.4: Richtig! Sie benötigen das JDK 5 von Sun.

Die menügeführte Installation stellt für Sie sicherlich kein Problem dar. Wichtig ist nach der Installation die Überprüfung der Umgebungsvariablen. Denn der Tomcat-Server sucht insbesondere die Werte `JAVA_HOME` mit dem absoluten Pfad zum JDK-Verzeichnis und `CLASSPATH` mit dem Pfad zur Bibliothek `tools.jar`. Sollten diese Variablen fehlen, tragen Sie diese Werte unter Windows XP unter SYSTEMSTEUERUNG > SYSTEM > ERWEITERT > UMGEBUNGSVARIABLEN nach.

```
JAVA_HOME=C:\Java\jdk
CLASSPATH=%JAVA_HOME%\lib\tools.jar
PATH=%JAVA_HOME%\bin;%path%
```

Unter Windows 98/Me editieren Sie hierfür die `autoexec.bat`. Beispiel:

```
set JAVA_HOME=C:\Java\j2sdk
set CLASSPATH= %JAVA_HOME%\lib\tools.jar
set path=%JAVA_HOME%\bin;%path%
```

Tomcat Add-On für XAMPP installieren

Das *Tomcat-XAMPP-Add-On* enthält bereits den Tomcat Connector (*mod_jk*) für den entsprechenden Apache im XAMPP-Basispaket. Seit der Umstellung der Nomenklatur erkennen Sie die Kompatibilität sofort an dem Dateinamen. Das Schema lautet:

`xampp-tomcat-addon-[Tomcat-Version]-[Version von Apache-Mod_jk]`

Ein Beispiel:

`xampp-tomcat-addon-5.5.15-2.2.0-installer.exe`

In diesem Fall hätten Sie nun das Installer-Add-On mit Tomcat 5.5.15 sowie eine *mod_jk*-Kompilation für den Apache 2.2.0. Besitzen Sie XAMPP mit dem Apache 2.2.0, können Sie dieses Add-On mit dem geeigneten JDK sorglos verwenden. Die Version des passenden JDK finden Sie übrigens in der Beschreibung des Tomcat Add-Ons. Als Einschränkung gilt aber wie immer, dass Sie Add-Ons mit einem Installer nur auf XAMPP-Basispakete mit Installer aufsetzen können. Besitzer eines Basispakets ohne Installer verwenden dagegen das ZIP-Archiv oder das selbstextrahierende 7-zip-Exe-Paket. Weiterführende Informationen zur Erweiterung von XAMPP und den Add-Ons erhalten Sie in Kapitel 2, »Installation«.

Hinweis



Seit der Version Tomcat Add-On 5.5.15 wurde *mod_jk2* auf *mod_jk1* umgestellt. Grund: Die Entwicklung von *mod_jk2* scheint eingestellt worden zu sein.

Vorgehen mit XAMPP als Installer-Version

Laden Sie das geeignete *Add-On* als *Installer* von der Website <http://www.apachefriends.org> herunter und führen Sie die menügeführte Installation aus. Alle weiteren Anpassungen werden damit automatisch durchgeführt.

Vorgehen mit XAMPP ohne Installer

Laden Sie das Add-On als *ZIP* bzw. *7-zip*-Paket herunter und extrahieren Sie es direkt in das XAMPP-Stammverzeichnis. Gegebenenfalls müssen dabei XAMPP-interne Dateien überschrieben werden.

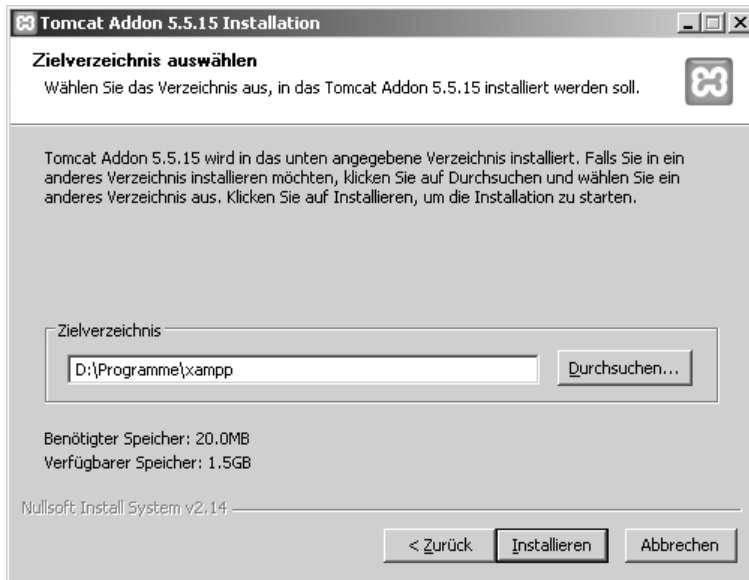


Abbildung 23.5: Das Tomcat Add-On 5.5.15 für den XAMPP-Installer

Zum Schluss initialisieren Sie das Add-On über die `<xampp>\setup_xampp.bat`. Dabei durchsucht ein PHP-Skript Ihre Registry nach einem `JavaHome`-Schlüsselwert mit dem Pfad zu Ihrem aktuellen JDK. Findet das Programm keinen solchen Wert, werden Sie aufgefordert, den Pfad zum Basisverzeichnis von JDK per Tastatur einzutragen.

Schließlich wird die Konfiguration im XAMPP angepasst und die folgenden Dateien werden erstellt:

1. eine Batch-Datei für den Handstart des Tomcat inklusive der notwendigen Umgebungsvariablen:

```
<xampp>\tomcat_start.bat
```

2. eine Batch-Datei für den manuellen Stopp des Tomcat inklusive der notwendigen Umgebungsvariablen:

```
<xampp>\tomcat_stop.bat
```

3. eine Konfigurationsdatei für Apache Tomcat Connector (`mod_jk`):

```
<xampp>\apache\conf\extra\java.conf
```

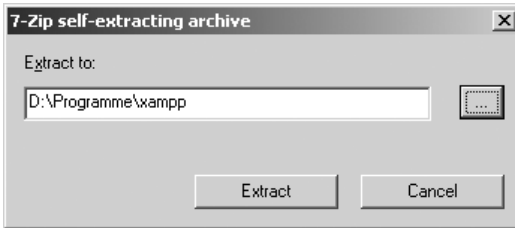


Abbildung 23.6: Das Add-On wird direkt in das XAMPP-Verzeichnis extrahiert



Hinweis

Standardmäßig wird nun das Connector-Modul geladen. Mit der `setup-xampp.bat` können Sie allerdings je nach Wunsch `mod_jk` nachträglich deaktivieren oder erneut aktivieren.

Natürlich finden Sie jetzt auch im XAMPP-Bereich eine Java-Sektion mit Beispielen. Wichtig ist, dass Sie sowohl den Apache- als auch den Tomcat-Server starten, um das komplette Paket zu nutzen.

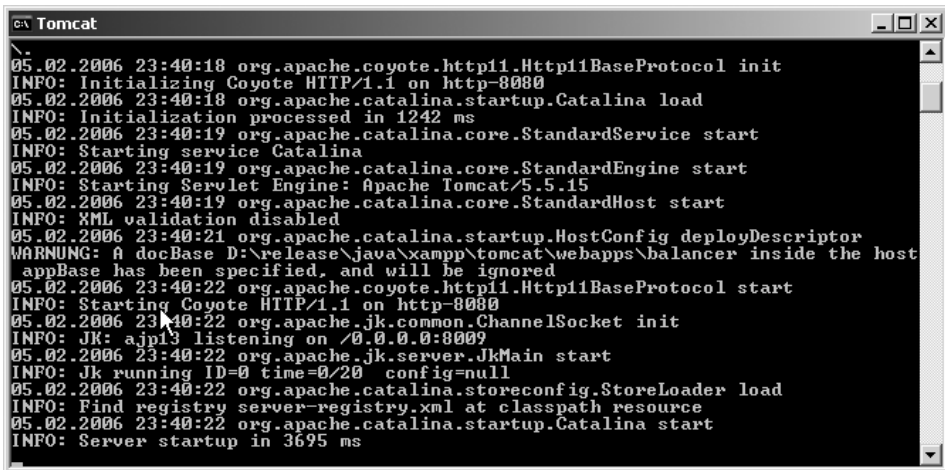


Abbildung 23.7: Der Tomcat-Server unter Windows nach einem »Handstart«.

Tomcat als Dienst installieren

Installation: `<xampp>\tomcat\tomcat_service_install.bat`

Deinstallation: `<xampp>\tomcat\tomcat_service_uninstall.bat`

Die Systemvariable `CATALINA_HOME` (der absolute Pfad zum Tomcat-Server) sowie die bereits angesprochene Umgebungsvariable `JAVA_HOME` und ggf. auch `CLASSPATH` müssen zuvor in Ihrem System fest eingerichtet sein.

23.1.3 Tomcat und Connector individualisieren

In diesem Abschnitt erhalten Sie Tipps zur individuellen Konfiguration des Tomcat und seines Connectors (*mod_jk*).

Die wesentlichen Konfigurationsdateien des Tomcat Connectors lauten:

Windows: `<xampp>\apache\extra\java.conf`
 `<xampp>\tomcat\conf\workers.properties`
 Linux: `/opt/lampp/etc/extra/java.conf`
 `/opt/lampp/etc/workers.properties`
 (oder) `$CATALINA_HOME/conf/workers.properties`

Die wesentliche Konfigurationsdatei des Tomcat-Servers heißt:

Windows: `<xampp>\tomcat\conf\server.xml`
 Linux: `$CATALINA_HOME/conf/server.xml`

Und das Dokumentenverzeichnis des Tomcat-Servers befindet sich in:

Windows: `<xampp>\tomcat\webapps`
 Linux: `$CATALINA_HOME/webapps`

Ein neues Webapps-Verzeichnis definieren

Sie wollen ein neues *Webapps-Unterverzeichnis* für *Java Server Pages* und *Servlets* im Tomcat einrichten und über den Connector auf Port 80 verfügbar machen. In diesem Beispiel wird ein neuer Webapps-Ordner `jhome` erstellt und initialisiert.

1. Legen Sie ein neues Verzeichnis im Webapps-Bereich des Tomcat an.

Windows:

```
mkdir xampp\tomcat\webapps\jhome
```

Linux:

```
su
mkdir $CATALINA_HOME/webapps/jhome
```

2. Kopieren Sie den Ordner `servlets-examples/WEB-INF` in das neue `jhome`-Verzeichnis:

```
cp -R $CATALINA_HOME/webapps/servlets-examples/WEB-INF \
$CATALINA_HOME/webapps/jhome
```

Sie können bei Bedarf die `webapps/WEB-INF/web.xml` Ihren Bedürfnissen anpassen. Wichtig ist hierbei, dass Sie Ihre neu erstellten Servlet-Klassen im `jhome`-Bereich in das Unterverzeichnis `WEB-INF/classes` speichern.

3. Legen Sie die Verzeichnisrechte fest (nur unter Linux):

```
chown -R nobody.root $CATALINA_HOME/webapps/jhome
```

4. Testen Sie das neue Webapps-Verzeichnis.

Hierfür rufen Sie zu Beginn die `HelloWorldExample`-Klasse auf:

`http://localhost:8080/jhome/servlet/HelloWorldExample`

5. Danach erstellen Sie ein einfaches `HelloWelt.jsp` direkt im `jhome`-Ordner. Linux-Benutzer sollten das neue Skript ggf. wieder mit `chown` bearbeiten.

```
<html>
<body>
<%
    out.println("Hallo JSP Welt");
%>
</body>
</html>
```

Wenn jetzt alles funktioniert hat, können Sie sich der Anpassung des Tomcat Connectors widmen.

Neues `jhome`-Verzeichnis über den Connector »mounten«

1. Öffnen Sie die Konfigurationsdatei des Connectors (z. B. `java.conf` unter Windows). Linux:

```
/opt/lampp/etc/extra/java.conf
```

Windows:

```
<xampp>\apache\conf\extra\java.conf
```

2. Binden Sie die neue `JkMount`-Anweisung ein.

Suchen Sie hierfür den Bereich der `JkMount`-Anweisungen in der `java.conf` auf und fügen Sie folgende Zeile ein:

```
JkMount /jhome/* ajp13
```

Damit wird der Inhalt des Verzeichnisses `jhome` über das Vermittlungsprotokoll `ajp13` zum Tomcat durchgereicht.

3. Starten Sie den Apache und ggf. auch den Tomcat neu.

4. Testen Sie nun den `JkMount` über die beiden bereits eingerichteten Java-Programme:

`http://localhost/jhome/servlet/HelloWorldExample`

`http://localhost/jhome/HalloWelt.jsp`

Hiermit überprüfen Sie die bereits eingerichteten Java-Programme, nun allerdings über den Standardport 80.

23.1.4 Die Administrationsbereiche des Tomcat sichern

Der Tomcat besitzt zwei administrative Bereiche: einen *Admin-Bereich* und einen *Manager-Bereich*. Die Verwaltung erfolgt über authentifizierte Benutzer, die über die entsprechenden Administrationsrechte (d.h. eine Mitgliedschaft in einer oder mehreren sog. Rollen) besitzen.

`http://localhost:8080/admin`

`http://localhost:8080/manager/html`

Konfigurationsdatei: `tomcat-users.xml`

Linux: `$CATALINA_HOME/conf/tomcat-users.xml`

Windows: `<xampp>\tomcat\conf\tomcat-users.xml`

Windows

Im *XAMPP Add-On* bis zur Version *5.5.15-2.2.0* sind ein administrativer Beispielbenutzer *xampp* und das Passwort *xampp* im Klartext enthalten. Dieser Benutzer soll gelöscht bzw. durch einen anderen Administrator mit einem verschlüsselten Kennwort ersetzt werden.

1. Erstellen Sie ein *md5*-Passwort mit der `digest.bat`:

```
<xampp>\tomcat\bin\digest -a md5 meinpasswort
```

Es erscheint folgende Ausgabe:

```
meinpasswort:050e4d72c6214f1a058e518bfd93b579
```

Das *digest*-Programm funktioniert in der Regel nur, wenn die Umgebungsvariable `CATALINA_HOME` festgelegt wurde. Sie können den Wert aber auch direkt in die zweite Zeile der `digest.bat` eintragen. Ein Beispiel:

```
@echo off
set CATALINA_HOME=D:\Programme\xampp\tomcat
```

2. Setzen Sie einen neuen administrativen Benutzer in die `tomcat-users.xml` ein:

Konfigurationsdatei: `<xampp>\tomcat\conf\tomcat-users.xml`

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
```

```

<role rolename="tomcat"/>
<role rolename="role1"/>
<role rolename="manager"/>
<role rolename="admin"/>
<user username="root" password="050e4d72c6214f1a058e518bfd93b579"
roles="admin,manager"/>
</tomcat-users>

```

Der Administrator heißt nun *root* und das Passwort ist verschlüsselt. Nun müssen wir dem Tomcat-Server den entsprechenden Algorithmus der Verschlüsselung bekannt geben.

Das md5-Kennwortverschlüsselung in der server.xml aktivieren

Konfigurationsdatei: `<xampp>\tomcat\conf\server.xml`

1. Suchen Sie folgende Zeilen in der `server.xml`:

```

<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
resourceName="UserDatabase"/>

```

2. Ergänzen Sie den Container durch die Zeile `digest="md5"`:

```

<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
digest="md5"
resourceName="UserDatabase"/>

```

3. Starten Sie nach der Prozedur den Tomcat-Server neu.

In diesem Beispiel könnte sich der Benutzer *root* mit dem Kennwort *meinpasswort* einloggen. Aber bitte! Das ist doch nur ein Beispiel!

Linux

Die Prozedur ist identisch mit der unter Windows. Deshalb halten wir uns hier kurz.

1. Erstellen Sie das *md5*-Passwort mit der `digest.sh`.

```
$CATALINA_HOME/bin/digest.sh -a md5 meinpasswort
```

Es erscheint folgende Ausgabe:

```
meinpasswort:050e4d72c6214f1a058e518bfd93b579
```

2. Setzen Sie einen neuen administrativen Benutzer in die `tomcat-users.xml` ein:

Konfigurationsdatei: `$CATALINA_HOME/conf/tomcat-users.xml`

```

<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <role rolename="manager"/>
  <role rolename="admin"/>

```

```
<user username="root" password="050e4d72c6214f1a058e518bfd93b579"
roles="admin,manager"/>
</tomcat-users>
```

3. Aktivieren Sie die *md5*-Kennwortverschlüsselung in der *server.xml*.

Konfigurationsdatei: `$CATALINA_HOME/conf/server.xml`

Suchen Sie folgende Zeilen in der *server.xml*:

```
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
resourceName="UserDatabase"/>
```

Ergänzen Sie den Container durch die Zeile `digest="md5"`:

```
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
digest="md5"
resourceName="UserDatabase"/>
```

4. Zum Schluss starten Sie den Tomcat-Server neu.

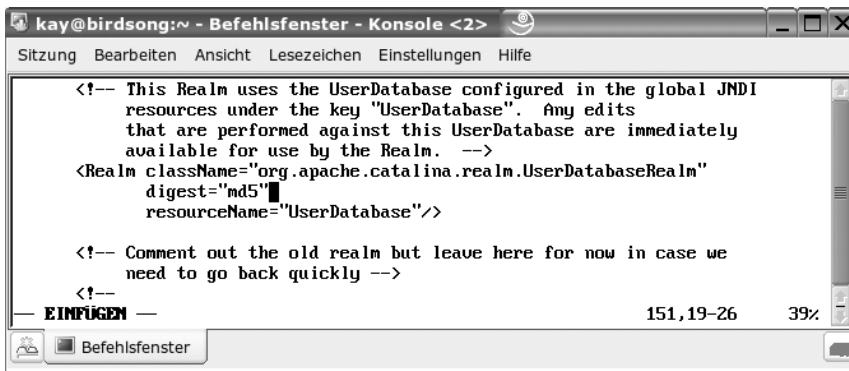


Abbildung 23.8: Die *md5*-Einstellung in der *server.xml* im *vim*

23.2 XML

Zum Abschluss dieses Abschnitts möchten wir noch zwei kurze, aber wichtige Hinweise für die Unterstützung von XML geben.

23.2.1 XML Parser

Java 5 (JDK 1.5 bzw. J2SE 5 von Sun Microsystems) enthält bereits die XML Parser *Xerces* und *Xalan* standardgemäß. Auch die Ausführung über den Tomcat-Server funktioniert in der Regel problemlos.

23.2.2 Cocoon

Cocoon ist Ihnen vielleicht als das *XML Publishing System* der *Apache Group* bekannt. Cocoon wurde als Framework mit einem XML-basierten Datenaustausch mit formatierter XSL-Ausgabe konstruiert. Generell benötigt Cocoon den Tomcat als Servlet-Container – und selbstverständlich *Java SDK*.

Seit der *Version 2.0.4* wird Cocoon nicht mehr als Binärpaket angeboten. Anscheinend gab es Probleme mit der Kompatibilität zu den einzelnen Java-Teilversionen. Das ist auch einer der Gründe, warum das Cocoon XAMPP-Add-On vorübergehend eingestellt wurde. Wenn Sie aber JDK installiert haben, sollten Sie versuchen, das Paket selbst zu kompilieren.

Homepage: <http://cocoon.apache.org>

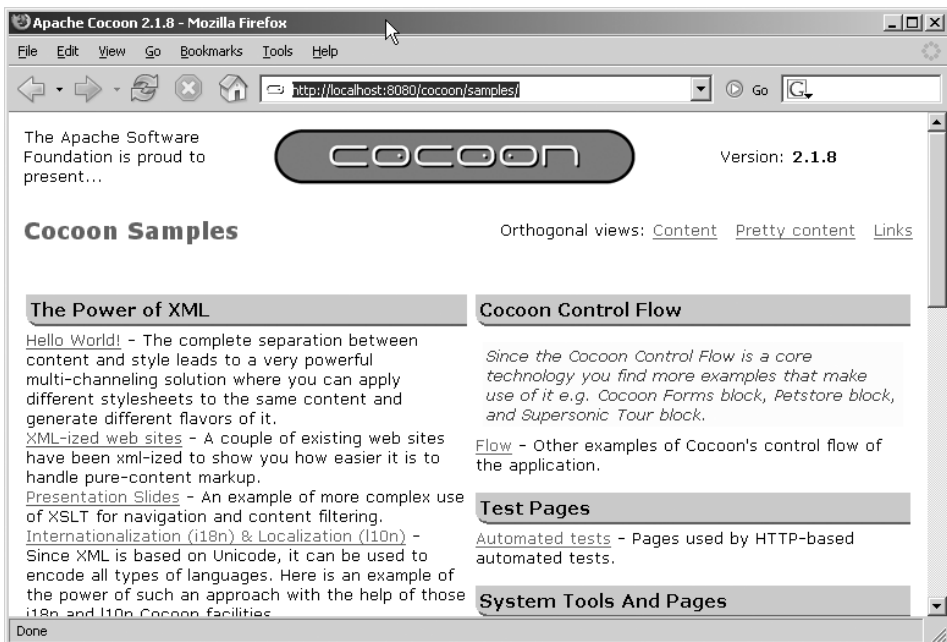


Abbildung 23.9: Cocoon 2.1.8 in Tomcat, hier auf einem Windows-System

Windows

Beispielpaket: `cocoon-2.1.8-src.zip`

1. Entpacken Sie das ZIP-Archiv `cocoon-2.1.8-src.zip`.
2. Starten Sie mit der `cocoon.bat` die Übersetzung:

```
cd cocoon-2.1.8
cocoon.bat
```

3. Kopieren Sie das `build\webapp`-Verzeichnis in den Tomcat:

```
cp -R build\webapp D:\Programme\tomcat\webapps
```

4. Benennen Sie den neuen Ordner in `cocoon` um. Dieser Schritt ist optional. Aus dem Unterordner

```
<xampp>\tomcat\webapps\webapp
```

wird folglich:

```
<xampp>\tomcat\webapps\cocoon
```

5. Starten Sie den Tomcat neu und rufen Sie Cocoon auf:

<http://localhost:8080/cocoon/>

Linux

Beispieldpaket: `cocoon-2.1.8-src.tar.gz`

1. Extrahieren Sie den Quellcode:

```
tar xvzf cocoon-2.1.8-src.tar.gz
```

2. Starten Sie mit der `cocoon.sh` die Übersetzung:

```
cd cocoon-2.1.8
```

```
su
```

```
./cocoon.sh
```

3. Verschieben Sie das `build/webapp`-Verzeichnis als `cocoon` in den Tomcat:

```
mv -R build/webapp $CATALINA_HOME/webapps/cocoon
```

4. Starten Sie den Tomcat neu und rufen Sie Cocoon auf:

<http://localhost:8080/cocoon/>

23.3 Java Server Pages und Datenbanken

23.3.1 MySQL

Für den Zugriff verwenden wir den *JDBC (Java Database Connectivity)* für MySQL. JDBC für Java ist mit dem universellen DBI-Konzept für Perl vergleichbar und eine gute Wahl. Der von uns bevorzugte Treiber heißt *MySQL Connector/J 5.0*, der uns in einer Betaversion vorlag:

<http://www.mysql.com/products/connector/j/>

Die Installation ist übrigens wirklich nicht schwierig. Sie müssen das Archiv `mysql-connector-zip` temporär extrahieren und die Datei `mysql-connector-java-x.y.z-bin.jar` in den `lib`-Ordner im `common`-Verzeichnis Ihres Tomcat-Servers kopieren. Alternativ positionieren Sie den Connector in den `WEB-INF\lib`-Ordner der jeweiligen `webapps`-Verzeichnisse.

Beispiel unter Windows:

```
<xampp>\tomcat\common\lib\mysql-connector-java-5.0.0-beta-bin.jar
```

Beispiel unter Linux:

```
$CATALINA_HOME/common/lib/mysql-connector-java-5.0.0-beta-bin.jar
```

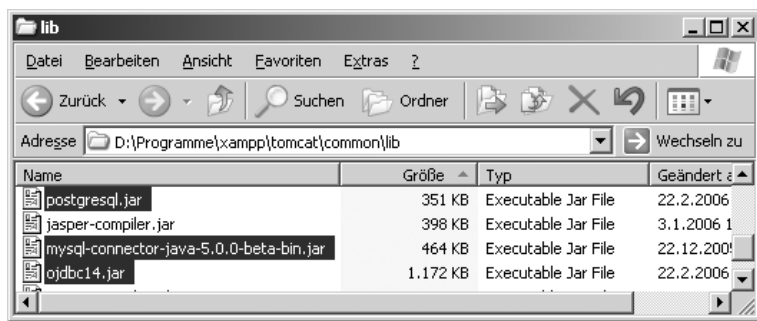


Abbildung 23.10: Die neuen JDBC-Treiber im Tomcat-Server unter Windows

Ein Neustart des Tomcat ist nun unbedingt notwendig! Damit haben wir den neuen JDBC-Treiber geladen und erstellen eine JSP-Datei, die wir direkt in den `jsp-examples`-Ordner legen. Der Zugriff auf den MySQL-Server erfolgt nun in der für JDBC üblichen Art und Weise.

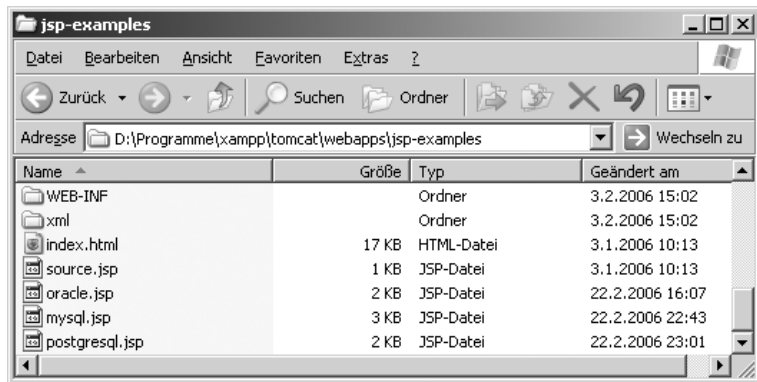


Abbildung 23.11: Die neuen JSP-Programme im `webapps/jsp-examples`-Ordner

Listing 23.1: Zugriff auf MySQL in Java mit JDBC

```

<%@ page import="java.sql.*,java.util.*,javax.naming.*" %>
<%
    String loginUrl = "jdbc:mysql://127.0.0.1/cdcol";
    String loginUser = "kvo";
    String loginPasswd = "passwort";
    Connection dbcon = null;
    try
    {
        Class.forName("com.mysql.jdbc.Driver").newInstance();
        dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
    }
    catch (ClassNotFoundException ex)
    {
        System.err.println("ClassNotFoundException: " + ex.getMessage());
        throw new ServletException("Class not found Error");
    }
%>

```

Nun sollten Sie Ihr JSP noch etwas »aufpeppen«. In dem Beispiel `mysql.jsp` wird die Tabelle `cds` der MySQL-Datenbank `cdcol` mit den SQL-Statements `SELECT`, `INSERT`, `UPDATE` und `DELETE` verwendet.

Listing 23.2: Die `mysql.jsp` mit unterschiedlichen SQL-Statements

```

<html>
<body>
<%@ page import="java.sql.*,java.util.*,javax.naming.*" %>
<h3>CD Musik Datenbank</h3> <p>
<%
    String sDbDrv = "org.postgresql.Driver";
    String loginUrl = "jdbc:mysql://127.0.0.1/cdcol";
    String loginUser = "root";
    String loginPasswd = "";
    String sTable = "cds";
    String sSql = "SELECT * FROM cds";
    Connection dbcon = null;
//SELECT
    try
    {
        Class.forName("com.mysql.jdbc.Driver").newInstance();
        dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
        Statement statement = dbcon.createStatement();
        ResultSet rs = statement.executeQuery(sSql);
        ResultSetMetaData rsmd = rs.getMetaData();
        int n = rsmd.getColumnCount();
        while( rs.next() )
    {

```

```

        out.println( "ID: " + rs.getString( 4 ) + "<br>Interpret: " + rs.get
String(
    2 ) + "<br>Album: " + rs.getString( 1 ) + "<br>Erscheinungsjahr: " +
rs.getString( 3 ) + "<br><br>");
    }
}
catch (ClassNotFoundException ex)
{
    System.err.println("ClassNotFoundException: " + ex.getMessage());
    throw new ServletException("Class not found Error");
}

//INSERT
try
{
    Class.forName("com.mysql.jdbc.Driver").newInstance();
    dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
    Statement insert = dbcon.createStatement();
    insert.executeUpdate("INSERT INTO cds (interpret, titel, jahr) VALUES
('Styx ', 'The Complete Wooden Nickel Rec Sampler', '2005')");
}
catch (Exception fehler)
{
    System.out.println("Fehler!!!\n");
    fehler.printStackTrace();
}

//UPDATE
try
{
    Class.forName("com.mysql.jdbc.Driver").newInstance();
    dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
    Statement update = dbcon.createStatement();
    update.executeUpdate("UPDATE cds SET titel='The Complete Wooden Nickel
Sampler 2005' WHERE interpret like '%Styx%' AND jahr = '2005'");
}
catch (Exception fehler)
{
    System.out.println("Fehler!!!\n");
    fehler.printStackTrace();
}

//DELETE
try
{
    Class.forName("com.mysql.jdbc.Driver").newInstance();
    dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
    Statement delete = dbcon.createStatement();
    delete.executeUpdate("DELETE FROM cds WHERE interpret like '%Styx%'");
}
catch (Exception fehler)

```



```

    {
        System.out.println("Fehler!!!\n");
        fehler.printStackTrace();
    }
%>
</body>
</html>

```

23.3.2 PostgreSQL

Zu Beginn etablieren Sie den JDBC-Treiber (Java Database Connectivity) für PostgreSQL als eine neue Java-Klasse. Den für Ihr Java JDK-Paket passenden Treiber besorgen Sie sich unter der URL <http://jdbc.postgresql.org>.

In diesem Beispiel setzen wir JDK 1.5 ein. Für diese Version ist die JDBC 3 gültig. Wir entschieden uns in diesem Fall für die letzte stabile Version 8.0 für Java 5. Nach dem Download benennen Sie den Treiber in `postgresql.jar` um und kopieren die Datei in den `common\lib`-Ordner im Tomcat.

Beispiel unter Windows:

```
<xamp>\tomcat\common\lib\postgresql.jar
```

Beispiel unter Linux:

```
$CATALINA_HOME/common/lib/postgresql.jar
```

Damit das JSP-Skript die Klasse auch laden kann, legen Sie es im gleichen Webapps-Verzeichnis, beispielsweise `jsp-examples`, ab. Ein erster Kontakt zu unserer Beispieldatenbank `cds` in der Datenbank `cdcol` sollte nun Erfolg zeigen. Lesen Sie bitte hierfür auch Kapitel 18, »PostgreSQL«.

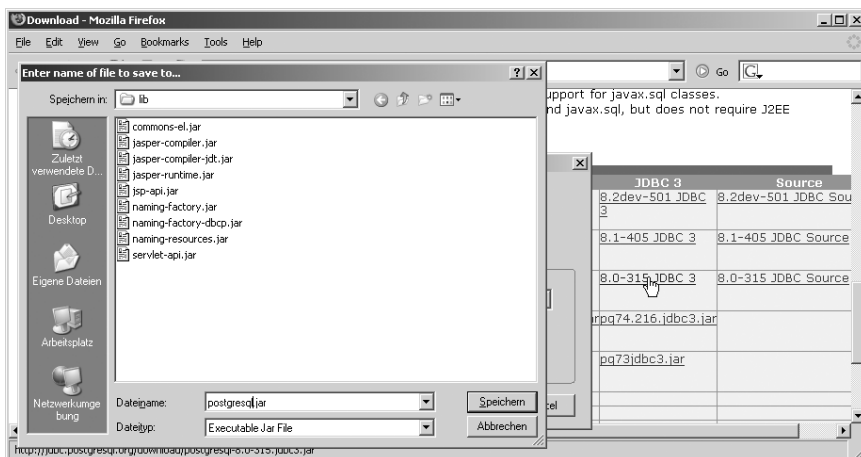


Abbildung 23.12: Den Treiber `postgresql.jar` speichern Sie gleich in das richtige Verzeichnis

Listing 23.3: Die postgresql.jsp mit dem Treiber und kurzem SELECT-Statement

```

<html>
<body>
<%@ page import="java.sql.*,java.util.*,javax.naming.*" %>
<h3>CD Musik Datenbank</h3> <p>
<%
    String loginUrl = "jdbc:postgresql://127.0.0.1/cdcol";
    String loginUser = "zaphod";
    String loginPasswd = "passwort";
    String sTable = "cds";
    String sSql = "SELECT * FROM cds";
    Connection dbcon = null;
    try
    {
        Class.forName("org.postgresql.Driver"); // Treiber laden
        dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
        Statement statement = dbcon.createStatement();
        ResultSet rs = statement.executeQuery(sSql);
        ResultSetMetaData rsmd = rs.getMetaData();
        int n = rsmd.getColumnCount();
        while( rs.next() )
        {
            out.println( "ID: " + rs.getString( 4 ) + "<br>Interpret: " + rs.getString( 2 ) + "<br>Album: " + rs.getString( 1 ) + "<br>Erscheinungsjahr: " + rs.getString( 3 ) + "<br><br>");
        }
    }
    catch (ClassNotFoundException ex)
    {
        System.err.println("ClassNotFoundException: " + ex.getMessage());
        throw new ServletException("Class not found Error");
    }
%>
</body>
</html>

```

Weitere SQL-Anweisungen können Sie nun analog zum MySQL-Beispiel (`mysql.jsp`) in das Skript einbinden.

23.3.3 Oracle

Kommen wir gleich auf die mögliche Misere zu sprechen. Wer von der Oracle-Homepage den für sein JDK und seinen Oracle-Server korrekten JDBC-Treiber lädt und im Verzeichnis `<tomcat>\common\lib` etabliert, könnte nach dem Neustart »Wunderbares« erleben.

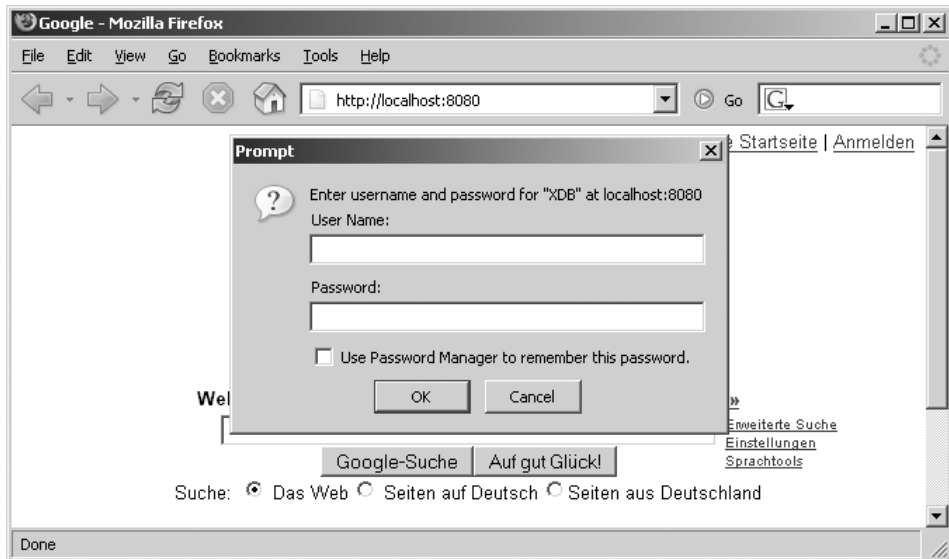


Abbildung 23.13: Der XDB-Server stahl tatsächlich dem Tomcat auf Port 8080 die Show!

Was ist also passiert? Sie haben mit dieser Installation einen schlummernden Dienst geweckt: den *Oracle-XDB-Server*. Dieser Server ist Bestandteil der Oracle-Versionen 9 und 10 und wird jetzt, d.h. mit den neuen Java-Klassen, automatisch über den *Oracle-Listener* aktiviert. Der Server erlaubt den Zugriff auf das *Oracle XML DB Repository* via Web. So weit wäre das in Ordnung, wenn der XDB-Server hierfür nicht unbedingt den Standardport 8080 des Tomcat besetzen würde. Damit fehlt Ihnen nun der Zugriff auf Ihren Tomcat.

- Lösung 1: Sie ändern den Port für die XML-Datenbank im *Oracle Enterprise Manager*, beispielsweise von Port 8080 auf Port 8082 (siehe Abbildung 23.14).
- Lösung 2: Alternativ wechseln Sie den Serverport des Tomcat in der *server.xml* von Port 8080 beispielsweise auf Port 8081.

Konfigurationsdatei: `<tomcat>\conf\server.xml`

Nach diesem kleinen Intermezzo in Sachen XBD-Server sollten Sie sich nun auf die *Oracle-JDBC-Installation* einlassen. Hierfür legen Sie den Treiber wie gewohnt unter `<tomcat>\common\lib` ab. Sie erhalten die benötigten Klassen über folgende URL:

http://www.oracle.com/technology/software/tech/java/sqlj_jdbc

Beispiel unter Windows:

`<xampp>\tomcat\common\lib\ojdbc14.jar`

Beispiel unter Linux:

`$CATALINA_HOME/common/lib/ojdbc14.jar`

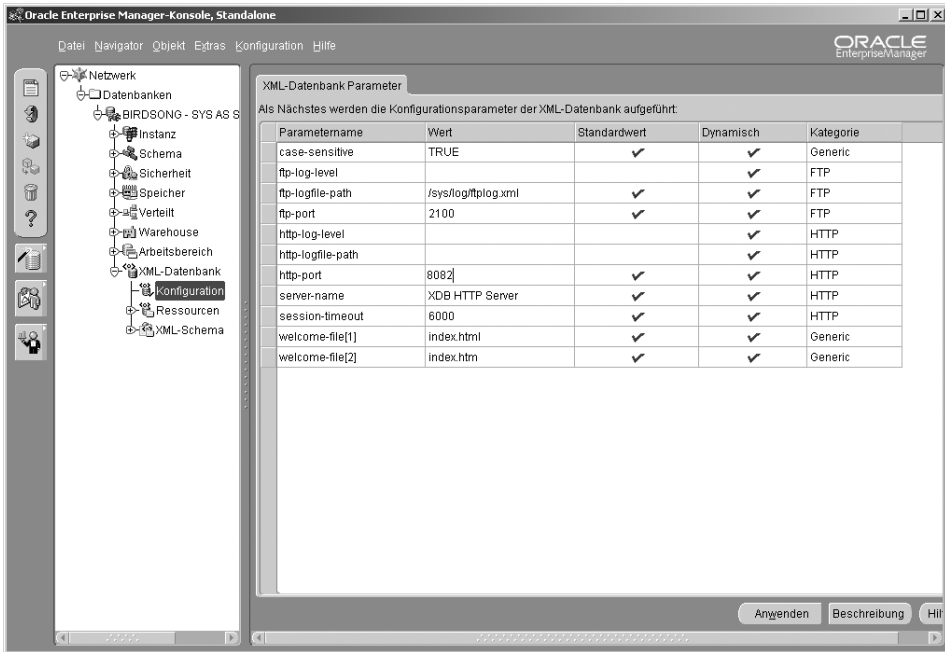


Abbildung 23.14: Den Port 8080 von XDB über den Oracle Enterprise Manager ändern

Nun ist es an der Zeit, ein neues JSP-Skript mit dem Namen `oracle.jsp` z.B. unter `webapps/jsp-examples` zu erstellen. Die grundsätzliche Systematik von JDBC in *Java Server Pages* ist Ihnen ja bereits durch die Zugriffe auf die bereits behandelten Datenbankserver wie MySQL bekannt. Schauen Sie sich bitte trotzdem diese Zeile etwas genauer an:

```
String loginUrl = "jdbc:oracle:thin:@localhost:1521:BIRDSONG";
```

JDBC-Treiber: `jdbc:oracle:thin`

Host: `localhost`

Port: `1521`

SID nur bei (m)einem Oracle-Server: `Birdsong`

Der Fallstrick liegt hier im *SID*. Die Information über die *korrekte SID* Ihrer konkreten Serverinstanz erhalten Sie übrigens über den *Oracle Enterprise Manager*.

Listing 23.4: Die `oracle.jsp` mit einem `SELECT` auf dem lokalen Oracle-Server

```
<html>
<body>
<%@ page import="java.sql.*,java.util.*,javax.naming.*" %>
<h3>CD Musik Datenbank</h3> <p>
```

```

<%
String loginUrl = "jdbc:oracle:thin:@localhost:1521:BIRDSONG";
String loginUser = "oswald";
String loginPasswd = "geheim";
String sTable = "cds";
String sSql = "SELECT * FROM cds";
Connection dbcon = null;
try
{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
    Statement statement = dbcon.createStatement();
    ResultSet rs = statement.executeQuery(sSql);
    ResultSetMetaData rsmd = rs.getMetaData();
    int n = rsmd.getColumnCount();
    while( rs.next() )
    {
        out.println( "ID: " + rs.getString( 4 ) + "<br>Interpret: " + rs.getString( 2 ) + "<br>Album: " + rs.getString( 1 ) + "<br>Erscheinungsjahr: " + rs.getString( 3 ) + "<br><br>" );
    }
}
catch (ClassNotFoundException ex)
{
    System.err.println("ClassNotFoundException: " + ex.getMessage());
    throw new ServletException("Class not found Error");
}
%>
</body>
</html>

```

23.3.4 Microsoft SQL Server

Hierbei können Sie zwischen zwei *JDBC-Treibern* wählen. Als Erstes haben Sie selbstverständlich die Möglichkeit, auf den *SQL Server JDBC-Treiber* von Microsoft zurückzugreifen. Den Treiber erhalten Sie über die Microsoft-Website:

<http://www.microsoft.com/sql/>

In diesem Beispiel verwenden wir jedoch die freie Alternative *jTDS*, eine FreeTDS-Variante für Java, die unter der Lizenz von *GNU LGPL* steht und unter folgender URL heruntergeladen werden kann:

<http://jtds.sourceforge.net>

jTDS

Nach dem Extrahieren des Pakets, in unserem Fall `jtds-1.2-dist.zip`, kopieren Sie die Jar-Bibliothek `jtds-1.2.jar` in das `common\lib`-Verzeichnis Ihres Tomcat-Servers. Dieser Connector in der Version 1.2 konnte übrigens bereits auf den MS Server 2005 zugreifen.

Windows (mit Tomcat Add-On):

```
<xampp>\tomcat\common\lib
```

Linux:

```
$CATALINA_HOME/tomcat/common/lib
```

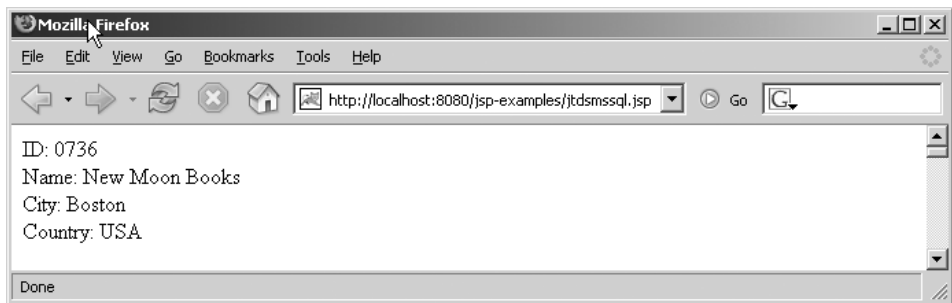


Abbildung 23.15: Die `jtdsmssql.jsp` im Einsatz mit einem MS SQL Server 2000

In dem Beispiellisting greifen wir auf die Datenbank `pubs` eines MS SQL Servers 2000 zu. Über eine kurze `SELECT`-Anweisung sollen alle Einträge der Tabelle `publishers` ausgegeben werden.

Listing 23.5: Die `jtdsmssql.jsp` für den Einsatz mit der Beispieltabelle `publishers`

```
<%@ page import="java.sql.*,java.util.*,javax.naming.*" %>
<%
String sDbDrv = "net.sourceforge.jtds.jdbc.Driver";
String loginUrl = "jdbc:jtds:sqlserver://127.0.0.1:1433/pubs";
String loginUser = "sa"; // Der SQLAdmin als Benutzer
String loginPasswd = "geheim"; // Das Passwort
String sSql = "SELECT * FROM publishers";
Connection dbcon = null;
//SELECT
try
{
    Class.forName("net.sourceforge.jtds.jdbc.Driver").newInstance();
    dbcon = DriverManager.getConnection(loginUrl, loginUser, loginPasswd);
    Statement statement = dbcon.createStatement();
    ResultSet rs = statement.executeQuery(sSql);
```

```
ResultSetMetaData rsmd = rs.getMetaData();
int n = rsmd.getColumnCount();
while( rs.next() )
{
    out.println( "ID: " + rs.getString( 1 ) + "<br>Name: " + rs.getString( 2 ) + "<br>City: " + rs.getString( 3 ) + "<br>Country: " + rs.getString( 5 ) + "<br><br>" );
}
}
catch (ClassNotFoundException ex)
{
    System.err.println("ClassNotFoundException: " + ex.getMessage());
    throw new ServletException("Class not found Error");
}

%>
```




24 Applikationsbeispiele

24.1 Joomla!

Joomla! gehört zu den populärsten Content-Management-Systemen, die es zurzeit gibt. Es ist im August 2005 aus Mambo entstanden und hat sich seitdem als eigenständiges CMS etabliert. Natürlich ist die enge Verwandtschaft zu Mambo immer noch stark zu merken.

24.1.1 Installation

Die Installation von Joomla! ist wie bei den meisten heutzutage erhältlichen PHP-Applikationen sehr einfach und innerhalb von wenigen Schritten bequem über eine Folge von Webformularen durchzuführen.

1. Laden Sie zunächst das Joomla-Paket von der Joomla-Homepage <http://www.joomla.org> herunter. In den folgenden Schritten verwenden wir die ZIP-Version des Pakets, da dieses Format unter Windows und auch unter Linux funktioniert. Wenn Sie wollen, können Sie unter Linux aber auch das tar.gz-Paket verwenden.
2. Entpacken Sie das heruntergeladene Paket in den `htdocs`-Ordner Ihrer XAMPP-Installation und nennen Sie den Ordner in `joomla` um.
3. Rufen Sie in Ihrem Browser <http://localhost/joomla/> auf. Sie sehen den PRE-INSTALLATION CHECK der Joomla-Installationsoberfläche. Hier wird angezeigt, ob Ihre Webserver-Umgebung mit Apache und PHP den Anforderungen von Joomla genügt. Da Sie XAMPP verwenden, wird es hier keine Probleme geben.

Unter Linux werden Sie hier allerdings feststellen, dass einige Verzeichnisse für Joomla schreibbar sein müssen. Am bequemsten erreichen Sie dies dadurch, dass Sie die Verzeichnisse dem Benutzer *nobody* zuordnen. Unter XAMPP für Linux läuft der Apache (und damit auch PHP) unter diesem Benutzer und wenn Sie die Verzeichnisse diesem Benutzer zuordnen, dann kann Joomla in sie schreiben.

Melden Sie sich als *root* auf Ihrem System an und rufen Sie die folgenden Befehle auf:

```
chown nobody joomla/administrator/backups/  
chown nobody joomla/administrator/components/  
chown nobody joomla/administrator/modules/
```

```

chown nobody joomla/administrator/templates/
chown nobody joomla/cache/
chown nobody joomla/components/
chown nobody joomla/images/
chown nobody joomla/images/banners/
chown nobody joomla/images/stories/
chown nobody joomla/language/
chown nobody joomla/mambots/
chown nobody joomla/mambots/content/
chown nobody joomla/mambots/editors/
chown nobody joomla/mambots/editors-xtl/
chown nobody joomla/mambots/search/
chown nobody joomla/media/
chown nobody joomla/modules/
chown nobody joomla/templates/

```

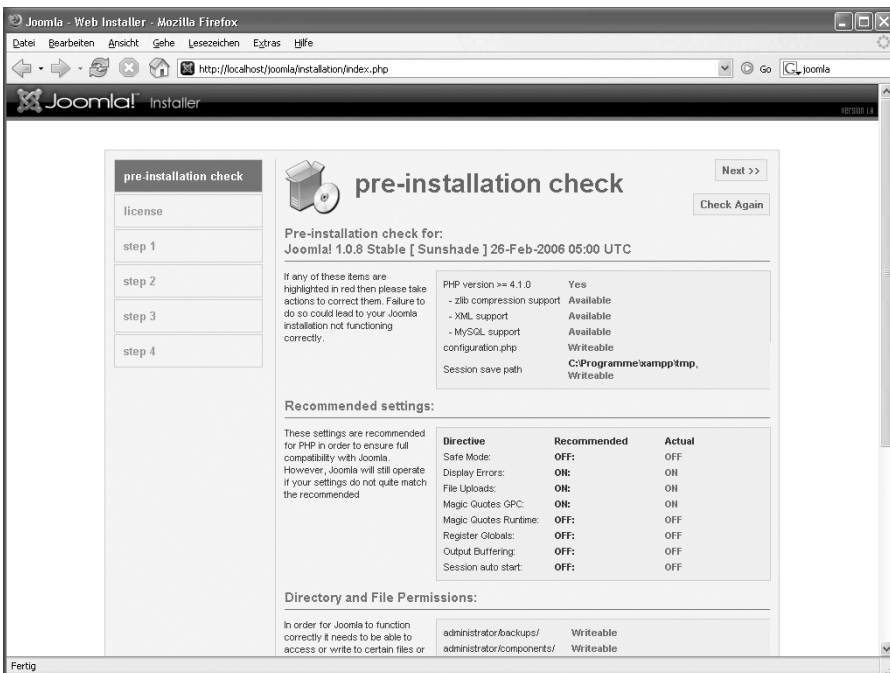


Abbildung 24.1: Joomla überprüft vor der Installation, ob alle Anforderungen erfüllt sind

Laden Sie danach die Joomla-Seite neu oder drücken Sie auf den CHECK AGAIN-Button. Die Warnungen, dass die Verzeichnisse nicht schreibbar sind, sollten nun verschwunden sein.

4. Klicken Sie auf NEXT >> und bestätigen Sie die Joomla-Lizenz.
5. Auf der nächsten Seite werden Sie nach der MySQL-Datenbankkonfiguration gefragt. Übernehmen Sie hierfür die folgenden Einstellungen:

HOST NAME	localhost
MYSQL USER NAME	root
MYSQL DATABASE NAME	joomla
MYSQL TABLE PREFIX	jos_

Bei MYSQL PASSWORD geben Sie Ihr MySQL-Root-Passwort an.

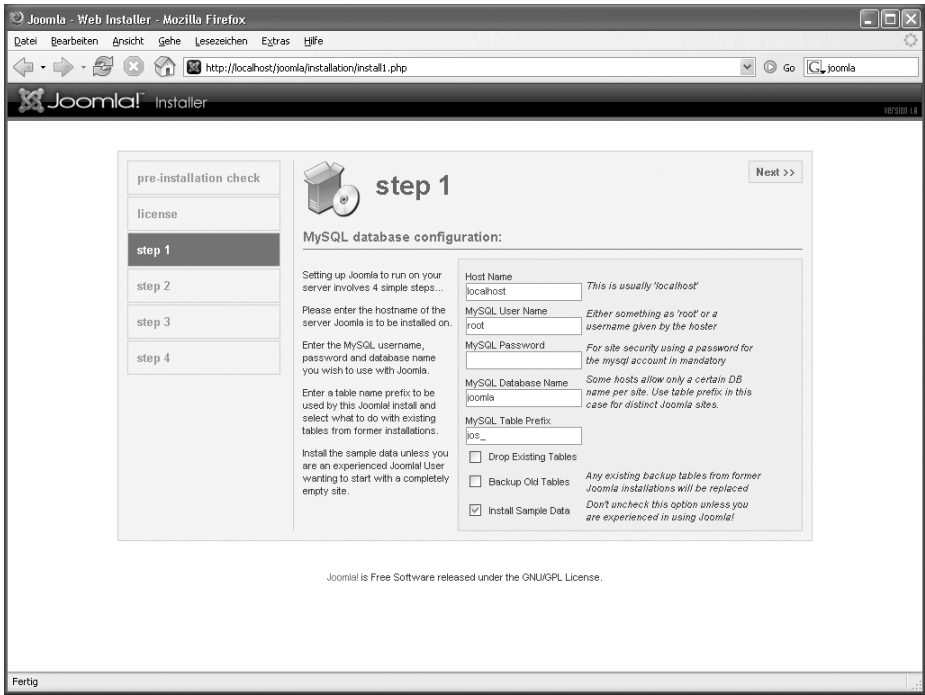


Abbildung 24.2: Dir Datenbankkonfiguration des Joomla-Installers

Die Verwendung des Benutzers *root* ist für eine richtige Joomla-Installation allerdings nicht zu empfehlen. Sie sollten lieber (zum Beispiel mit phpMyAdmin) einen eigenen MySQL-Benutzer und eine eigene MySQL-Datenbank für die Joomla-Installation einrichten und dessen Zugangsdaten hier angeben.

Klicken Sie auf NEXT >>.

6. Geben Sie nun Ihrer Joomla-Site unter ENTER THE NAME OF YOUR JOOMLA SITE einen Namen. Zum Beispiel: Meine Joomla Welt. Sie können diese Einstellung später jederzeit wieder ändern.

Klicken Sie danach auf NEXT >>.

7. In diesem Schritt können Sie noch ein paar Daten zu Ihrer Joomla-Installation vervollständigen:

- die URL Ihrer Joomla-Installation
- den Pfad auf der Festplatte zu Ihrer Joomla-Installation
- Ihre E-Mail-Adresse
- das Admin-Passwort

In der Regel werden Sie hier nur Ihre E-Mail-Adresse nachtragen müssen. Vielleicht wollen Sie das Admin-Passwort noch ändern, aber auch das vorgeschlagene Passwort sollte eine gute Zufallswahl sein.

Danach klicken Sie auf NEXT >>.

8. Das war's fast schon. Joomla ist nun vollständig installiert. Ihnen wird noch einmal Ihr Admin-Passwort angezeigt. Notieren Sie es sich an geeigneter Stelle bzw. merken Sie es sich gut.

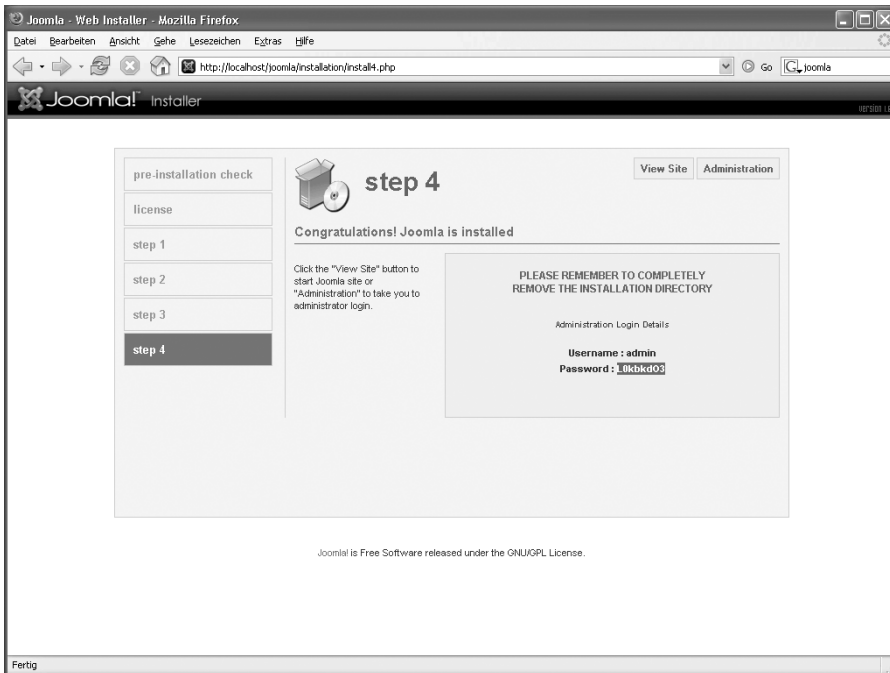


Abbildung 24.3: Die Installation von Joomla ist abgeschlossen. Sie sollten sich das Passwort an einer geeigneten Stelle aufbewahren.

Rechts oben werden nun zwei Buttons angezeigt:

- VIEW SITE führt Sie zu der Joomla-Website, so, wie sie von den Besuchern Ihrer Website gesehen wird. Zum Beispiel: <http://localhost/joomla/index.php>.

- **ADMINISTRATION** führt Sie zur Redaktions- bzw. Administrationsoberfläche von Joomla. Hier können Sie sämtliche Einstellungen an Joomla vornehmen, Inhalte ändern oder neue Inhalte eingeben. Wenn Sie Joomla so installiert haben, wie es auf den letzten Seiten beschrieben wurde, dann sollten Sie diese Oberfläche auch unter der folgenden URL erreichen: <http://localhost/joomla/administrator/index.php>.
9. Um die Installation von Joomla nun abzuschließen, wechseln Sie in das `joomla`-Verzeichnis in Ihrem `htdocs`-Ordner und löschen das `installation`-Verzeichnis. Es wird nun nicht mehr benötigt und würde nur noch ein Sicherheitsrisiko darstellen.

Das war's. Joomla ist nun auf Ihrem System installiert.



Abbildung 24.4: Die fertige Joomla-Installation von außen gesehen. Links unten in der Navigation finden Sie den Punkt Administration. Mit ihm gelangen Sie in die Administrationsoberfläche von Joomla.

24.1.2 Ein allererster Schritt

Nun ist dies ein XAMPP und kein Joomla-Buch, und so wollen wir jetzt auch nur noch einen ganz kleinen Schritt in die Benutzung von Joomla machen: Beispielhaft soll einer der Demoinhalte auf der Startseite von Joomla verändert werden.

1. Melden Sie sich an der Administrationsoberfläche von Joomla unter <http://localhost/joomla/administrator/index.php> an.

Nach der erfolgreichen Anmeldung sehen Sie das Hauptmenü der Joomla-Administrationsoberfläche.

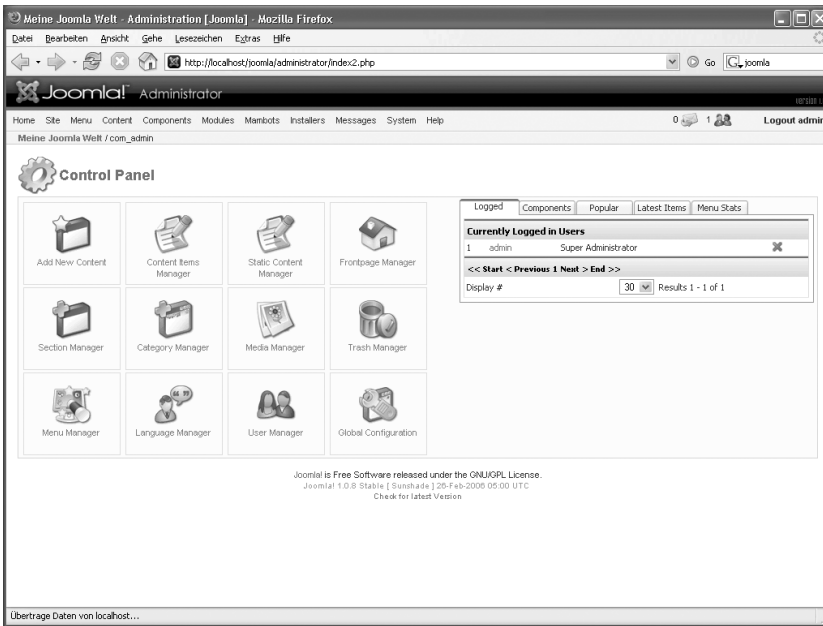


Abbildung 24.5: Das Hauptmenü der Joomla-Administrationsoberfläche

2. Wählen Sie hier den FRONTPAGE MANAGER aus, um Inhalte der Startseite zu verändern.
3. Wählen Sie nun den Inhaltspunkt WELCOME TO JOOMLA! aus und nehmen Sie eine Änderung an dem Text vor. Drücken Sie danach auf den SAVE-Button mit dem Diskettensymbol oben rechts.
4. Wenn Sie nun die Joomla-Außensicht unter <http://localhost/joomla/> aufrufen, finden Sie Ihren geänderten Text wieder.

Im Grunde funktioniert alles Weitere nach genau dem gleichen Prinzip. Ein guter Start für weiterführende Dokumentation dürfte <http://www.joomla.de/> sein, die Homepage der deutschen Joomla-Community.

Tipp



Wer das gedruckte Wort der Online-Dokumentation vorzieht, dem sei das ausgezeichnete Einsteiger-Buch von Hagen Graf, *Joomla!*, erschienen bei Addison-Wesley, empfohlen.

24.2 TYPO3

TYPO3 ist mit Sicherheit das bekannteste Open-Source-CMS und auch das mächtigste: Mit TYPO3 können Sie so ziemlich alles machen, was Sie wollen. Die Liste an Features und Erweiterungsmöglichkeiten ist wirklich sehr, sehr lang.

Aber TYPO3 ist ein ebenso populäres wie überaus kompliziertes und schlecht dokumentiertes System. Die Vielseitigkeit von TYPO3 hat ihren Preis. Und der wird in diesem Fall nicht in Euro gemessen.

Sie sollten es sich wirklich gut überlegen, ob Sie TYPO3 verwenden wollen. Wir können nichts Besseres empfehlen, müssen Sie aber im gleichen Atemzug vor der Komplexiertheit dieser Software warnen.

24.2.1 Installation ImageMagick

ImageMagick ist ein Paket zur automatisierten Umwandlung und Bearbeitung von verschiedensten Grafikformaten und Bilddaten. TYPO3 nutzt dieses Paket in hohem Maße und Sie sollten es auf alle Fälle installieren, um so auch den größten Nutzen aus TYPO3 ziehen zu können.

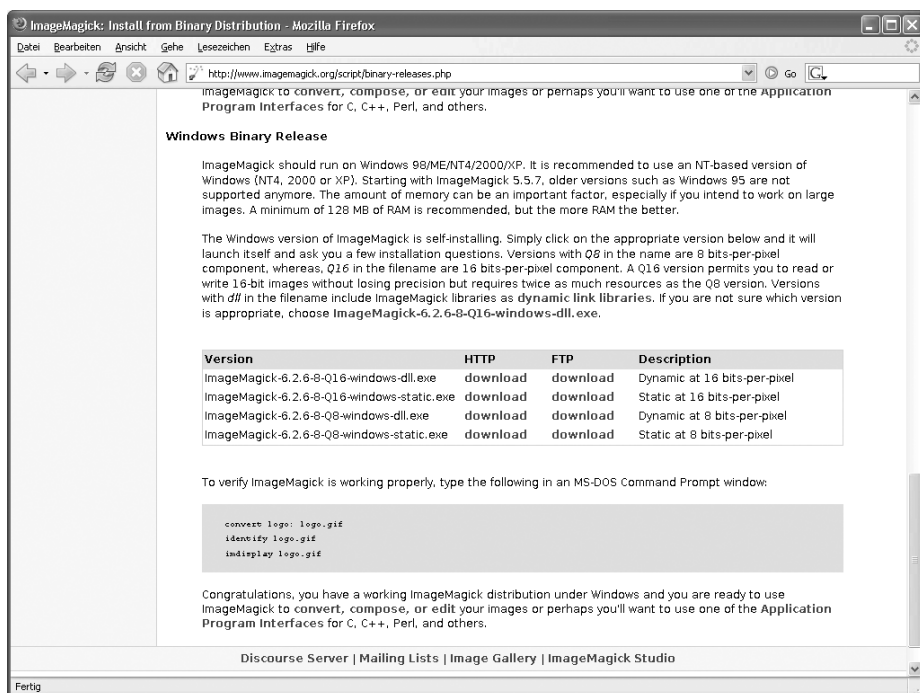


Abbildung 24.6: Die richtige ImageMagick-Datei für Windows zum Downloaden ist nicht leicht zu finden.

Unter Linux verwenden Sie zur Installation von ImageMagick das Paketmanagement Ihrer Distribution.

Unter Windows laden Sie das Paket von der ImageMagick-Website unter <http://www.imagemagick.org/script/binary-releases.php> herunter. Ganz weit unten unter *Windows Binary Release* finden Sie die ImageMagick-6.2.6-8-Q16-windows-dll.exe. Laden Sie dieses Paket herunter und installieren Sie es unter C:\Programme\ImageMagick. (Der Installer wird Ihnen C:\Programme\ImageMagick-6.2.6-Q16 vorschlagen. Kürzen Sie den Verzeichnisnamen um die Versionsnummer, so ersparen Sie sich spätere Tippfehler.)

24.2.2 Installation TYPO3

Nachdem Sie ImageMagick installiert haben, können Sie mit der eigentlichen TYPO3-Installation beginnen.

1. Laden Sie die beiden Dateien `typo3_src-4.0.zip` und `dummy-4.0.zip` von der Webseite <http://typo3.org/download/packages/> herunter.
2. Entpacken Sie das Paket `dummy-4.0.zip` in das `htdocs`-Verzeichnis Ihrer XAMPP-Installation und benennen Sie den Ordner von `dummy-4.0` nach `dummy` um.
3. Jetzt wird es ein bisschen kompliziert: In der `typo3_src-4.0.zip` finden Sie den Ordner `typo3_src-4.0`. Entpacken Sie den Inhalt dieses Ordners direkt in den `dummy`-Ordner. Unter `dummy` muss es nun einen Ordner `typo3` geben und das Ganze sollte jetzt so aussehen:

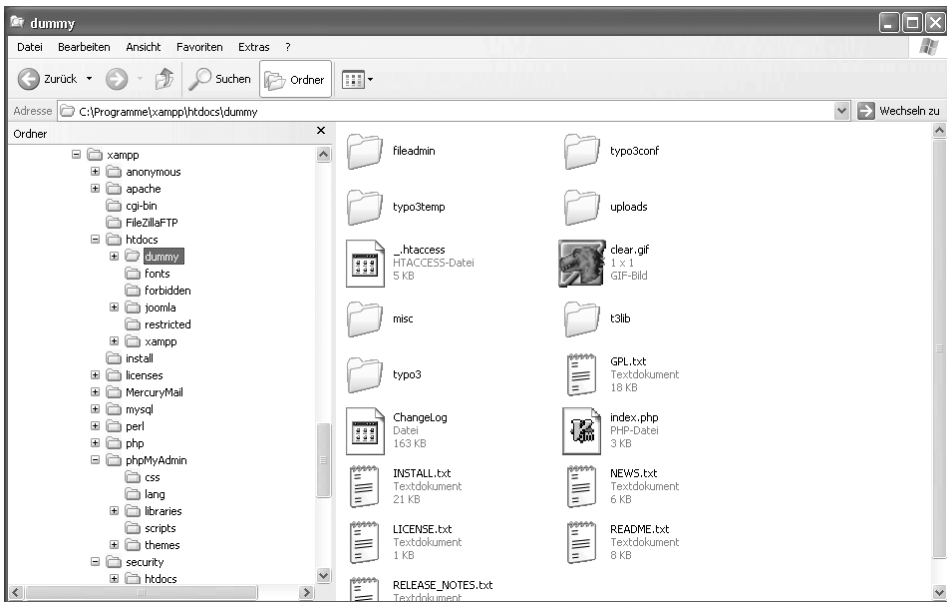


Abbildung 24.7: Nach dem Auspacken beider ZIP-Archive muss sich der `dummy`-Ordner in `htdocs` befinden und der `typo3`-Ordner im `dummy`-Ordner

4. Gehen Sie nun in die webbasierte Installationssoftware von TYPO3 und rufen Sie <http://localhost/dummy/typo3/install/> in Ihrem Browser auf.



Hinweis

Eventuell sehen Sie in diesem Schritt die folgende Fehlermeldung:

In the source distribution of TYPO3, the install script is disabled by a die() function call.

Fix: Open the file typo3/install/index.php and remove/out-comment the line that outputs this message!

Öffnen Sie in diesem Fall die Datei `dummy/typo3/install/index.php` bzw. `dummy\typo3\install\index.php` in einem Texteditor. Ziemlich am Anfang dieser Datei finden Sie die folgenden Zeilen:

```
if (1==0 || (substr($_SERVER['REMOTE_ADDR'],0,7)!='192.168' &&
$_SERVER['REMOTE_ADDR']!= '127.0.0.1')) {
    die("In the source distribution of TYPO3, the install script
is disabled by a die() function call.<br/><b>Fix:</b> Open the file
typo3/install/index.php and remove/out-comment the line that outputs
this message!");
}
```

Setzen Sie zwei Slashes – also `//` – vor den Aufruf von `die()` und deaktivieren Sie ihn so. Speichern Sie die Datei und rufen Sie die URL <http://localhost/dummy/typo3/install/> noch einmal auf.

5. Sie werden nach einem Installationspasswort gefragt. Melden Sie sich hier mit dem Passwort `joh316` an.
6. Nun werden Sie aufgefordert, ein neues Passwort für die Installation zu setzen. Geben Sie das neue Passwort zweimal ein und drücken Sie auf SET NEW PASSWORD.

Unter Linux erhalten Sie daraufhin die folgende Fehlermeldung:

/opt/lampp/htdocs/dummy/typo3conf/localconf.php is not writable!

Der Installer von TYPO3 hat nicht die Rechte, das neue Passwort zu speichern. Um ihm die nötigen Rechte zu geben, rufen Sie eine *root*-Shell auf und geben die folgenden beiden Befehle ein:

```
chown nobody /opt/lampp/htdocs/dummy/typo3conf/localconf.php
chown nobody /opt/lampp/htdocs/dummy/typo3conf
```

Drücken Sie im Browser auf AKTUALISIEREN bzw. NEU LADEN.

7. Um die Installation fortzusetzen, werden Sie nach dem neuen Passwort gefragt. Geben Sie dieses Passwort ein.
8. Sie sehen nun das TYPO3-Installationsmenü. Wenn noch nicht automatisch ausgewählt, dann wählen Sie hier den ersten Punkt: 1: BASIC CONFIGURATION.

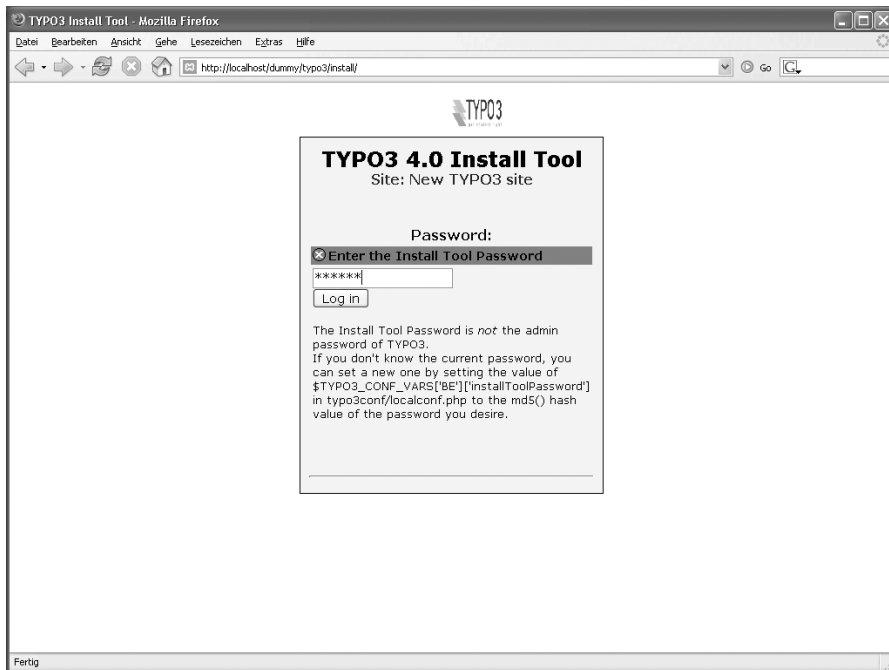


Abbildung 24.8: Zu Beginn der Installation werden Sie nach dem Install Tool Password gefragt. Solange Sie es noch nicht selbst gesetzt haben, ist es joh316.

TYPO3 überprüft, ob die TYPO3-Installation so weit in Ordnung ist. Es wird Ihnen dann anzeigen, dass noch ein paar Verzeichnisse fehlen:

```
typo3temp/pics/ directory does not exist
typo3temp/temp/ directory does not exist
typo3temp/l1xml/ directory does not exist
typo3temp/cs/ directory does not exist
typo3temp/GB/ directory does not exist
```

Erstellen Sie diese fünf Verzeichnisse im Ordner `dummy/typo3temp` bzw. `dummy\typo3temp`.

9. Laden Sie die Seite 1: BASIC CONFIGURATION erneut in Ihrem Browser. Die Warnungen sollten unter Windows nun komplett verschwunden sein.

Unter Linux werden Sie allerdings weiterhin Warnungen erhalten: Einige Verzeichnisse müssen für TYPO3 schreibbar sein. Rufen Sie eine *root*-Shell auf und geben Sie dort diese Verzeichnisse dem Benutzer *nobody*. Unter XAMPP für Linux läuft der Apache (und damit auch PHP) unter diesem Benutzer und wenn Sie die Verzeichnisse diesem Benutzer zuordnen, kann TYPO3 automatisch auch in diese schreiben.

```

chown nobody /opt/lampp/htdocs/dummy/typo3temp/pics
chown nobody /opt/lampp/htdocs/dummy/typo3temp/temp
chown nobody /opt/lampp/htdocs/dummy/typo3temp/llxml
chown nobody /opt/lampp/htdocs/dummy/typo3temp/cs
chown nobody /opt/lampp/htdocs/dummy/typo3temp/GB
chown nobody /opt/lampp/htdocs/dummy/typo3temp
chown nobody /opt/lampp/htdocs/dummy/typo3conf/ext
chown nobody /opt/lampp/htdocs/dummy/typo3conf/llon
chown nobody /opt/lampp/htdocs/dummy/uploads
chown nobody /opt/lampp/htdocs/dummy/uploads/pics
chown nobody /opt/lampp/htdocs/dummy/uploads/media
chown nobody /opt/lampp/htdocs/dummy/uploads/
chown nobody /opt/lampp/htdocs/dummy/uploads/tf
chown nobody /opt/lampp/htdocs/dummy/fileadmin/_temp_
chown nobody /opt/lampp/htdocs/dummy/fileadmin
chown nobody /opt/lampp/htdocs/dummy/typo3/ext

```

Alternativ – zwar ein wenig unsicher, aber vielleicht einfacher zu tippen – können Sie auch alle obigen Befehle durch diesen einen ersetzen:

```
chown -R nobody /opt/lampp/htdocs/dummy
```

Damit übergeben Sie die gesamte TYPO3-Installation dem Benutzer *nobody*. Sicherlich ein wenig zu viel des Guten, aber eigentlich spricht auch kaum etwas dagegen.

10. Bleiben Sie noch auf der Seite 1: BASIC CONFIGURATION und scrollen Sie die Seite weiter herunter.

Unter UPDATE LOCALCONF.PHP können Sie nun die Zugangsdaten zu Ihrer MySQL-Datenbank angeben.

Geben Sie hier folgende Daten ein:

USERNAME	root
HOST	localhost

Bei PASSWORD geben Sie Ihr MySQL-Root-Passwort an.

Die Verwendung des Benutzers *root* ist für eine richtige TYPO3-Installation allerdings nicht zu empfehlen. Sie sollten lieber (zum Beispiel mit phpMyAdmin) einen eigenen MySQL-Benutzer und eine eigene MySQL-Datenbank für die TYPO3-Installation einrichten und die Zugangsdaten des neuen MySQL-Benutzers hier angeben.

Scrollen Sie nun ganz nach unten und klicken Sie auf den Button UPDATE LOCALCONF.PHP.

11. Gehen Sie erneut auf die Seite 1: BASIC CONFIGURATION und scrollen Sie nach unten.

Etwa auf der Hälfte der Seite – unter CHECK IMAGE MAGICK: – finden Sie die ImageMagick-Konfiguration. Geben Sie unter CHECK THIS PATH FOR IMAGEMAGICK INSTALLATION: folgenden Wert ein:

Unter Linux: `/usr/bin`

Unter Windows: `C:/Programme/ImageMagick`

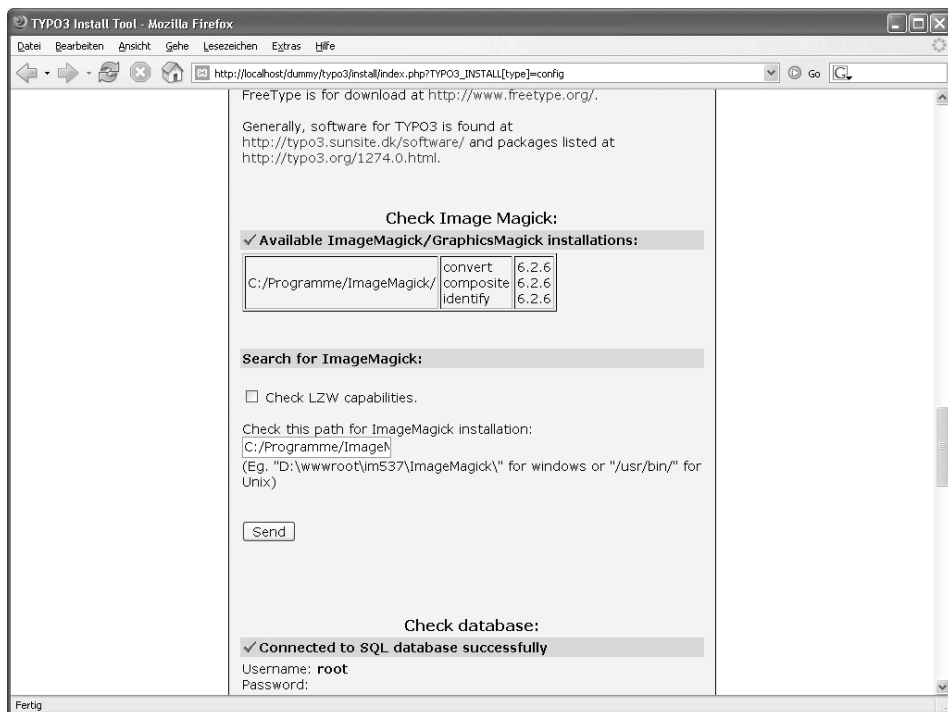


Abbildung 24.9: Die Konfigurationseinstellungen von TYPO3 sind oft versteckt in sehr langen und unübersichtlichen HTML-Seiten. Hier sehen Sie die Pfadeinstellung zu ImageMagick.

12. Bleiben Sie auf 1: BASIC CONFIGURATION und scrollen Sie noch weiter nach unten.

Unter CHECK DATABASE: können Sie nun die MySQL-Datenbank auswählen, die TYPO3 verwenden soll.

Sie können hier eine bestehende Datenbank auswählen oder automatisch eine neue Datenbank anlegen lassen. Wenn Sie im vorletzten Schritt extra für TYPO3 eine Datenbank angelegt haben, dann wählen Sie hier den Namen dieser Datenbank. Ansonsten geben Sie einfach `typo3` als Datenbanknamen ein. Der Installer wird dann diese Datenbank automatisch einrichten.

13. Etwas weiter unten finden Sie die Konfiguration für den ENCRYPTION KEY. Klicken Sie auf den Button GENERATE RANDOM KEY darunter, um einen Zufallsschlüssel zu generieren.
14. Bleiben Sie immer noch auf dieser Seite und scrollen Sie ganz zum Ende.
Direkt über den UPDATE LOCALCONF.PHP-Button finden Sie ein Eingabefeld für den Wert [GFX][TTFDPI]. In dem Feld wird 72 stehen, ändern Sie den Wert in 96.
15. Klicken Sie nun auf UPDATE LOCALCONF.PHP, um Ihre vier letzten Einstellungen endlich zu speichern.
Damit ist der Menüpunkt 1: BASIC CONFIGURATION abgeschlossen.
16. Wählen Sie nun den zweiten Menüpunkt 2: DATABASE ANALYSER aus. Scrollen Sie nach unten, bis Sie folgende Zeile sehen:

Update required tables	COMPARE
Dump static data	IMPORT
Compare with \$TCA Create "admin" user Reset user preferences Clear tables	

Klicken Sie auf COMPARE. Eine lange Liste mit SQL-Anweisungen erscheint. Scrollen Sie bis ans Ende der Seite und klicken Sie auf WRITE TO DATABASE.

Mit diesem Schritt haben Sie nun die Datenbank für TYPO3 eingerichtet.

Nun müssen Sie noch einen Administratorbenutzer für TYPO3 einrichten.

17. Wählen Sie dazu erneut den Menüpunkt 2: DATABASE ANALYSER aus. Scrollen Sie an die gleiche Stelle wie bei Schritt 16. Sie finden dort den Link CREATE »ADMIN« USER. Klicken Sie auf diesen Link und ein Formular mit der Überschrift CREATE ADMIN USER erscheint etwas weiter unten auf der Seite.

Geben Sie hier einen Benutzernamen und ein Passwort für den TYPO3-Admin an und klicken Sie auf WRITE TO DATABASE. Etwas oberhalb des Formulars erscheint die Meldung USER CREATED.

Mit diesem Schritt haben Sie den Administratorzugang eingerichtet.

18. Wählen Sie nun den Menüpunkt 3: UPDATE WIZARD aus. Ein Text weist Sie darauf hin, dass Ihre TYPO3-Installation sich im TYPO3 3.8-Kompatibilitätsmodus befindet. Klicken Sie auf den CHANGECOMPATIBILITYVERSION-Button.

Scrollen Sie ein wenig nach unten und aktivieren Sie die beiden Checkboxes vor

Yes, I know that I will need to update my stylesheets to comply with this change.

und

Ignore selection above – WARNING: this might break the output of your website.

Klicken Sie danach auf den Button 3 -PERFORM UPDATES!

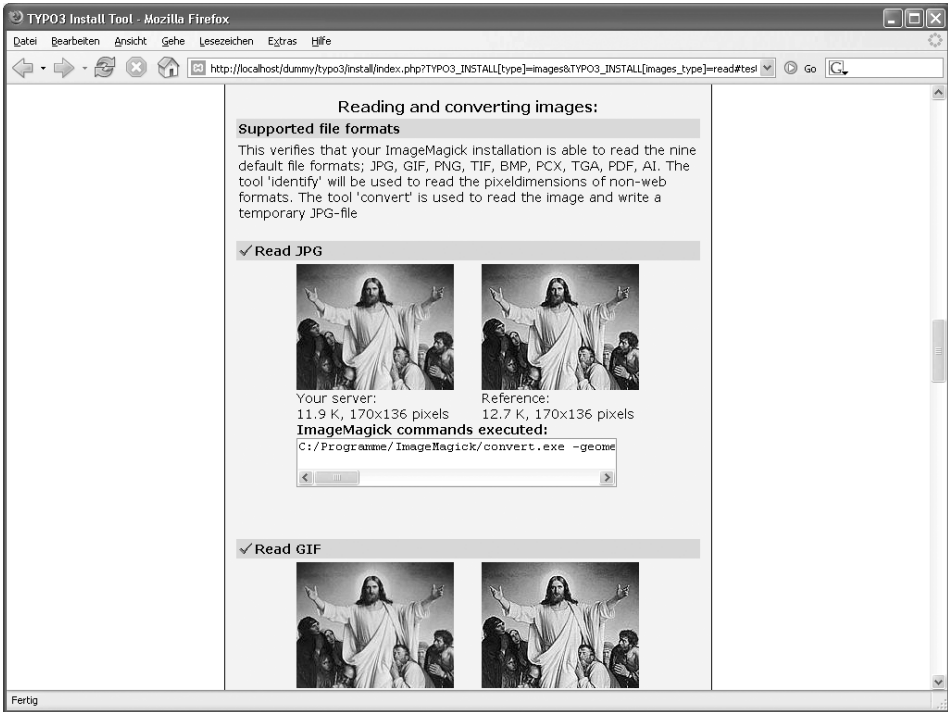


Abbildung 24.10: Zum Ende der Installation sollten Sie noch überprüfen, ob ImageMagick funktioniert

19. Jetzt ist eigentlich alles erledigt. Zur Sicherheit und wenn Sie noch Lust haben, gehen Sie jetzt noch zu 4: IMAGE PROCESSING und scrollen diese Seite bis ans Ende herunter. Hier können Sie das Funktionieren von ImageMagick und GD überprüfen.

Sie finden ein kleines Menü mit den Punkten:

- 1: READING IMAGE FORMATS
- 2: WRITING GIF AND PNG
- 3: SCALING IMAGES
- 4: COMBINING IMAGES
- 5: GD LIBRARY FUNCTIONS

Rufen Sie zumindest den ersten Test auf, um zu überprüfen, ob Ihr ImageMagick auch funktioniert. Wenn der erste Test bestanden wird, dann wird in der Regel auch der Rest funktionieren.

Das war's. TYPO3 ist nun auf Ihrem System installiert.



Abbildung 24.11: Nach der Installation können Sie sich im TYPO3-Backend anmelden

Scrollen Sie nun noch einmal auf einer der Installer-Seiten ganz nach unten. Dort finden Sie die folgenden zwei unscheinbaren Links:

- **BACKEND ADMIN IN NEW WINDOW** führt Sie in die Redaktions- bzw. Administrationssicht von TYPO3, in TYPO3 gerne das *Backend* genannt. Wenn Sie sich an die Installationsanweisungen auf den letzten Seiten gehalten haben, wird diese Sicht unter der Adresse <http://localhost/dummy/typo3/> erreichbar sein.
- **Der Link auf FRONTEND WEBSITE IN NEW WINDOW** führt Sie auf die Außensicht von TYPO3, also das, was der Besucher Ihrer Website sieht. Im TYPO3-Jargon wird diese Sicht auch gerne *Frontend* genannt. Nach der Installation, wie sie auf den letzten Seiten beschrieben wurde, führt dieser Link zu <http://localhost/dummy/>.

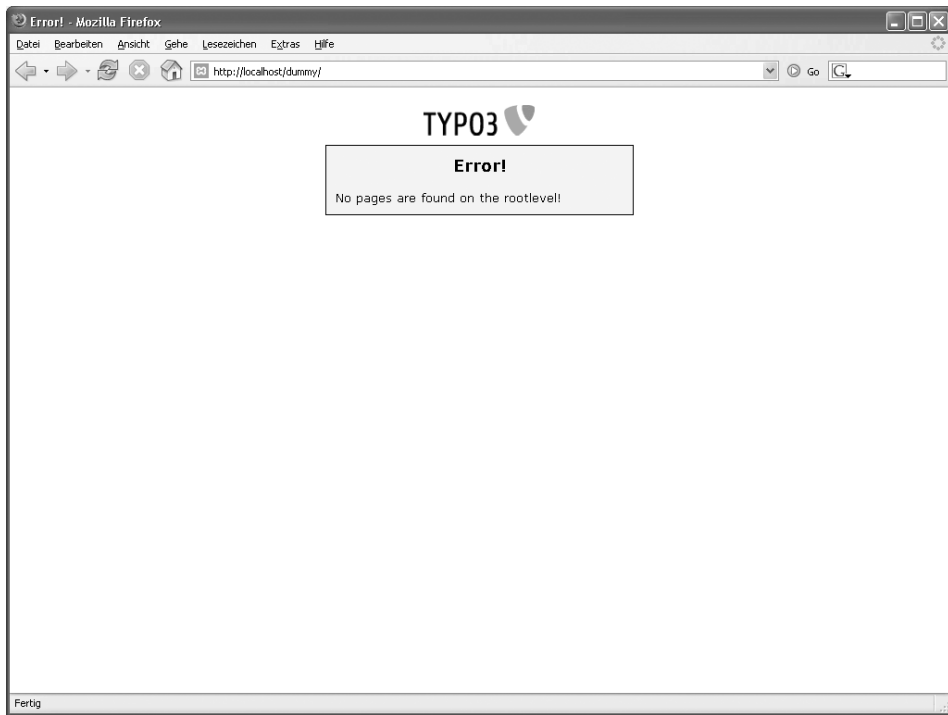


Abbildung 24.12: Wenn Sie sich das TYPO3-Frontend anschauen, werden Sie außer einer Fehlermeldung nichts finden.

24.2.3 Ein allererster Schritt

Wie schon am Anfang dieses Abschnitts erwähnt, ist TYPO3 kein einfaches CMS. Daher sollten Sie auch von dem allerersten Schritt nicht zu viel erwarten.

Wenn Sie sich gerade das Frontend von TYPO3 ansehen, sehen Sie dort lediglich die Fehlermeldung *No pages are found on the rootlevel!*. Diese Fehlermeldung werden Sie mit den nächsten Schritten verschwinden lassen. Dazu werden Sie

- eine neue Seite im TYPO3-Backend anlegen und
 - für diese Seite ein neues Template anlegen.
1. Melden Sie sich im TYPO3-Backend als Administrator an.
 2. Links sehen Sie die Hauptnavigation von TYPO3. Wählen Sie hier den ersten Punkt PAGE aus. In der Mitte erscheint eine Übersicht über die bestehenden Seiten. Im Moment dürfte es noch keine Seiten geben und dort dürfte nur der Name Ihrer TYPO3-Website stehen: NEW TYPO3 SITE.
 3. Klicken Sie auf die Weltkugel links von NEW TYPO3 SITE. Ein kleines Popup-Menü erscheint:

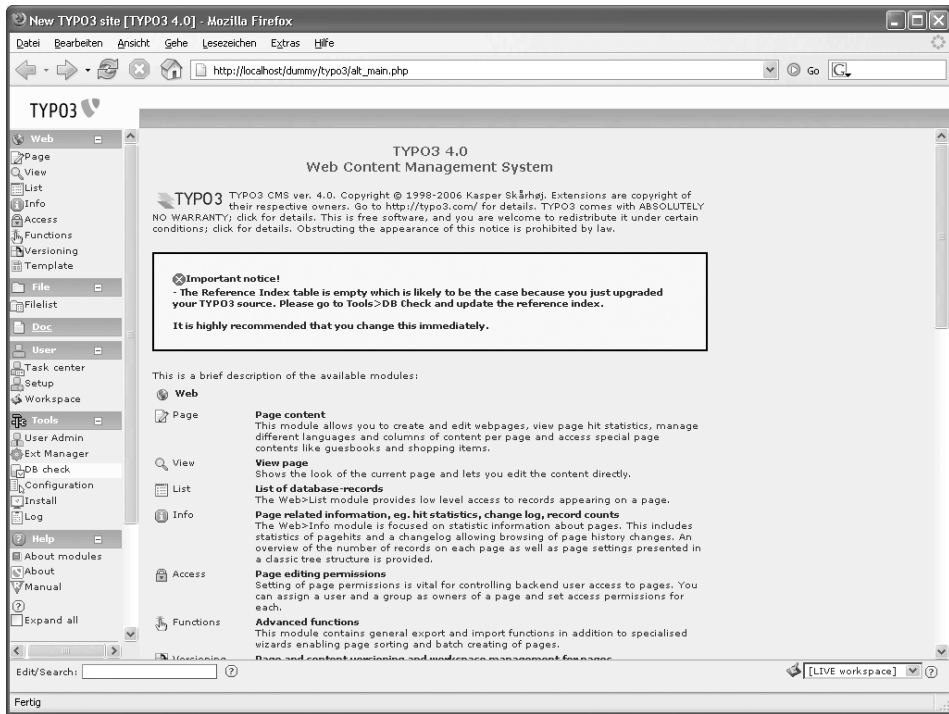
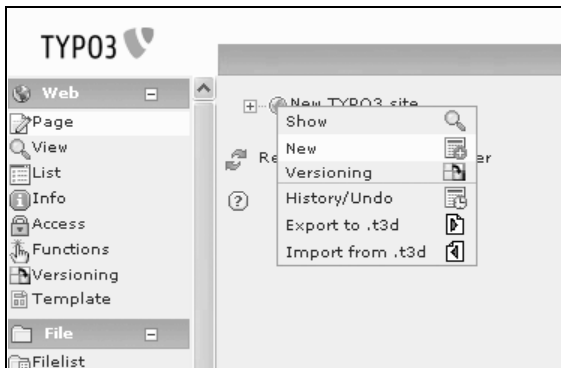


Abbildung 24.13: Evtl. bekommen Sie im TYPO3-Backend diese Warnung angezeigt. Gehen Sie in diesem Fall links in der Navigation auf DB Check, wählen Sie oben in der Mitte Manage Reference Index aus und klicken Sie schließlich auf Update now!.



Wählen Sie hier NEW aus.

4. Auf der rechten Seite erscheint eine Auswahl von Seitenelementen, die Sie nun neu erstellen können. Wählen Sie hier PAGE (INSIDE) aus. Es erscheint die Eingabemaske für eine neue Seite:

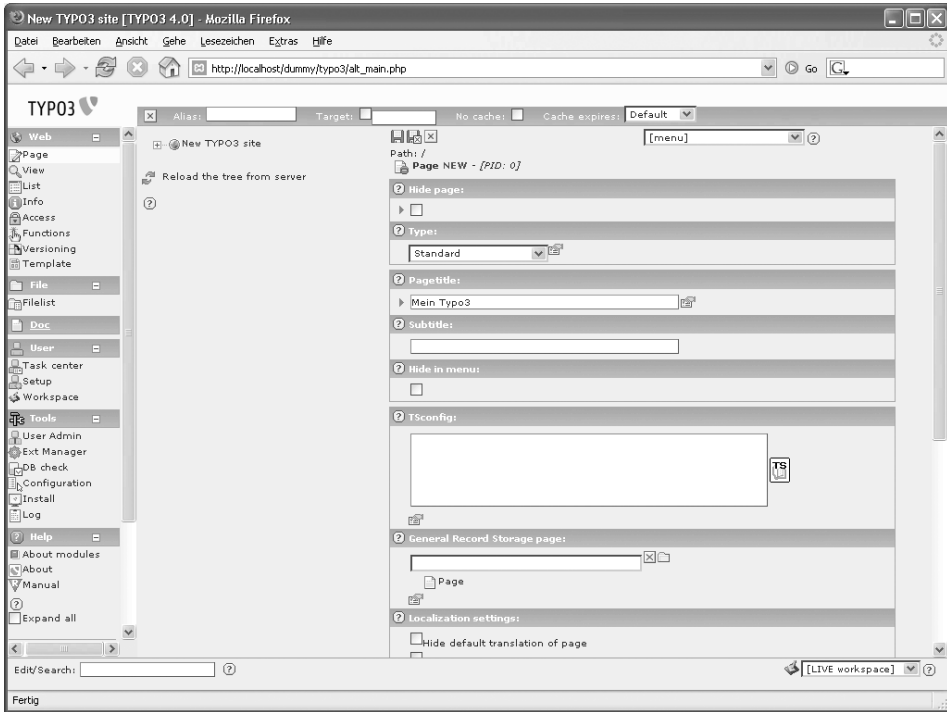


Abbildung 24.14: Das Hinzufügen einer neuen Seite in TYPO3

- Entfernen Sie das Häkchen bei HIDE PAGE ganz am Anfang dieses Formulars und geben Sie dann unter PAGETITLE einen beliebigen Namen für Ihre neue Seite ein. Zum Beispiel: Mein TYPO3.

Drücken Sie auf das Diskettensymbol ganz oben, um Ihre Einstellungen zu speichern.

Wenn Sie nun im Frontend unter <http://localhost/dummy/> nachsehen, werden Sie feststellen, dass es zwar eine andere, aber immer noch eine Fehlermeldung gibt:

No template found!

Zwar gibt es nun eine Seite, aber diese Seite hat noch kein Template. Mit Templates werden der Aufbau und die Struktur einer Seite in TYPO3 beschrieben. TYPO3 weiß also noch nicht, wie diese Seite überhaupt dargestellt werden soll.

- Klicken Sie auf das kleine Quadrat vor der Weltkugel, um den Ordner NEW TYPO3 SITE zu öffnen. Sie sehen unterhalb von NEW TYPO3 SITE den Titel Ihrer neuen Seite angezeigt.
- Klicken Sie einmal auf den Titel Ihrer neuen Seite und danach links in der Navigation auf TEMPLATE.

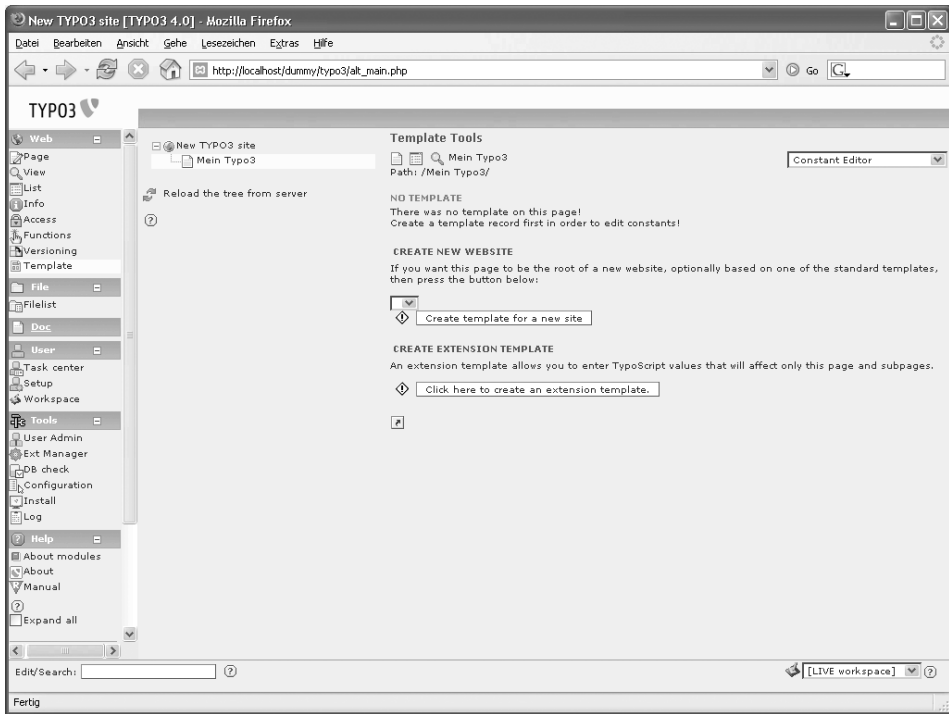


Abbildung 24.15: Jede Seite in TYPO3 braucht auch ein Template

8. Klicken Sie nun im rechten Bereich auf den Button CREATE TEMPLATE FOR A NEW SITE.

Damit haben Sie nun ein Standardtemplate eingerichtet. Wenn Sie erneut einen Blick auf das Frontend unter *http://localhost/dummy/* werfen, werden Sie keine Fehlermeldung, sondern folgenden Text vorfinden:

HELLO WORLD!

Damit sehen Sie die Ausgabe des Standardtemplates von TYPO3. Um nun noch wenigstens einen eigenen Text von TYPO3 angezeigt zu bekommen, sollten Sie auch noch die nächsten Schritte durchspielen und das neu eingerichtete Template ein wenig anpassen.

9. Nachdem Sie auf den Button CREATE TEMPLATE FOR A NEW SITE geklickt haben, erscheint oben rechts eine Auswahlliste, in der CONSTANT EDITOR angezeigt wird. Wählen Sie hier INFO/MODIFY aus.
10. Es erscheint eine kleine Tabelle, in deren letzten Zeile SETUP steht. Vor dem Setup finden Sie ein kleines Stiftsymbol. Klicken Sie darauf und es erscheint eine Maske, mit der Sie den Setup-Bereich eines TYPO3-Templates bearbeiten können.

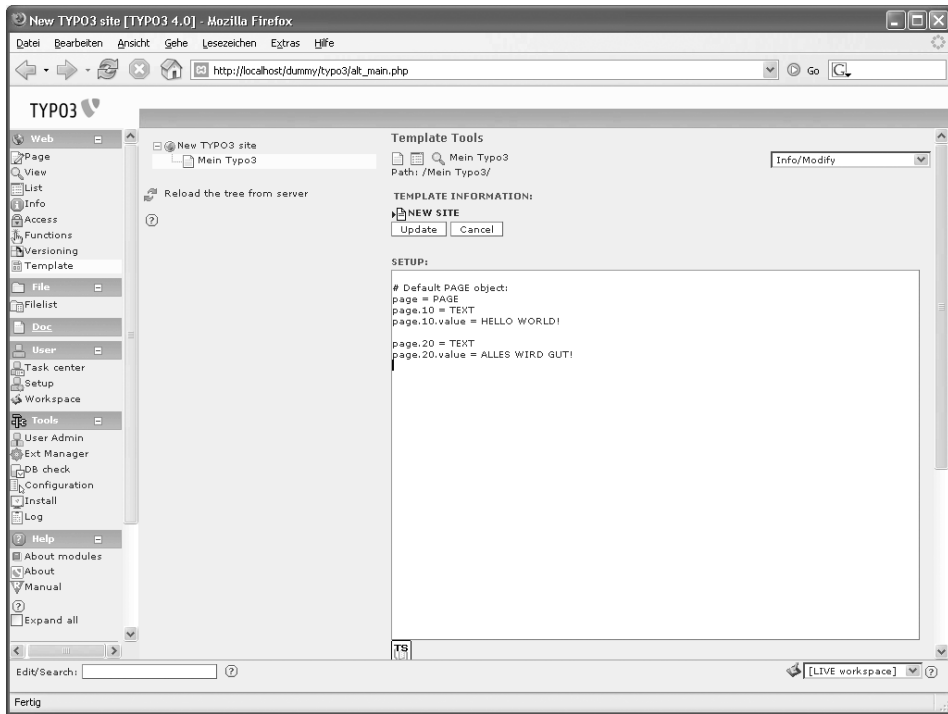


Abbildung 24.16: Hier können Sie Einstellungen an einem TYPO3-Template vornehmen

11. In der erscheinenden Eingabemaske sollten Sie folgendes Listing finden:

```
# Default PAGE object:
page = PAGE
page.10 = TEXT
page.10.value = HELLO WORLD!
```

Dieses Listing ist in Typo-Script verfasst und beschreibt das aktuelle Template. Sie können dieses Listing etwa so lesen:

Dieses Template besteht aus einer Seite (*page*). An Position 10 dieser Seite steht ein Text (*TEXT*). Dieser Text hat einen statischen Wert (*value*) und dieser Wert ist HALLO WELT!.

Fügen Sie unterhalb dieses Textes folgende zwei Zeilen hinzu:

```
page.20 = TEXT
page.20.value = ALLES WIRD GUT!
```

Damit erweitern Sie das Template um einen zweiten Text an Position 20 der Seite. Dieser Text hat den Wert: ALLES WIRD GUT!.

Wenn Sie nun im Browser auf Ihr TYPO3-Frontend schauen, werden Sie sehen, dass dort die Seite um diesen zweiten Text erweitert wurde.

Normalerweise wird man solche Texte nicht über ein Template eingeben, sondern das Template nur dazu verwenden, um das HTML und die Struktur einer Seite zu beschreiben. Aber ein Template zu erstellen, das mit dynamischen Inhalten gefüllt werden kann, würde den Rahmen dieses Buches sprengen. Allein um diese Änderung durchzuführen, bedarf es elf Schritte.

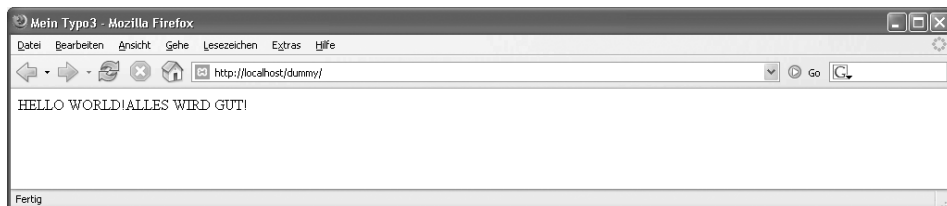


Abbildung 24.17: Eine erste, fast eigene Seite mit TYPO3



Tipp

Für alle weiteren Schritte sollten Sie sich ein TYPO3-Buch gönnen. Zum Beispiel *TYPO3 für Anwender* von Joscha Feth, erschienen bei Addison-Wesley. Die Online-Dokumentation zu TYPO3 ist nicht wirklich zu empfehlen.

24.3 Gallery

Gallery ist ein in PHP programmiertes Fotoalbum für das Web. Hierüber können Sie nicht nur Ihre Alben managen, sondern auch über eine Benutzerverwaltung die Organisation einzelner Alben delegieren.

Für dieses Buch wurde die zum Zeitpunkt der Drucklegung aktuelle Version 2.0.2 von *Gallery* verwendet. Die Homepage heißt <http://gallery.menalto.com>.

24.3.1 Voraussetzungen

Als Grundlage für eine gelungene Installation von *Gallery* brauchen Sie:

- Apache HTTPD
- MySQL
- PHP mit GD-Unterstützung

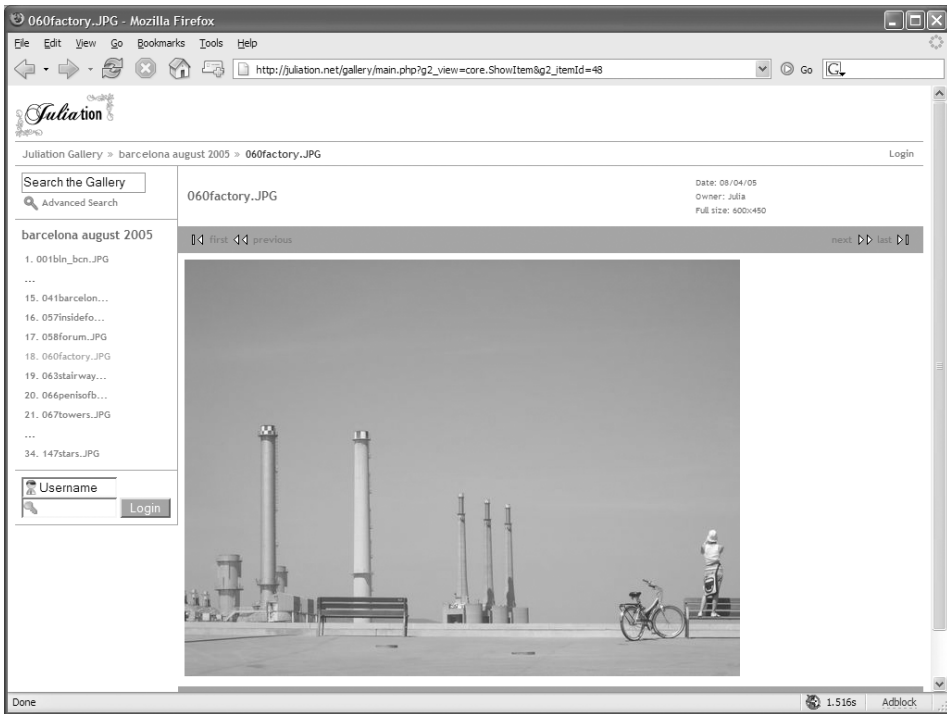


Abbildung 24.18: Gallery-Beispiel unter <http://julation.net/gallery/>

Ein FTP-Server ist darüber hinaus praktisch, aber nicht unbedingt notwendig. Sollten Sie auf einem Remote-Server unter Linux arbeiten, sollten Sie für Ihren Bereich die Read-, Write- und Execute-Rechte über die `chmod`-Anweisung für Ihre Verzeichnisse verändern können. Ein SSH-Zugang hierfür ist dabei sehr von Vorteil.

Alles, was Sie brauchen, ist also XAMPP. Möchten Sie Ihre Bilder auch im Internet veröffentlichen? Dann installieren Sie XAMPP auf einen Root-Server. Aber denken Sie hierbei bitte auch an eine mögliche Traffic-Begrenzung Ihres Anbieters. Je nach Attraktivität Ihrer Bilder kann diese sehr schnell überschritten sein.

24.3.2 Vorbereitungen

Im Folgenden wird die Installation unter Linux sowie unter Windows vorgestellt.

1. Besorgen Sie sich das aktuelle Gallery2-Paket von <http://gallery.menalto.com/downloads>.
2. Entpacken Sie das Paket (*tar.gz* für Linux, *zip* für Windows) direkt in das `htdocs`-Verzeichnis im XAMPP.

Linux:

```
tar xvfz gallery-2.0.2-full.tar.gz -C /opt/lampp/htdocs
```

Windows:

Extrahieren Sie die `gallery-2.0.2-full.zip` nach `\xampp\htdocs`.

3. Richten Sie auf dem Zielsystem eine neue Datenbank mit dem Namen *gallery2* für MySQL ein. phpMyAdmin eignet sich dazu sehr gut.

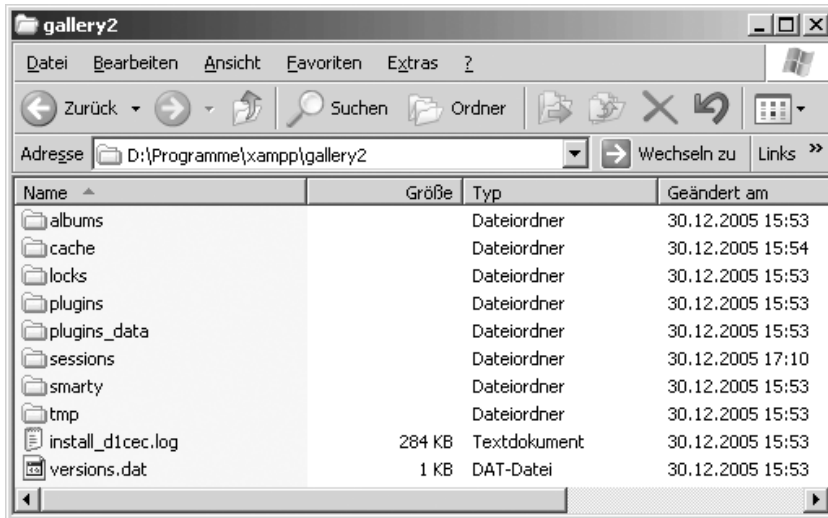


Abbildung 24.19: Das Gallery2-Verzeichnis nach vollendeter Installation unter Windows

4. Um in das Gallery-Setup zu gelangen, rufen Sie nun Gallery2 über `http://localhost/gallery2/` auf. Sollten Sie auf einem Remote-Rechner arbeiten, ändern Sie die URL entsprechend: `http://remotedomain/gallery2/`.

24.3.3 Setup

Das Setup erfolgt ab jetzt über den webbasierten *Gallery Installer 2.0* in elf einzelnen Schritten bzw. *steps*, wie es in Gallery heißt.

1. Authentifizierung

Das Programm möchte sicherstellen, dass wirklich nur Sie derjenige sind, der Zugriff auf den Zielsystem besitzt. Hierfür erzeugt Gallery eine zufällige Zeichenkette (beispielsweise `eae09f2713d626664f9441d6906cbc6b`). Diese Zeichenkette müssen Sie nun in eine Textdatei kopieren und diese mit dem Namen `login.txt` in das Gallery2-Stammverzeichnis speichern. In diesem Beispiel wäre das in das `gallery2`-Verzeichnis im `htdocs`-Bereich von XAMPP. Unter Linux können Sie die Datei bequem mit dem `vim` einrichten. Besondere Zugriffsrechte braucht diese Datei nicht. Unter Windows benutzen Sie hierfür einen einfachen Texteditor wie das *Notepad*.

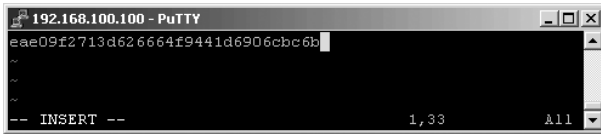


Abbildung 24.20: Die login.txt auf einem Debian-Rechner

2. Systemcheck

Mit dem XAMPP sollte eigentlich alles funktionieren.

3. Installationsart

Wir benutzen hier die STANDARD INSTALLATION. Die zweite Möglichkeit, die MULTISITE INSTALLATION, ist dann sinnvoll, wenn Sie auf einem Webserver mehrere getrennte Auftritte mit separaten Fotoalben einrichten möchten. Diese Installation ist umfangreicher und bedarf einem aktivierten *Rewrite Engine* (*mod_rewrite* muss geladen sein) in der *httpd.conf*. Wie bereits erwähnt, beschränken wir uns hier auf die Darstellung der STANDARD INSTALLATION.

4. Storage-Setup

In diesem Schritt legen Sie das Speicherverzeichnis für Ihre Bilder fest. Gallery schlägt Ihnen ein Verzeichnis namens *g2data* in Ihrem Gallery2-Verzeichnis vor.

Unter Windows legen Sie dieses Verzeichnis zum Beispiel mit dem Windows Explorer an.

Unter Linux verwenden Sie dazu am besten die folgenden beiden Befehle:

```
mkdir /opt/lampp/htdocs/gallery2/g2data  
chmod 777 /opt/lampp/htdocs/gallery2/g2data
```

5. Datenbank-Setup

Hier weisen Sie Gallery nun Ihre Datenbank in MySQL zu. Hierfür haben Sie ja bereits eine Datenbank mit dem Namen *gallery2* eingerichtet. Übrigens können Sie als Datenbankserver neben MySQL auch PostgreSQL oder sogar Oracle einsetzen.

6. Administrativen Benutzer einrichten

In diesem Schritt legen Sie einen Administrator für Gallery an, der später im Konfigurationsbereich neue Benutzer, Fotoalben und so manches mehr verwalten wird. Vergessen Sie den Benutzernamen und das Passwort nicht. Beides brauchen Sie noch!

7. Konfigurationsdatei erstellen

Gallery möchte nun eine Konfigurationsdatei mit allen Einstellungen mit dem Namen *config.php* erstellen. Sie finden diese Datei dann wieder im Stammordner von Gallery. Unter Linux müssen Sie zuvor diese Datei manuell (*touch*) anlegen und mit *chmod 666* bearbeiten. Unter Windows heißt es wieder einmal »die Hände in den Schoß legen«. In diesem Fall geht es also gleich weiter!

Linux:

```
touch /opt/lampp/htdocs/gallery2/config.php
chmod 666 /opt/lampp/htdocs/gallery2/config.php
```

8. Programmkern erstellen

Das Hauptprogramm von Gallery wird erstellt – hoffentlich erfolgreich, denn eingreifen können Sie hierbei nicht.

9. Installation zusätzlicher Module

Gallery bietet Unterstützung für zahlreiche weitere Programme oder Module insbesondere in den Bereichen Grafikbearbeitung und Darstellung an. Diese Module sind zwar nicht zwingend erforderlich, können aber die Funktionalität Ihrer Fotoalben erheblich steigern. Viele Module können aber erst aktiviert werden, wenn Sie das dazugehörige Hauptprogramm auf Ihrer Plattform installiert haben.

10. Sicherheitscheck

Das betrifft wie so oft insbesondere die Linux-Anwender. Hier rät Gallery Ihnen, den Zugriff auf die `config.php` einzuschränken. Diesem Ratschlag folgen Sie natürlich gerne.

Linux:

```
chmod 666 /opt/lampp/htdocs/gallery2/config.php
```



Abbildung 24.21: Der Sicherheitshinweis von Gallery unter Linux

Fertig! Nun können Sie Gallery das erste Mal aufrufen. Aber bitte erschrecken Sie nicht! Das erste Mal sieht alles sehr leer und fad aus. Noch fehlen die Fotoalben und auch von einer Gestaltung kann keine Rede sein (Stichwort: *Themes*). Das alles können Sie nun in Ruhe einrichten, wenn Sie sich über LOGIN als Administrator in die Konfigurationsebene von Gallery2 einloggen.

24.3.4 Konfiguration

In der Konfigurationsebene, der *Gallery Site Administration*, machen Sie den Feinschnitt. Dazu folgen Sie zu Beginn dem Link in die Site-Administration. Dort legen Sie im Abschnitt *Allgemein* alle grundlegenden Einstellungen fest.

Spracheinstellungen

Hier legen Sie Ihre bevorzugte Sprache fest.

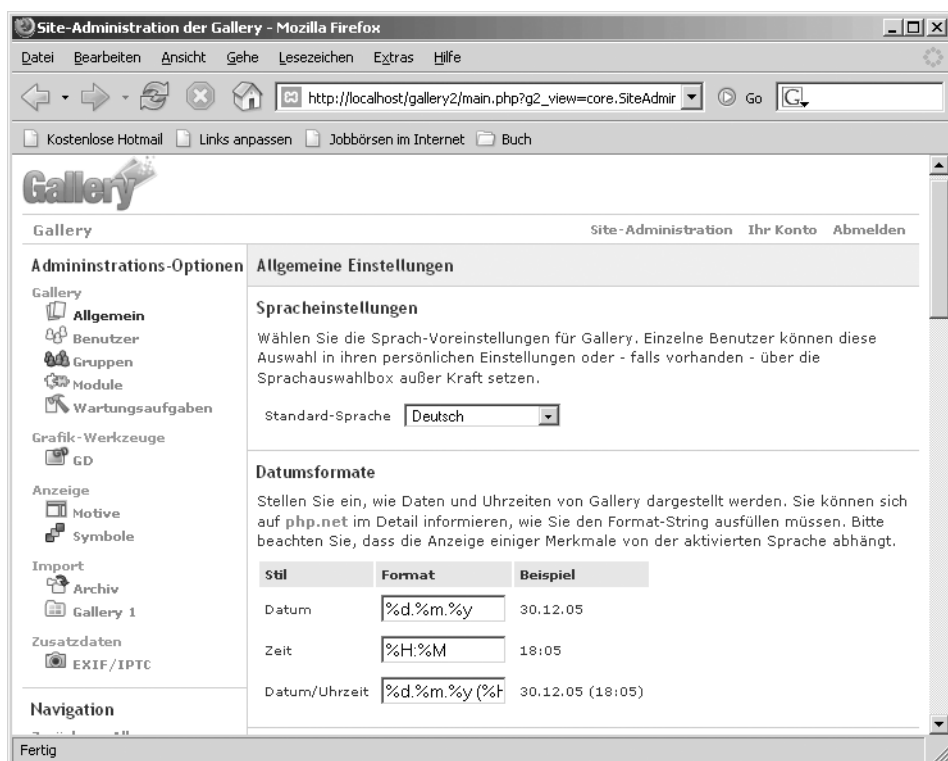


Abbildung 24.22: Die Site-Administration von Gallery2

Datumsformate

Hier definieren Sie die Datums- und Zeitformate mit der PHP-Funktion `strftime`. Für eine deutsche Sprachwahl ist selbstverständlich ein deutsches Zeitformat zu empfehlen. Beispiel:

Datum: `%d.%m.%y` 30.12.05

Zeit: `%H:%M`16:13

Datum/Uhrzeit: `%d.%m.%y (%H:%M)`30.12.05 (16:13)

Hochlade-Pfad auf dem Server

Der Pfad, in dem Benutzer ihre Fotos hochladen und ablegen, um sie danach in Gallery zu importieren. Unter Linux muss der Ordner für den Zugriff freigegeben sein. Beispiel:

Linux: /tmp

Windows: D:\Programme\xampp\tmp

Sitzungseinstellungen

Wie lange soll nach dem erfolgreichen Login die Sitzung (Session) Gültigkeit besitzen? Gallery2 ist hier sehr großzügig. Die Sitzungslaufzeit ist ewig, die Zeitüberschreitung bei Untätigkeit beträgt 2 Wochen. Das schreit ja geradezu nach einer Einschränkung. Beispiel:

Sitzungslaufzeit: 6 Stunden

Zeitüberschreitung bei Untätigkeit: 1 Stunde

Eingebettete Auszeichnung

Die Benutzer können für ihre zu betreuenden Galerien einzelne Bilder mit Unterschriften und Kommentaren versehen. Die Schrift kann dabei verändert werden (fett, kursiv usw.). Die Auszeichnung dieser Formatierung kann über BBCode (wie `[B]`) oder über HTML (wie `<b>`) geschehen. Oder Sie verbieten diese Auszeichnung einfach ganz.

E-Mail

Hier definieren Sie einen externen E-Mail-Server (SMTP) zum Versenden Ihrer E-Mails. Sollten diese Felder leer bleiben, versucht Gallery, über die Mailfunktion von PHP zu versenden. Hierfür sollten Sie sicherstellen, dass Ihr E-Mail-Server ordnungsgemäß funktioniert.

Cookies

Gallery arbeitet per Standard mit angehängtem *GALLERYSID-String*. Alternativ kann Gallery2 zur Authentifizierung mit einer verdeckten Cookie-Methode arbeiten d. h., es kann an eine Cookie-Methode einer weiteren Webapplikation wie CMS oder Foren gebunden werden. In den meisten Fällen reicht die Definition des Werts `Pfad` aus, um

den Informationstransfer zu gewährleisten. Der Pfad entspricht dem Verzeichnispfad innerhalb des DocumentRoot-Verzeichnisses des Apache, in diesem Fall also `/gallery2/`.

Die *Cookie-Domäne* ist schließlich dann notwendig, wenn Gallery im Vergleich zu der Anwendung über eine andere Subdomain erreichbar ist. Ist die Anwendung unter <http://www.beispiel.de> und Gallery unter der URL <http://photos.beispiel.de> erreichbar, ist die Cookie-Domain *beispiel.de* (*Second-Level-Domain.Top-Level-Domain*).

Sperrungs-Methode

Bei zahlreichen Benutzern, die Zugriff auf dieselben Galerien haben, kann es vorkommen, dass sich die Bildbearbeitung an den einzelnen Objekten überlappt. Um das zu verhindern, sperrt Gallery den Zugriff bis zum Abschluss der Arbeiten des aktiven Benutzers. Die Methode dieser Sperrungen ist standardmäßig an eine Datenbank gekoppelt (*LOCK*). Das Sperren kann auch über eine Textdatei organisiert werden, wird hier aber nicht empfohlen.

Nun haben Sie die Grundeinstellung erfolgreich hinter sich gebracht. Jetzt kann es in die Praxis gehen. Hierfür haben die Entwickler von Gallery eine umfangreiche Dokumentation im Internet bereitgestellt: <http://codex.gallery2.org>.

Doch zum Schluss noch ein paar Sätze zum strukturellen Aufbau von Gallery. Eine Galerie besteht aus verschiedenen Alben. Wenn Sie nun Bilder hineinstellen (d.h. hochladen) möchten, ist es sinnvoll, zuvor ein neues Album hierfür anzulegen. Die angelegten Alben können Sie selbstverständlich später wieder löschen. Also scheuen Sie sich nicht, ein oder mehrere Testalben einzurichten. Erst danach ist es sinnvoll, Benutzer und Benutzergruppen zu definieren. Denn der Kern der Benutzerverwaltung ist die Festlegung der Zugriffsrechte auf die einzelnen Alben. Und das ist ohne Alben eben schwer zu realisieren.

24.4 SquirrelMail

SquirrelMail (*Webmail for Nuts*) ist ein freier PHP-Webmailer insbesondere für den IMAP-Server. Neben den weiteren Protokollen wie POP3 und SMTP unterstützt SquirrelMail auch SSL-Übertragungsarten. Die Homepage lautet <http://www.squirrelmail.org>.

Tipp



Ein weiterer Webmailer nicht zuletzt für POP3-Server ist *NOCC*. Die Homepage lautet <http://nocc.sourceforge.net>. NOCC können Sie ebenfalls frei beziehen und ist in PHP programmiert.

Im XAMPP für Linux fehlt ein E-Mail-Serverpaket und damit auch ein IMAP-Server. Also müssen Sie für einen lokalen Test sicherstellen, dass Sie auf Ihrem System einen funktionstüchtigen IMAP-Server installiert haben. Oder Sie blättern für die IMAP-Konfiguration gleich zum Abschnitt »Verwendung von SquirrelMail mit einem externen IMAP-Server«. Die gängigen freien IMAP-Server unter Linux waren zum Zeitpunkt der Arbeit an diesem Buch der einfachere WU IMAPD (University of Washington IMAPD, Homepage: <http://www.washington.edu/imap/>) und der komplexere, aber auch flexiblere Cyrus IMAP Server (Homepage: <http://asg.web.cmu.edu/cyrus/imapd/>).

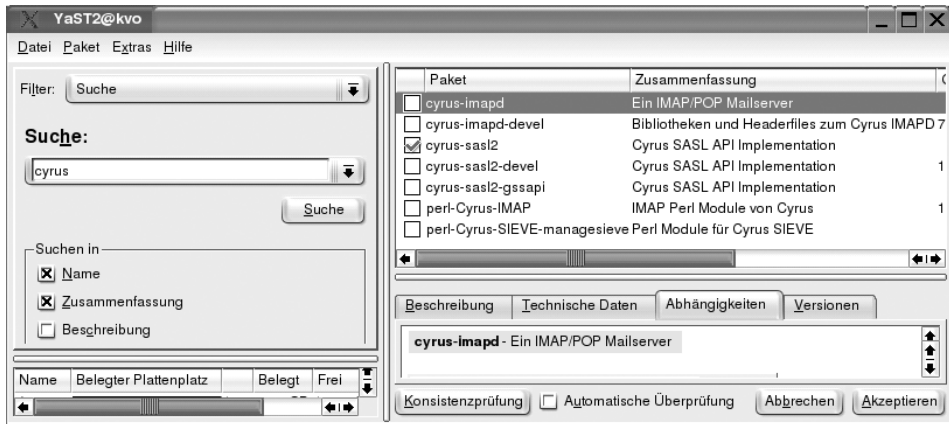


Abbildung 24.23: Das Cyrus-Paket in der YaST2-Paketverwaltung

Mit der apt-Paketverwaltung geht es selbstverständlich auch. In diesem Beispiel wurden die zum Zeitpunkt der Drucklegung dieses Buches aktuellen IMAP-Server zur Installation verwendet.

WU IMAPD:

```
apt-get install wu-imapd
```

Cyrus:

```
apt-get install cyrus23-common
```

24.4.1 Entpacken unter Linux

Nach dem Download extrahieren Sie das SquirrelMail-Paket direkt in das `htdocs`-Dokumentenverzeichnis und benennen das neue Verzeichnis danach am besten gleich um.

```
tar xvfz squirrelmail-x.y.z.tar.gz -C /opt/lampp/htdocs
mv /opt/lampp/htdocs/squirrelmail-x.y.z.tar.gz /opt/lampp/htdocs/squirrelmail
```

24.4.2 Entpacken unter Windows

Nach dem Download der `squirrelmail-x.y.z.zip` packen Sie das Paket einfach direkt in den `htdocs`-Ordner aus. Am besten Sie benennen das Verzeichnis `squirrelmail-x.y.z` in `squirrelmail` um, um die URL zu vereinfachen.

24.4.3 Setup

Jetzt schreiten Sie zur Konfiguration, die sich nur wenig zwischen den Plattformen unterscheidet.

1. Passen Sie die `php.ini` unter Windows an.
Vergewissern Sie sich, dass das PHP-IMAP-Modul in der `\xampp\apache\bin\php.ini` geladen wird.

```
extension=php_imap.dll
```

Hinweis



Unter *Windows 95/98/ME* kann es unter Umständen Probleme mit dem Laden der IMAP-Erweiterung geben.

2. Erstellen Sie die `config.php`.
Linux: `/opt/lampp/htdocs/squirrelmail/config/config_local.php`
Windows: `\xampp\htdocs\squirrelmail\config\config_local.php`
3. Öffnen Sie die Datei `config_local.php` und speichern Sie diese als `config.php` ab. In der neu erstellten `config.php` passen Sie die Werte an Ihre E-Mail-Konfiguration an. Hier arbeiten wir mit Mercury als IMAP-Server. In diesem Fall brauchen Sie so gut wie keine Anpassungen in der `config.php` vorzunehmen.
Benutzen Sie unter Linux `sendmail` als SMTP, dann:

```
$useSendmail = true;
```

In diesem Beispiel für Windows arbeiten wir nicht mit `sendmail`, sondern mit Mercury als SMTP. Unter Linux verwenden Sie auch kein `sendmail`? Dann:

```
$useSendmail = false;
```

Alle E-Mails werden lokal mit `localhost` versendet.

```
$smtpServerAddress = 'localhost';
```

Natürlich, denn unser IMAP-Server ist auf dem lokalen Rechner.

```
$imapServerAddress = 'localhost';
```

Die URL zu SquirrelMail auf unserem Server lautet:

```
$provider_uri = 'http://localhost/squirrelmail/';
```

Wenn Sie nicht genau wissen, was für einen IMAP-Server Sie verwenden, dann setzen Sie die Variable `$imap_server_type` auf den Wert `other`.

```
$imap_server_type = 'other';
```

Damit wird SquirrelMail auf alle Fälle funktionieren. Allerdings wird SquirrelMail damit nicht in der Lage sein, auf besondere Features des IMAP-Servers einzugehen.

Für Profis ist sicherlich eine feste Definition des IMAP-Servers interessant. Eine Liste der von SquirrelMail unterstützten IMAP-Server finden Sie in der Datei `squirrelmail/doc/presets.txt`. Für den Mercury im XAMPP für Windows heißt die korrekte Definition der Variablen:

```
$imap_server_type = 'mercury32';
```

Der Link zu unserem SquirrelMail muss für einen externen Zugriff angepasst werden:

<http://www.meinedomain.org/squirrelmail/>

3. Unterziehen Sie den Webmailer einer Funktionsprüfung.



Abbildung 24.24: Das Login von SquirrelMail im XAMPP für Windows

Nun ist es Zeit für den ersten Kontakt. Das Login finden Sie unter der URL <http://localhost/squirrelmail>. In dem Windows-Beispiel melden Sie sich am Mercury Mail Server an. Sie erinnern sich vielleicht aus Kapitel 7, »XAMPP und E-Mail«: Der Testbenutzer hatte den Namen *newuser* mit dem Passwort *wampp*. Unter Linux nutzen Sie den von Ihnen konfigurierten E-Mail-Server. Oft ist hier die Anmeldung an den Server identisch mit der Anmeldung Ihres Linux-User-Accounts.

Erfolgreich angemeldet, sollten Sie über das Menü OPTIONS SquirrelMail individuell anpassen: Sprache, Schriftgröße, Style und vieles mehr. Hierbei sollten Sie den Link *Refresh Page* öfter mal betätigen. Dieser setzt insbesondere Ihre Veränderungen im Layout sofort um.

Ein Sendetest gefällig? Senden Sie doch eine E-Mail an den Mercury-Benutzer *admin* (*admin@localhost*). Im geöffneten Mercury Mail Server im XAMPP für Windows können Sie übrigens in Echtzeit nachvollziehen, ob der SMTP-Server ordnungsgemäß den Auftrag abgearbeitet hat. Natürlich prüfen Sie nun den E-Mail-Eingang mit Squirrel-Mail. Unter Windows melden Sie sich mit dem Benutzer *admin* an. Haben Sie bereits für *admin* ein Passwort gesetzt? Dann benutzen Sie dieses. Ansonsten besitzt der Benutzer *admin* kein Passwort. Also lassen Sie das Passwort-Feld einfach frei. Erfolgreich angemeldet, sehen Sie nun unter INBOX die E-Mail des Benutzers *newuser@localhost*.

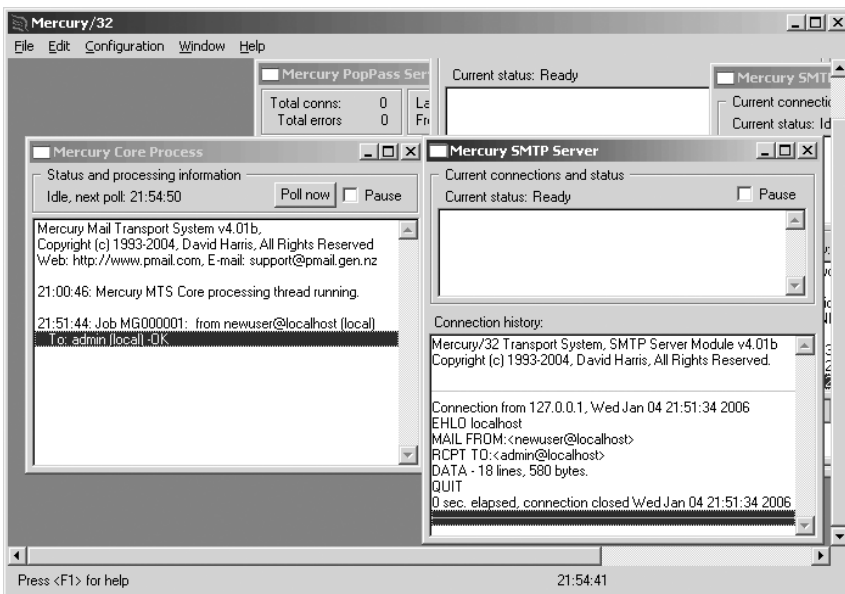


Abbildung 24.25: Das Versenden über SquirrelMail live im Mercury für Windows nachvollzogen

Da im XAMPP für Linux aus guten Gründen kein E-Mail-Server beiliegt, können hier auch keine konkreten Ratschläge gegeben werden. Als Funktionstest senden Sie sich selbst oder einem nicht wesentlichen Benutzer eine E-Mail und überprüfen den Eingang.

24.4.4 Verwendung mit einem externen IMAP-Server

Vielfach ist es aber einfach praktischer, den externen E-Mail-Server Ihres Providers oder Hosters zu benutzen. Sie setzen beispielsweise XAMPP zum Hosten von SquirrelMail ein, verwenden aber einen externen IMAP- und SMTP-Server. Und auch das funktioniert sehr gut mit SquirrelMail.

1. Konfigurieren Sie die `config.php` für den externen IMAP-Server.

Passen Sie Ihre externe IMAP-Domain an:

```
$domain = 'meinedomain.de';
```

Der IMAP-Server heißt:

```
$imapServerAddress = 'imap.provider.de';
```

Der Login-Mechanismus ist meist:

```
$imap_auth_mech = 'login';
```

Weitere Einstellungen der Variablen `$imap_auth_mech` und `$smtp_auth_mech` sind 'none', 'login', 'plain', 'cram-md5', oder 'digest-md5'.



Achtung

Der Login-Name ist vielfach nicht einfach die E-Mail-Adresse! Oft besitzen die Postfächer eigene Namen, und einzig und allein diese werden akzeptiert.

2. Konfigurieren Sie die `config.php` für den externen SMTP-Server.

Der SMTP-Server heißt:

```
$smtpServerAddress = 'smtp.provider.de';
```

Der Login-Mechanismus ist meistens:

```
$smtp_auth_mech = 'login';
```

Öffentliche SMTP-Server, die ohne Benutzerauthentifizierung versenden, gibt es glücklicherweise sehr wenige. Oft wird aber ein SMTP-Server mit `localhost`-Einschränkung (Standard z. B. bei Postfix) für das Versenden verwendet. Hierbei ist ein SMTP-Login nicht notwendig, weil man davon ausgeht, dass der bereits legitimierte Benutzer am lokalen Rechner vertrauenswürdig ist. Existieren also der SMTP-Server und XAMPP auf dem gleichen Rechner, heißt es oft:

```
$smtp_auth_mech = 'none';
```

Achtung



Wenn der Server nur den Namen des Postfachs akzeptiert, sendet dieser auch mit dem Postfachnamen: *postfachname@ihredomain.de*. Das ist wahrlich nicht hübsch und sollte geändert werden. Melden Sie sich also auf SquirrelMail an und gehen Sie in das OPTIONS-Menü. Im ersten Abschnitt PERSONAL INFORMATION geben Sie Ihre E-Mail-Adresse an. Danach versendet SquirrelMail standardmäßig mit der von Ihnen hier angegebenen Adresse.

Testen Sie bitte das Versenden mit der neuen Konfiguration in SquirrelMail. Häufig sehen Sie nach dem Versenden die folgende merkwürdige Meldung



Abbildung 24.26: Merkwürdige Meldung von SquirrelMail nach dem Versenden

Alles umsonst? Nicht gleich die Flinte ins Korn werfen! Diese Meldung ist im strengen Sinne keine Fehlermeldung. Hier hat der externe SMTP-Server eine Antwort beispielsweise über den erfolgreichen Vollzug der Versendung zurückgegeben, die SquirrelMail allerdings nicht verstanden hat. Also schauen Sie erst einmal, ob das Versenden tatsächlich erfolgreich war. Oft ist alles gut gegangen.

24.5 phpBB

Mit der freien und kostenlosen Forensoftware *phpBB* erhalten Sie eine umfangreiche Kommunikationsplattform für Ihren Webauftritt. phpBB ist, wie der Name schon suggeriert, in PHP programmiert und benutzt zusätzlich einen Datenbankserver, meist MySQL. Die URL der Homepage lautet: <http://www.phpbb.com>. Wir stellen hier phpBB vor, da wir es sehr schätzen und es selbst für unser Apache Friends-Forum unter <http://www.apachefriends.org/f/> einsetzen.



Exkurs

Ist phpBB unsicher?

Wir wollen aber auch nicht verschweigen, dass es kritische Stimmen zu phpBB gibt. So wird zum Beispiel oft bemängelt, dass relativ oft neue Versionen von phpBB erscheinen, um Sicherheitsprobleme in der bestehenden Version zu beheben. Das ist aber auch damit zu erklären, dass phpBB sehr viel öfter eingesetzt wird als vergleichbare andere Forenprogramme und damit einer erhöhten Aufmerksamkeit ausgesetzt ist.

Interessant ist auch das deutschsprachige Supportforum, erreichbar unter der URL <http://www.phpbb.de>. Hier finden Sie neben zahlreichen Anleitungen und unterstützenden Foren auch Sprachpakete für die Umstellung von phpBB2 in ein deutschsprachiges Layout.

Zum Zeitpunkt der Erstellung dieses Buches war phpBB in der Version 2 (kurz: *phpBB2*) aktuell. In der Betaphase befand sich die Version *phpBB3*.

24.5.1 Installation

Linux

In diesem Beispiel kommt phpBB2 in der Version 2.0.19 (*phpBB-2.0.19.tar.gz*) zum Einsatz. Optional können Sie zusätzlich deutschsprachige Sprach- sowie Grafikpakete von der Webseite <http://www.phpbb.de/download.php> installieren.

1. Entpacken Sie den phpBB2-Quellcode:

```
tar xvfz phpBB-2.0.19.tar.gz -C /opt/lampp/htdocs
```

2. Erstellen Sie eine phpBB2-Datenbank.

Zur Erstellung der Datenbank bevorzugen die Konsolenliebhaber selbstverständlich den MySQL-Monitor:

```
mysql>CREATE DATABASE phpbb2;
```

Oder Sie benutzen zum Anlegen der Datenbank alternativ ein grafisches Frontend wie das webbasierte *phpMyAdmin*.

3. Setzen Sie die Benutzerrechte.

Damit Apache zugreifen darf, wechseln Sie mit `chown` den Besitzer des phpBB2-Verzeichnisses. Neuer Besitzer ist nun der Apache-Standardbenutzer *nobody* in der *root*-Gruppe.

```
chown -R nobody.root /opt/lampp/htdocs/phpBB2
```

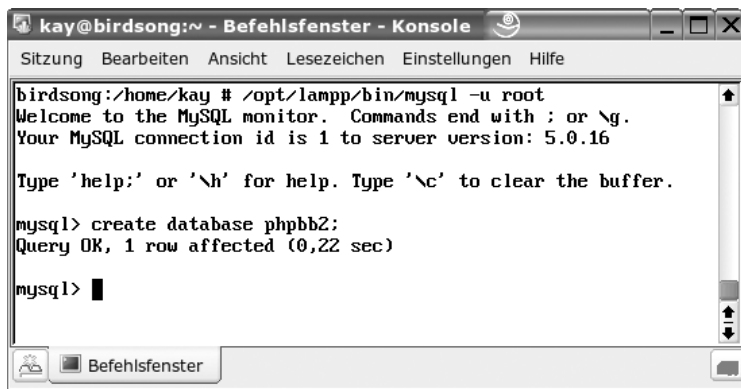


Abbildung 24.27: Eine Datenbank für phpBB2 über den MySQL-Monitor erstellen

4. Konfigurieren Sie die Installation von phpBB2.

Weiter geht es jetzt über die URL <http://localhost/phpBB2/>. Konfigurieren Sie hier den MySQL-Zugang und den Administrationsaccount.

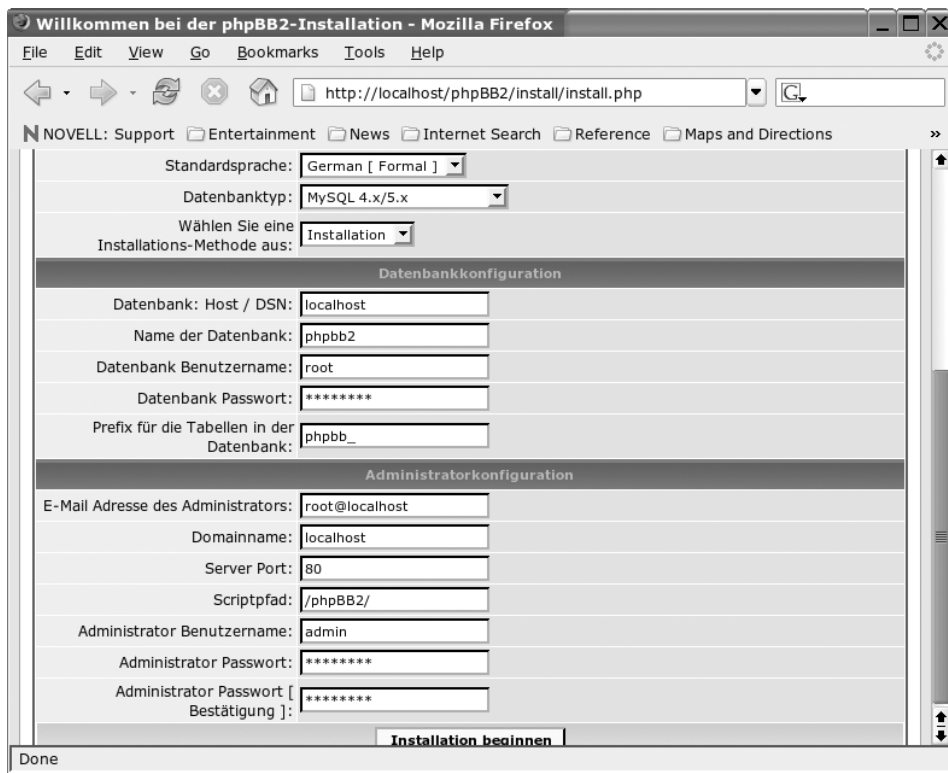


Abbildung 24.28: Die Konfiguration von phpBB2

5. Löschen Sie die Verzeichnisse `install` und `contrib`.

```
rm -R /opt/lampp/htdocs/phpBB2/install
rm -R /opt/lampp/htdocs/phpBB2/contrib
```



Abbildung 24.29: Die Verzeichnisse `install` und `contrib` müssen gelöscht werden.

Windows

Die Installation unter Windows ist im Prinzip identisch mit dem Vorgehen unter Linux. Hier kommt ebenfalls phpBB2 in der Version 2.0.19 (phpBB-2.0.19.zip) zum Einsatz.

1. Entpacken Sie die phpBB2-ZIP-Datei in das Verzeichnis `htdocs` in ihrer XAMPP-Installation.
2. Erstellen Sie eine phpBB2-Datenbank.

Zur Erstellung der Datenbank bevorzugen die Konsolenliebhaber selbstverständlich den MySQL-Monitor.

```
mysql>CREATE DATABASE phpbb2;
```

Alternativ benutzen Sie hierfür einfach das webbasierte phpMyAdmin.

3. Konfigurieren Sie die Installation von phpBB2:
Weiter geht es jetzt über die URL `http://localhost/phpBB2/`. Konfigurieren Sie hier den MySQL-Zugang und Ihren Administrationsaccount.
4. Löschen Sie die Verzeichnisse `install` und `contrib`.

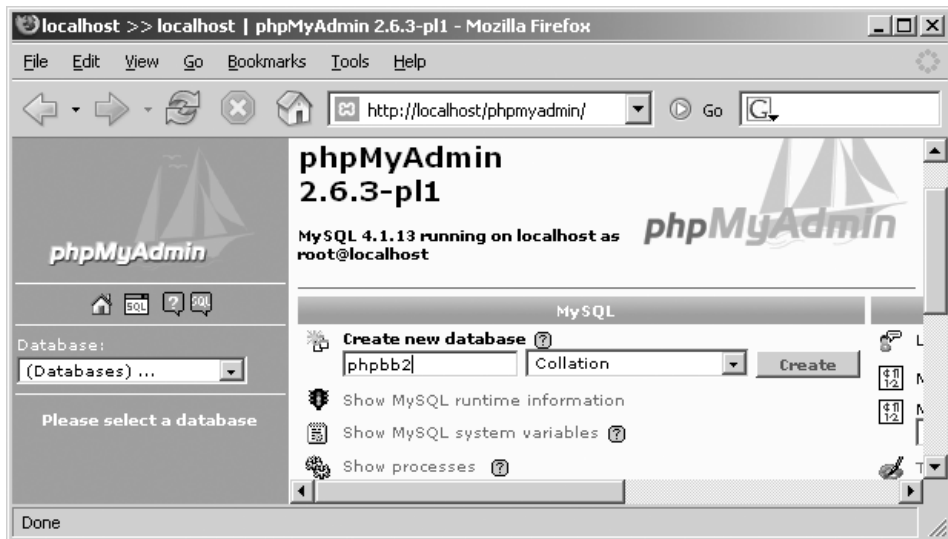


Abbildung 24.30: Die Erstellung der Datenbank phpbb2 über phpMyAdmin

Löschen Sie die Verzeichnisse `install` und `contrib` über den Explorer. Oder benutzen Sie als Freund der guten alten Windows-Eingabeaufforderung `del`:

```
del /S <xampp>\htdocs\phpBB2\install
del /S <xampp>\htdocs\phpBB2\contrib
```

Bei beiden Plattformen gelangen Sie zum Abschluss der Installation in das Login von phpBB2. Hier können Sie sich als administrativer Benutzer anmelden, und schon landen Sie bei der Konfiguration. Die URL lautet <http://localhost/phpBB2/>, wo die Entwickler schon ein Testforum für Sie vorbereitet haben.

24.5.2 Konfiguration

Wie bei allen Beispielkonfigurationen können wir hier nicht alle Facetten der Applikation darstellen. Es werden also auch hier wieder lediglich Teilbereiche und wichtige Fragestellungen behandelt, die wir im Rahmen eines XAMPP-Buches für wichtig halten. Eine sehr gute weiterführende Dokumentation in deutscher Sprache finden Sie aber unter:

<http://www.phpbb.de/doku/>

Ein klein wenig Forenphilosophie

Vielleicht hier kurz ein paar Sätze zu den strukturellen Merkmalen eines phpBB2-Forums. Eine Forum-Plattform kann mehrere Foren enthalten. Diese heißen bei phpBB2 *Kategorien*. Die Kategorien können hingegen selbst weitere Unterkategorien enthalten, die dann meist nach thematischen Schwerpunkten erstellt werden. Diese Unter-

kategorien sind die eigentlichen Foren (Singular: *Forum*). Verstanden? Leichter geht es mit einem Beispiel, und zwar dem Apache Friends-Forum (kurz: *AForum*) mit der URL <http://www.apachefriends.org/f/>.

Hier ein Ausschnitt unserer Plattform in tabellarischer Form, der Plattform *AForum* mit den Kategorien *Apache Friends* sowie *XAMPP* und den zugehörigen Foren.

Plattform	Kategorien	Foren
AForum	Apache Friends	Apache Friends
AForum	Apache Friends	Ankündigungen und Neuigkeiten
AForum	Apache Friends	Off Topic
AForum	XAMPP	XAMPP für Windows
AForum	XAMPP	XAMPP für Linux
AForum	XAMPP	XAMPP für Mac OS X
AForum	XAMPP	XAMPP für Solaris
AForum	XAMPP	(und weitere Foren)

Tabelle 24.1: Die Struktur des Apache Friends-Forums

Wichtig sind die drei Ebenen von phpBB2.

■ *Ebene 1: Öffentliche Ebene*

Jeder darf in den öffentlichen Foren lesen. Ein Interessent kann sich hier als Forumsbenutzer anmelden, um auch schreiben zu dürfen.

■ *Ebene 2: Registrierte Ebene*

Wer bereits registrierter Benutzer ist, darf über das Login diese Ebene betreten. Je nach Berechtigung kann der Benutzer hier Beiträge (*Threads*) schreiben, verändern oder löschen (Letzteres gilt für Moderatoren). Und Forum-Administratoren weisen hier neue Benutzer einer Gruppe zu. Die gesamte kommunikative Arbeit passiert in der Regel hier.

■ *Ebene 3: Administrations-Bereich*

Diesen darf eigentlich nur der Admin-Benutzer betreten. Der Administrations-Bereich enthält die eigentliche Kernkonfiguration von phpBB2.

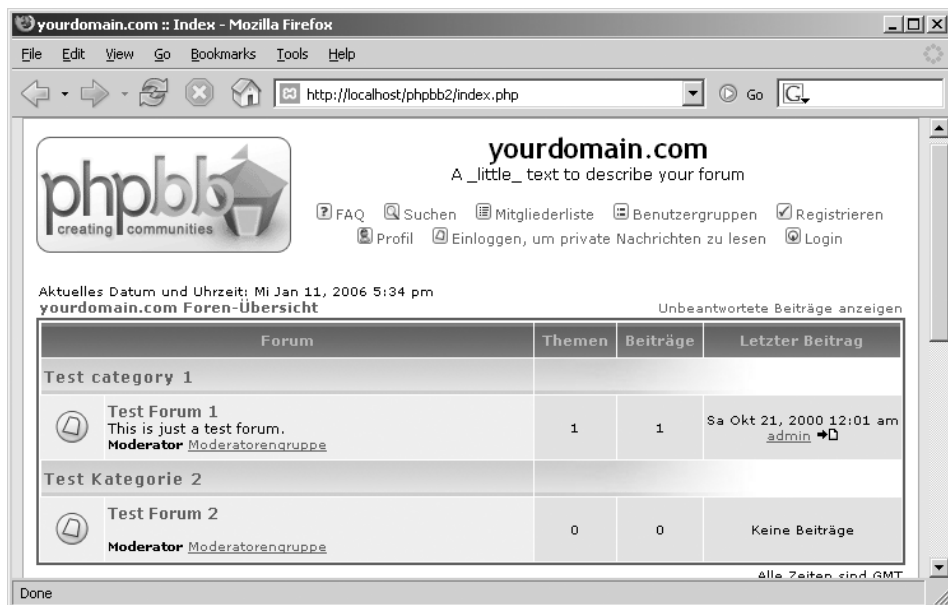


Abbildung 24.31: Die öffentliche Ebene von phpBB

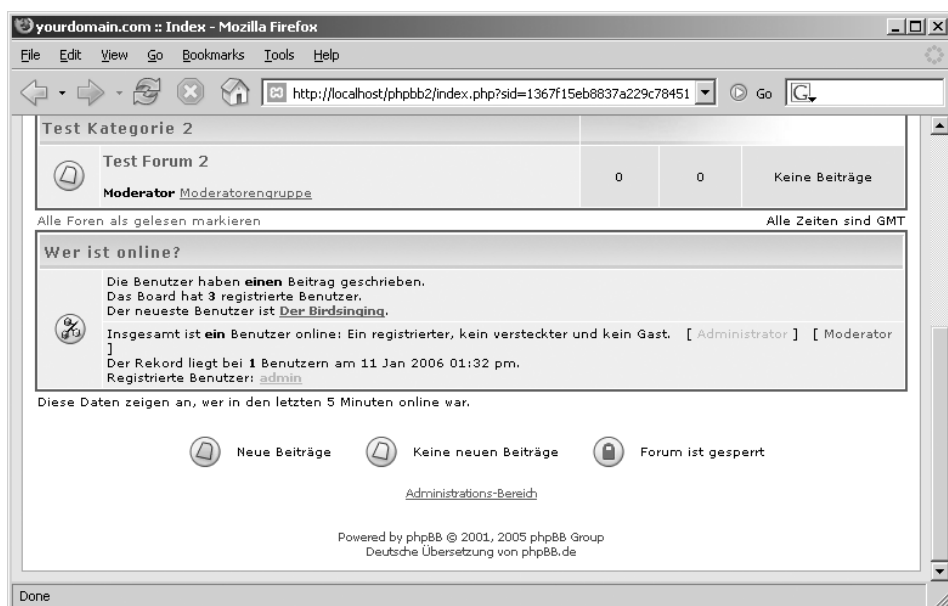


Abbildung 24.32: Über die registrierte Ebene geht es weiter zum Administrations-Bereich.

**Achtung**

Sie kommen erst in den Administrations-Bereich, wenn Sie sich bereits in der registrierten Ebene befinden.

Administrations-Bereich

Den wirklichen Kern der Konfiguration enthält der ADMINISTRATIONS-BEREICH. Hier können Sie die Serverdaten anpassen, Benutzergruppen sowie Foren und Unterforen erstellen. Weiterhin können Sie hier Ihre Sicherheitseinstellungen überprüfen. In diesem Bereich finden Sie darüber hinaus noch zahlreiche weitere Einstellungen, die hier gar nicht alle aufgezählt werden können.

Zu Beginn sollte der Bereich ALLGEMEINE BOARDEINSTELLUNGEN unter die Lupe genommen werden. In diesem Menü können Sie nicht nur die allgemeinen Server- und Forumseinstellungen definieren. Sie haben hier unter anderem auch die Möglichkeit, einen SMTP-Server beispielsweise für den Versand von Gruppenmails an alle Forumsmitglieder einzurichten. Dabei können Sie auch einen externen SMTP-Server angeben.

**Achtung**

Bitte vernachlässigen Sie nicht die Einrichtung des SMTP-Servers für Ihre E-Mail-Adresse. Neben dem Massenversand werden hierüber auch die Bestätigungen der Anmeldungen versendet, wenn Sie in der Konfiguration *Account-Freischaltung per E-Mail* definiert haben.

Forum Administration

Standardmäßig ist im Bereich FORUM ADMINISTRATION bereits eine Kategorie *Test category 1* mit dem Forum *Test Forum 1* zu finden. Hier können Sie noch weitere Kategorien oder Foren anlegen, verändern oder löschen.

Gruppenadministration

Langsam wird es für Sie Zeit, Gruppen zu definieren. Das geschieht in der Gruppenadministration. Eine Gruppe sollte dabei eine einfache *Benutzergruppe* sein. Eine weitere Gruppe soll dann aus *Moderatoren* bestehen. Der Unterschied zwischen einfacher Gruppenzugehörigkeit und Moderatoren besteht darin, dass Moderatoren Topics

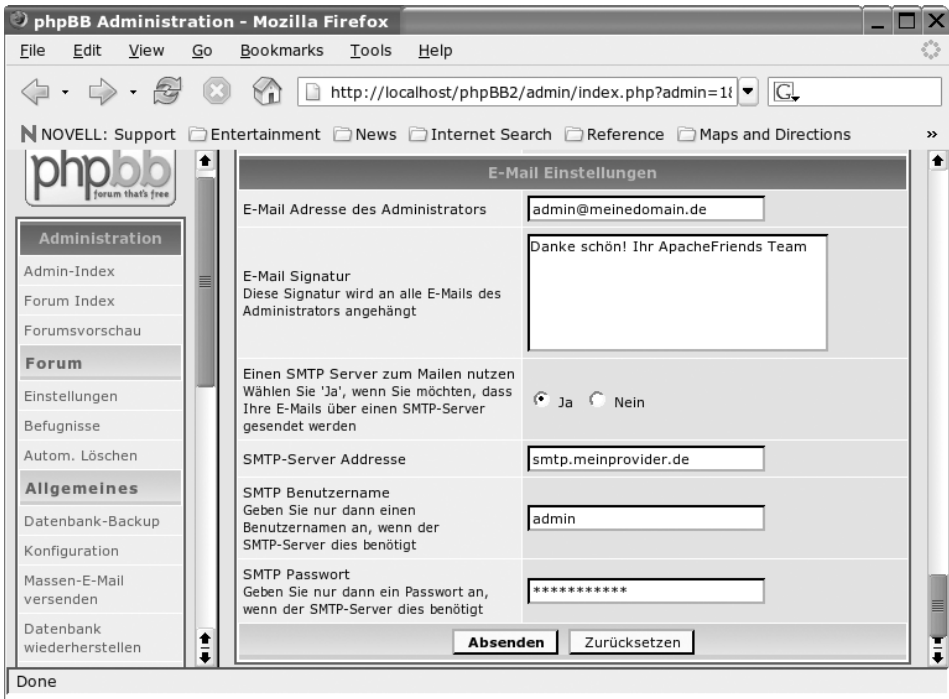


Abbildung 24.33: Die E-Mail-Konfiguration in den Allgemeinen Boardeinstellungen des Administrations-Bereichs bitte nicht vergessen!

erstellen, verändern und löschen können. Einfache Mitglieder des Forums dürfen hingegen Topics erstellen und ihre eigenen Beiträge ggf. auch verändern, aber eben nicht löschen oder globale Veränderungen im Forum vornehmen.

Zum Abschluss der Gruppenkonfiguration klicken Sie sich im Abschnitt GRUPPEN in die GRUPPENBEFUGNISKONTROLLE, um hier die Moderatorengruppe auszuwählen und in einem nächsten Schritt diese Gruppe als Moderatoren beispielsweise für das *Test Forum 1* einzurichten.

Einen neuen Benutzer anlegen

Aber wo nur? Jeder Einsteiger in das »Forumsgeschäft« steht wohl vor dieser Frage. Im Administrations-Bereich geht das nicht, so viel vorweg. Dass Sie keine Benutzer in diesem Bereich anlegen können, hat mit der Logik eines Forums zu tun. Denn jedem Interessenten ist es, soweit das Forum öffentlich ist, freigestellt, sich selbst anzumelden. Und das geschieht in der eigentlichen Forumsansicht unter dem Link *Registrieren*. Zu Beginn sind Sie der Befugte, der die Registrierung annimmt oder ablehnt. Den neuen Benutzer können Sie dann und erst dann im Administrationsbereich in eine Gruppe stecken und ihm darüber hinaus noch weitere Privilegien zubilligen.

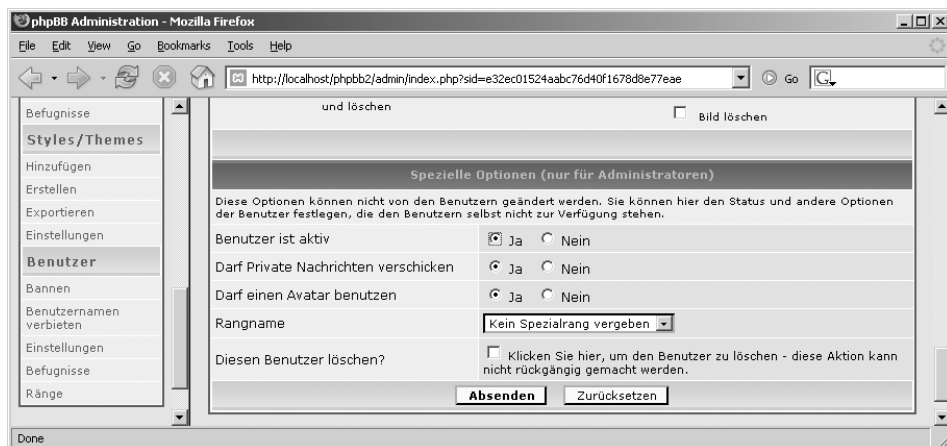


Abbildung 24.34: Einen neuen Benutzer im Administrations-Bereich aktivieren

Sie wollen also einen Bekannten für Ihr Forum als Moderator einsetzen? Dann registrieren Sie sich mit seinem Namen und seiner E-Mail-Adresse über die öffentliche Forumsansicht, nehmen im Administrations-Bereich diese Anmeldung an und weisen ihn schließlich im registrierten Bereich einer Gruppe zu. Voraussetzung ist die *Account-Freischaltung* durch den *admin*. Der angemeldete Benutzer erhält in der Regel mindestens eine E-Mail, die die Anmeldung bzw. die erfolgte Aufnahme bestätigt. Der Benutzer sollte also schon wissen, dass Sie ihn für Ihr Forum angemeldet haben. Das ist übrigens ein wesentlicher Grund, warum es so wichtig für die Funktionsweise eines Forums ist, einen wirksamen E-Mail-Server über den Administrations-Bereich definiert zu haben.

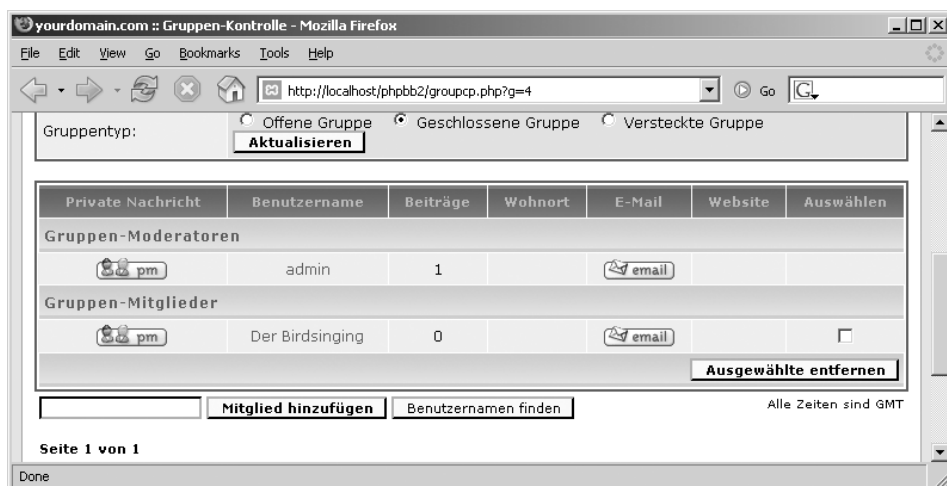


Abbildung 24.35: Einen Benutzer einer Gruppe im registrierten Bereich hinzufügen

**Tipp**

Überprüfen Sie alles mit einem Testbenutzer. Das ist die beste Art und Weise, letztendlich die gesamte Funktionalität Ihres Forums zu gewährleisten.

Eine andere Forum-URL

Vielleicht möchten Sie nicht die Erreichbarkeit Ihres Forums über die URL `http://www.ihredomain.org/phpBB2/` realisieren, sondern beispielsweise über die URL `http://www.ihredomain.org/meinforum/`. Der einfachste und sicherste Weg ist ein Alias in der Apache-Konfiguration. Hierfür ergänzen Sie die `httpd.conf` bzw. `httpd-xampp.conf` oder, wenn Sie einen virtuellen Host definiert haben, direkt den *VirtualHost*-Container (meist in der `httpd-vhosts.conf`) durch folgende Zeile:

```
Alias /meinforum "/opt/lampp/htdocs/phpBB2/"
```

Oder als Beispiel unter Windows:

```
Alias /meinforum "D:/Programme/xampp/htdocs/phpBB2/"
```

Um die Änderungen an der `httpd.conf` wirksam werden zu lassen, müssen Sie nun nur noch den Apache neu starten.

Einen anderen Style einsetzen

Nun ist ja der Standardstyle *subSilver* gar nicht so schlecht. Aber manche wollen es eben exklusiver haben. Dann rufen Sie die Style-Page der deutschen phpBB-Sektion auf: `http://www.phpbb.de/styles/`. Wichtig ist bei der Auswahl neuer Styles die Kompatibilität zu Ihrer phpBB-Version. In diesem Beispiel benutzen wir ja die Version 2. Die gültige Version des Styles steht übrigens auf der Übersichtsseite, auf der Sie ebenfalls die Links zu den Downloads finden. Neben dem eigentlichen Style sind oft auch weitere Sprachpakete verfügbar. In diesem Beispiel entscheiden wir uns für das Style *Cobalt 2.0* (`cobalt203.zip`) mit dem zusätzlichen deutschen Grafikpaket (`lang_german.zip`).

Linux:

```
unzip -o cobalt203.zip -d /opt/lampp/htdocs/phpBB2/templates
unzip -o lang_german.zip -d /opt/lampp/htdocs/phpBB2/language
```

Setzen Sie sicherheitshalber noch einmal die korrekten Benutzerrechte.

```
chown -R nobody.root /opt/lampp/htdocs/phpBB2
```

Windows:

Unzip cobalt203.zip nach <xampp>\htdocs\phpBB2\templates

Unzip lang_german.zip nach <xampp>\htdocs\phpBB2\language

Jetzt muss unser Cobalt-Design erst einmal installiert werden. Und das erledigen Sie im Administrations-Bereich unter STYLES/THEMES und HINZUFÜGEN. Nach erfolgreicher Installation aktivieren Sie dann Cobalt nur noch als Standardstyle. Das geschieht über ALLGEMEINE KONFIGURATION. Zusätzlich müssen Sie den Radio-Button Style überschreiben auf *Ja* setzen. Fertig. Der neue Cobalt-Look gefällt Ihnen nicht? Macht nix! Alles zurück auf das gute *subSilver* setzen Sie dann nach dem eben geschilderten Muster unter ALLGEMEINE KONFIGURATION.



Abbildung 24.36: Das neue Design Cobalt 2.0 unter www.phpbb.de/styles



25 Der BIND-Nameserver

Bereits in den Kapiteln über ProFTPD, FileZilla Server und Apache haben wir das Thema *DNS/Nameserver* gestreift. Nun wollen wir einen genaueren Blick auf das DNS (*Domain Name System*) werfen. Das DNS ist in einem TCP/IP-basierten Netzwerk die Technik, die verantwortlich für die Namensauflösung eines FQDN (*Fully Qualified Domain Name*) in die dazugehörige IP-Adresse ist.

Wenn Sie zum Beispiel in Ihrem Browser <http://www.apachefriends.org> eingeben, dann wird der FQDN www.apachefriends.org intern über eine Anfrage an das DNS in die dazugehörige IP-Adresse 213.239.205.48 aufgelöst. Erst über die IP-Adresse kann dann eine Verbindung zu dem Zielwebserver aufgenommen werden.

Diese Umwandlung passiert auf der Benutzerseite automatisch und der Benutzer bekommt kaum etwas von ihr mit. Vielleicht bemerkt er eine kleine Verzögerung, wenn er die Website seit langem wieder zum ersten Mal aufruft. Das ist dann genau der Moment, in dem der FQDN in eine IP-Adresse aufgelöst wird. Ruft man dieselbe Website gleich nochmal auf, gibt es diese Verzögerung nicht.

Aber auch auf der Betreiberseite einer Website passiert die Einrichtung des DNS in der Regel vollautomatisch. Ihr (Hosting-)Provider kümmert sich darum, dass die IP-Adresse – die er Ihnen zugewiesen hat – automatisch der Domain zugeordnet wird, die Sie bei ihm bestellt haben. Sie müssen sich also um nichts weiter kümmern. Die meisten Provider bieten Ihnen allerdings auch an, bequem über eine Weboberfläche Änderungen an den DNS-Einstellungen zu Ihrer Domain vorzunehmen. In der Regel können Sie hier aber nur bestimmte, eingeschränkte Änderungen vornehmen.

Einige Provider bieten Ihnen auch die Möglichkeit, einen komplett eigenen Nameserver zu betreiben. Hier haben Sie dann die absolute Kontrolle über Ihre Domain und können wirklich alle Details selbst einstellen. Diese absolute Kontrolle bedeutet aber auch, dass Sie Ihre Domain mit Fehleinstellungen absolut unbrauchbar machen können. Allerdings meist nur auf Zeit.

25.1 Warum ein eigener Nameserver?

Gründe, einen eigenen Nameserver betreiben zu wollen, gibt es viele. In der Regel werden aber die Möglichkeiten, die Ihnen Ihr Provider über die Weboberfläche anbietet, wirklich ausreichend sein. Nutzen Sie diesen Service und verschwenden Sie nicht

Ihre Zeit mit dem Betrieb eines eigenen Nameservers, den sie nicht wirklich brauchen. Schließlich muss so ein Nameserver gepflegt werden und birgt seinerseits wieder potenzielle Sicherheitsrisiken.

Letztendlich lässt sich die Notwendigkeit für einen eigenen Nameserver auf drei wesentliche Gründe reduzieren:

1. Die Weboberfläche Ihres Providers bietet nicht die Möglichkeiten, die Sie brauchen. Das wäre der Fall, wenn Sie mehrere Mail-Server für eine Domain verwalten oder weitere Backup-Nameserver einrichten wollten.
2. Sie wollen einfach nur in einem internen Netzwerk mit verschiedenen Virtual Hosts experimentieren und brauchen die Möglichkeit, verschiedene FQDNs zu einer IP-Adresse aufzulösen.
3. Sie wollen für zuhause oder Ihre Firma einen Nameserver-Cache betreiben, der die DNS-Anfragen zwischenspeichert und damit beschleunigt (dazu mehr im Abschnitt 25.4.1, *Caching Name Server*.)

Wenn es Ihnen allerdings nur darum geht, mit Virtual Hosts zu experimentieren, können sie alternativ neben dem DNS auch eine andere Methode zur Namensauflösung verwenden: die Namensauflösung über Dateien.

25.2 DNS für Sparsame: die Hosts-Datei

Bevor das Betriebssystem das DNS um die Namensauflösung bittet, sieht der interne *Resolver* des Betriebssystems zunächst in einer lokalen Datei nach und versucht über diese Datei den FQDN in eine IP-Adresse aufzulösen. Diese Datei ist die sog. Hosts-Datei.

Tabelle 25.1 gibt einen Überblick darüber, wo sich diese Hosts-Datei unter den verschiedenen Betriebssystemen befindet.

Betriebssystem	Datei
Linux	/etc/hosts
Windows NT/2000	C:\winnt\system32\drivers\etc\hosts
Windows XP/Server 2003	C:\windows\system32\drivers\etc\hosts
Windows 98	C:\windows\hosts

Tabelle 25.1: Wo befindet sich die Hosts-Datei?

Sollten Sie mit der Hosts-Datei arbeiten, denken Sie bitte daran, dass einige Programme wie Browser oder FTP-Clients die Auflösungen von FQDN zu IP-Adressen in einem eigenen Cache halten. Wenn Sie die Hosts-Datei verändern, sollten Sie

danach den Browser oder FTP-Client (was immer Sie auch gerade verwenden) schließen und neu starten. Nur so können Sie halbwegs sicher sein, dass auch die IP-Adresse aus der Hosts-Datei genommen wird.

Nachteil dieser Methode: Sie ist nicht netzwerkfähig und funktioniert logischerweise nur auf dem Rechner, auf dem die Hosts-Datei verändert wurde.

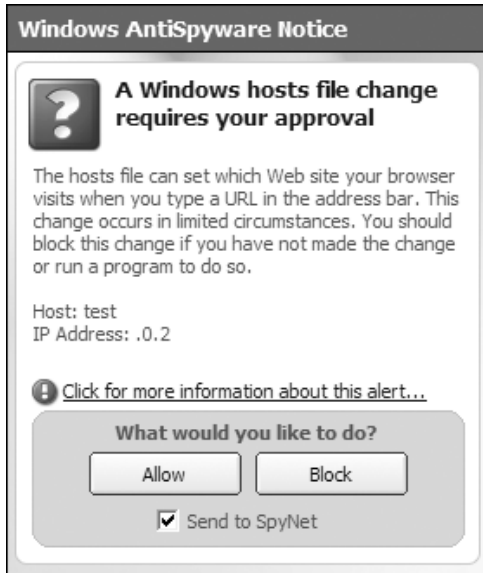


Abbildung 25.1: Wenn Sie ein Tool wie z. B. Microsoft AntiSpyware installiert haben, wird es Sie bei Veränderung der Hosts-Datei informieren und warnen. In diesem Fall können Sie diese Änderung aber getrost zulassen.

25.3 Eine kleine Einführung in das DNS

Wie schon Doctor McCoy vom Raumschiff Enterprise immer wieder sagte: »I'm a doctor, not a ...«, müssen auch wir in diesem Buch immer wieder sagen, dass dies ein Buch über XAMPP ist und nicht über ... Trotzdem wollen wir aber die Bereiche, die XAMPP berührt, nicht total außen vor lassen, und daher hier eine winzigkleine Einführung in das Domain Name System.

Das DNS besteht aus tausenden von so genannten *Name-* oder *DNS-Servern*, die weltweit im Internet verteilt sind. Die Basis dieses weltweiten DNS-Netzwerks bilden die so genannten *Root Server*. Sie stellen eine besondere Art von Nameserver dar, auf die wir hier aber nicht näher eingehen wollen. Jeder der normalen Nameserver verwaltet mindestens eine Domain.

In der Regel wird eine Domain zwei Nameserver haben: den so genannten *Primary Name Server* (oft auch *Master* genannt) und den *Secondary Name Server* (oft auch *Slave* genannt). Der Master ist der Nameserver, der die absolute Hoheit über die Domain-daten hat. Der Slave ist lediglich eine Art Backup-Server. In regelmäßigen Abständen gleicht er sich mit dem Master-Server ab.

Man spricht in diesen beiden Fällen auch von so genannten *Authoritative Name Servers*, also Nameservern, die zu einer bestimmten Domain autorisierte Daten liefern.

Die Daten für eine Domain werden in der so genannten *Zone* gespeichert. Diese Zone kann man sich als abstrakte Datensammlung vorstellen, die alle Daten einer Domain beinhaltet. Gespeichert werden diese Daten dann in dem so genannten *Zone File*, einer Datei, die irgendwo auf der Festplatte eines Servers liegt.

Das Zone File ist wiederum in mehrere *Resource Records* (auch gerne *RR* abgekürzt) unterteilt, die die einzelnen konkreten Daten zu einer Domain beinhalten.

25.3.1 Die Resource Records

Das Wissen um die Resource Records ist mit das Wichtigste, das Sie über das DNS wissen sollten. Je nach Aufgabe oder Bedeutung werden die RRs in verschiedene Typen unterteilt. Insgesamt gibt es etwa 60 solcher RR-Typen, wobei davon allerdings nur 8 Typen im Alltag benutzt werden. Eine komplette Liste aller Typen finden Sie unter <http://www.iana.org/assignments/dns-parameters>.

Ein typischer Resource Record besteht aus 5 Feldern (*name*, *ttl*, *class*, *type* und *data*) und ist folgendermaßen aufgebaut:

```
<name>           [<ttl>] [<class>] <type> <data>
```

Die Felder *ttl* und *class* sind optional und daher hier in eckige Klammern gestellt.

Am besten lassen sich diese Felder anhand von Beispielen erklären. Im Folgenden werden Sie die fünf wichtigsten RR-Typen, beispielhaft an der *apachefriends.org*-Domain, kennen lernen.

Hinweis



Die Benennung der Resource Records wird in der Regel nicht konsequent durchgehalten. So benennt man beispielsweise einen Resource Record vom Typ NS mit den folgenden synonym gebräuchlichen Begriffen: *NS resource record*, *NS record* oder *NS-RR*.

A

Der *A Record* (oder *Address Record*) ist mit Sicherheit der gebräuchlichste RR überhaupt. Mit ihm wird einem FQDN eine IP-Adresse zugewiesen.

Zum Beispiel:

```
www.apachefriends.org.      IN      A      213.239.205.48
```

Damit wird dem FQDN `www.apachefriends.org` die IP-Adresse `213.239.205.48` zugewiesen. Beachten Sie in diesem Fall den Punkt hinter dem `org`.

Da der Resource Record immer im Zone File einer Domain auftaucht, kann der Domainname auch weggelassen werden:

```
www                        IN      A      213.239.205.48
```

In diesem Fall wird intern das `www` um `apachefriends.org` erweitert. Beachten Sie, dass in diesem Fall der Punkt am Ende des Namens weggelassen wird.

MX

Jede Domain hat in der Regel mindestens einen *MX Record*. Der MX Record bestimmt den verantwortlichen Mailserver (*Mail eXchanger*) für eine Domain, also den Server im Internet, der die Mail für diese Domain empfangen soll.

Zum Beispiel:

```
apachefriends.org.      IN      MX      10 mail.apachefriends.org.
```

Mit diesem Eintrag wird der Domain `apachefriends.org` der Mailserver `mail.apachefriends.org` zugewiesen. Die 10 vor dem Mailserver ist der so genannte Preference Value. Mit ihm wird eine Mailserver-Präferenz angegeben. Je kleiner der Wert, desto höher die Präferenz für diesen Mailserver. Ein Wert, der natürlich erst dann Bedeutung bekommt, wenn es mehrere MX Records für dieselbe Domain gibt.

Oft wird man auch folgende abkürzende Schreibweise finden:

```
IN      MX      10 mail.apachefriends.org.
```

Wird das Name-Feld leer gelassen, dann wird das Name-Feld von dem vorherigen Resource Record verwendet. Da die MX Records in der Regel am Anfang eines Zone Files definiert werden, ist der Name des vorherigen Records der Domainname selbst. In diesem Beispiel eben `apachefriends.org`. Siehe dazu auch den *SOA Record* auf der folgenden Seite.

CNAME

Der *CNAME Record*, auch *Alias Record* genannt, ist eng verwandt mit dem A Record. Mit ihm können Sie einem FQDN über einen anderen FQDN indirekt eine IP-Adresse zuweisen.

Zum Beispiel:

```
irc                IN      CNAME    irc.kwm-web.info.
```

Der Adresse `irc.apachefriends.org` wird in diesem Fall keine IP-Adresse, sondern ein anderer FQDN – hier `irc.kwm-web.info` – zugewiesen. D. h., die eigentliche IP-Adresse wird nicht im Zone File der `apachefriends.org`-Domain bestimmt, sondern im Zone File der `kwm-web.info`-Domain.

NS

Über den *NS Record* bestimmen Sie, welche Nameserver für diese Domain zuständig sind. In der Regel wird eine Domain zwei Nameserver haben: den so genannten *Primary Name Server* und den *Secondary Name Server*.

Zum Beispiel:

```
apachefriends.org.  IN      NS      ns.apachefriends.org.
apachefriends.org.  IN      NS      ns2.apachefriends.org.
```

Ähnlich wie bei den MX Records stehen die NS Records am Anfang des Zone Files und daher wird hier oft das Name-Feld leer gelassen. In der Praxis wird man folglich eher die folgende abgekürzte Form finden:

```
      IN      NS      ns.apachefriends.org.
      IN      NS      ns2.apachefriends.org.
```

Zu beachten ist hier auch, dass zwar Primary und Secondary Name Server aufgeführt werden. Aber welcher von den beiden der Primary ist, wird nicht hier, sondern woanders eingestellt.

SOA

Der komplizierteste aller Resource Records ist der so genannte *SOA Record*. SOA steht für *Start of Authority* und definiert eine Reihe von Metainformationen einer Domain.

Zum Beispiel:

```
@                IN SOA ns.apachefriends.org. oswald.apachefriends.org. (
                        2005120701
                        10800
                        3600
                        604800
                        86400 )
```

Hier gibt es gleich zwei Besonderheiten:

1. Das @ am Anfang der Zeile ist eine Abkürzung für den Domainnamen dieses Zone Files. In diesem Fall kann man das @ also so lesen, also ob dort `apache-friends.org.` (inkl. Punkt) stehen würde.

2. Der Resource Record geht über mehrere Zeilen. Das ist durch die runden Klammern möglich. Lässt man die runden Klammern weg, dann muss der ganze SOA Record – wie sonst üblich – in einer Zeile stehen.

Die einzelnen Felder des SOA Records:

Das erste Feld (`ns.apachefriends.org.`) ist der Name des Primary Name Servers dieser Domain. Das zweite Feld (`oswald.apachefriends.org.`) ist die E-Mail-Adresse des für dieses Zone File Verantwortlichen, wobei das @ in der E-Mail-Adresse durch einen Punkt ersetzt wird.

In den Klammern stehen die folgenden Werte:

■ `serial`

Die Seriennummer oder Versionsnummer des Zone Files. Bei jeder Änderung an dem Zone File muss dieser Wert hoch gezählt werden. Üblicherweise kodiert man das Datum im Format `JJJJMMTT`, gefolgt von einem zweistelligen Zähler, in diesen Wert. Sollte man mehrere Änderungen an einem Tag vornehmen, würde man den Zähler jeweils hoch zählen.

Das Datum ist nicht technisch vorgeschrieben, es hat lediglich den Vorteil, dass man selbst auch noch überblicken kann, wann das Zone File das letzte Mal geändert wurde. Was auch immer man hier verwendet, wichtig allein ist nur, dass die Zahl bei jeder Änderung größer wird.

■ `refresh`

Dieser Wert (in Sekunden) gibt an, wie oft ein Secondary Name Server bei dem Primary Name Server nachfragen soll, ob eine neue Version des Zone Files vorliegt.

Ein typischer und guter Wert ist hier 10800 Sekunden (3 Stunden).

■ `retry`

Wenn der Secondary Name Server den Primary Name Server nicht erreichen kann, dann versucht er es nach *retry* Sekunden noch mal.

Ein typischer Wert für *retry* ist 3600 Sekunden (also eine Stunde).

■ `expire`

Dieser Wert (ebenfalls in Sekunden) gibt die Zeit an, nach der ein Secondary Name Server alle seine Daten zu einer Domain wegwirft, wenn er den Primary Name Server nicht erreichen kann. Ein typischer Wert ist hier 604800 Sekunden, also 1 Woche.

D. h. wenn der Secondary Name Server den Primary Name Server eine Woche lang nicht erreicht hat, dann wirft der Secondary sein Zone File weg und gibt auch keine Daten mehr für diese Domain heraus.

■ TTL

Der *Time to Live*-Wert gibt an, wie lange ein anderer Nameserver die Resource Records dieses Zone Files cachen soll. Je kürzer man diesen Zeitraum setzt, desto häufiger wird der eigene Nameserver gefragt.

Ein typischer und guter Wert ist hier 86400 Sekunden (1 Tag).

25.3.2 Ein komplettes Zone File

Hier noch einmal das komplette Zone File für die `apachefriends.org`-Domain im Überblick:

```
@           IN SOA ns.apachefriends.org. oswald.apachefriends.org. (
                                2005120701
                                10800
                                3600
                                604800
                                86400 )

           IN      NS      ns.apachefriends.org.
           IN      NS      ns2.apachefriends.org.
           IN      MX      10 mail.apachefriends.org.
           IN      A       213.239.205.48

ns         IN      A       213.239.205.48
ns2        IN      A       82.165.29.118
www        IN      A       213.239.205.48
mail       IN      A       213.239.205.48
irc        IN      CNAME   irc.kwm-web.info.
```

Hier sieht man deutlich, wie das Name-Feld am Anfang des Zone Files mit `@` auf den Wert `apachefriends.org.` gesetzt wird. Für die folgenden zwei NS Records, den MX Record und den ersten A Record gilt dann auch dieser Wert für das Name-Feld.

Dieser A Record ist dann auch dafür verantwortlich, dass die Website von Apache Friends nicht nur unter `www.apachefriends.org`, sondern auch unter `apachefriends.org` erreichbar ist.

25.4 Installation

Software für DNS-Server gibt es eine ganze Menge. Die wohl bekannteste und gebräuchlichste dürfte dabei BIND (*Berkeley Internet Name Domain*) sein. Neben der großen Verbreitung hat der BIND noch einen weiteren Vorteil: Es gibt ihn sowohl für Linux als auch für Windows-Systeme. D. h., Sie können ihn einerseits wunderbar zu Hause testen und ausprobieren und ihn andererseits auf Ihrem Produktions-Nameserver einsetzen, um Ihre Internetdomain zu verwalten.

25.4.1 Linux

Die Installation von BIND ist unter Linux eigentlich gar nicht der Rede wert. Er gehört zum Standardumfang jeder Linux-Distribution, sei es SUSE, RedHat/Fedora, Mandrake oder Debian, und kann mit dem jeweiligen Paketmanager bequem installiert werden.

Installieren Sie am besten auch gleich noch das Paket *bind-utils* (im Falle von Red-Hat/Fedora/openSUSE) oder *dnsutils* (im Falle von Debian oder Ubuntu). Dieses Paket enthält nützliche Tools, um auf einen DNS-Server zuzugreifen. Ohne diese Tools werden Sie kaum eine Chance bei einer eventuellen Fehlersuche haben.

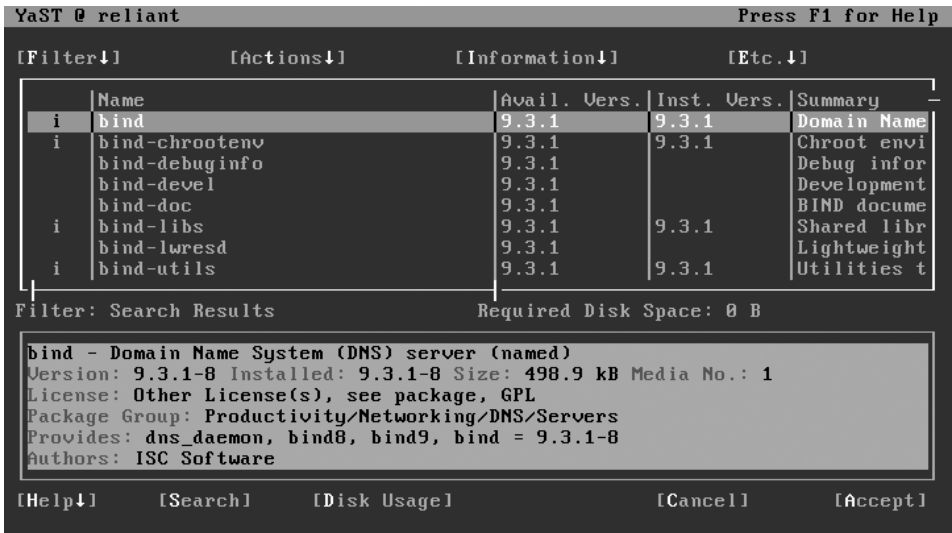


Abbildung 25.2: Die Installation von BIND unter openSUSE 10.0

Nach der Installation starten Sie den BIND. Entnehmen Sie dazu den Befehl der folgenden Tabelle:

Betriebssystem	Befehl
openSUSE	rcnamed start
RedHat/Fedora	/etc/init.d/named start
Ubuntu/Debian	/etc/init.d/bind start

Tabelle 25.2: Befehl zum Starten von BIND unter Linux

Ggf. muss noch dafür gesorgt werden, dass BIND auch beim Booten des Systems mit gestartet wird.

25.4.2 Windows

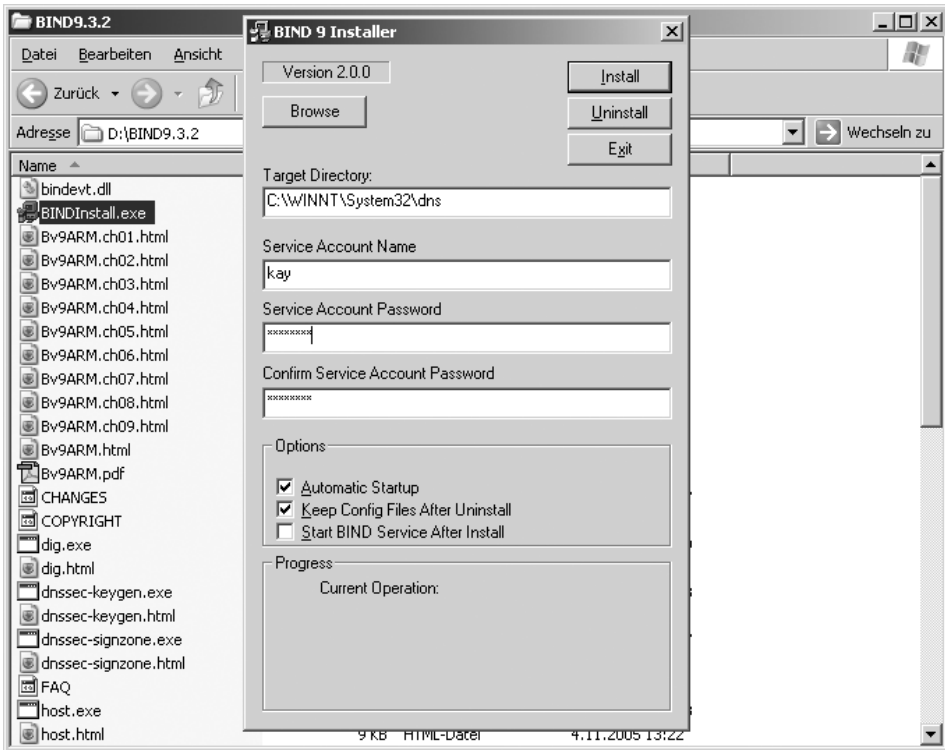


Abbildung 25.3: BIND entpacken und über BINDInstall installieren

Unter Windows NT benötigen Sie das aktuelle ZIP-Paket von der Webseite <http://www.isc.org/sw/bind/>. Extrahieren Sie das ZIP (in diesem Beispiel BIND9.3.2.zip) und starten Sie danach die BINDInstall.exe. Normalerweise bietet Ihnen nun BIND den Pfad C:\WINDOWS\system32\dns als Zielverzeichnis für eine Installation an. Sie sollten bei diesem Standard bleiben. Wichtig für eine Serviceinstallation ist, dass dieser das Passwort des Kontos erhält, mit dem der Dienst gestartet werden soll. Für Testzwecke reicht das Kennwort Ihres administrativen Kontos. Im produktiven Einsatz sollten Sie sich überlegen, dem Bind-Server ein eigenes Konto zuzuweisen.

Nach der Installation ist es ratsam, die Einstellungen des neuen Dienstes zu kontrollieren. Sie öffnen hierfür SYSTEMSTEUERUNG > VERWALTUNG > DIENSTE und suchen den Dienst ISC BIND heraus. Kontrollieren Sie den Starttyp von Bind und natürlich das Benutzerkonto auf der Karteikarte ANMELDEN.

Sie starten den Dienst über:

```
net start "ISC BIND"
```


Sie beenden den Dienst über:

```
net stop "ISC BIND"
```

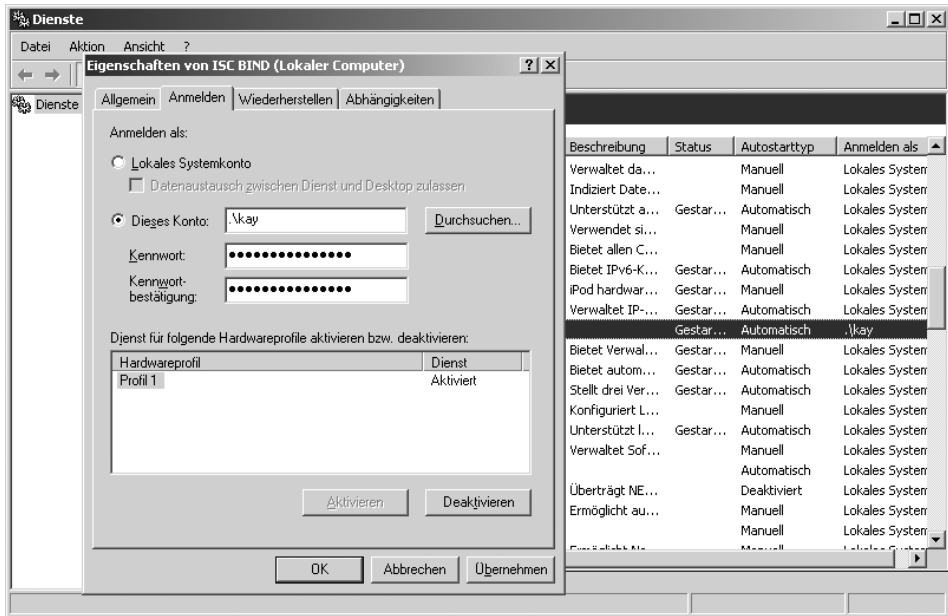


Abbildung 25.4: Die Dienste-Eigenschaften von ISC BIND überprüfen

Sie finden nach der Installation die folgenden Verzeichnisse vor:

```
C:\WINDOWS\system32\dns\bin
C:\WINDOWS\system32\dns\etc
```

Innerhalb des bin-Unterverzeichnisses finden Sie alle BIND-Server-spezifischen Programme sowie die sog. *DNS Utilities* vor. In dem noch leeren Unterverzeichnis etc hinterlegen Sie die zukünftigen Konfigurationsdateien. Linux-Benutzer haben es dabei einfacher, da in den entsprechenden Bind-Paketen meist eine funktionierende Einstellung vorhanden ist. Das ist unter Windows nicht so, so dass wir Ihnen eine kleine Beispielkonfiguration vorstellen.

Die zentrale Konfigurationsdatei named.conf

Erstellen Sie nun die zentrale Konfigurationsdatei C:\WINDOWS\system32\dns\etc\named.conf mit dem folgenden Inhalt:

```
options {
    directory "C:\WINDOWS\system32\dns\etc";
};
```

25.5 Konfigurationsbeispiele

Im nächsten Schritt sollten Sie Ihren Nameserver zum Funktionieren bringen. Am einfachsten wird das sein, wenn Sie ihn zunächst als *Caching Name Server* konfigurieren.

Betriebssystem	Datei
SUSE	/etc/named.conf
RedHat/Fedora	/etc/named.conf
Ubuntu/Debian	/etc/bind/named.conf.options /etc/bind/named.conf.local
Windows XP	C:\WINDOWS\system32\dns\etc\named.conf

Tabelle 25.3: Speicherort der BIND-Konfigurationsdatei unter den verschiedenen Linux-Distributionen und Windows

Hinweis



In den folgenden Beispielen wird immer der Linux-Pfad `/var/named/` verwendet. Im Falle von Windows ersetzen Sie diesen Pfad einfach durch `C:\WINDOWS\system32\dns\etc`.



Abbildung 25.5: Der etc-Ordner mit allen Konfigurationsdateien unter Windows XP

25.5.1 Caching Name Server

Der Caching Name Server dürfte die einfachste Konfiguration eines Nameservers sein. Dieser Nameserver verwaltet keine eigene Domain und beantwortet nur DNS-Anfragen.

Setzen Sie dazu einfach die folgenden beiden Optionen in der BIND-Konfigurationsdatei:

```
options {  
    ...  
    allow-query { 192.168.0.0/24; };  
    recursion yes;  
    ...  
};
```

Mit dem *allow-query* schränken Sie den Zugriff auf Ihren Nameserver ein. Nur Rechner, deren IP-Adresse mit 192.168.0 beginnt, dürfen auf den Nameserver zugreifen. Diese Einstellung ist wichtig, wenn Ihr Nameserver im Internet oder einem anderen größeren Netzwerk erreichbar ist. Haben Sie den Nameserver lediglich zu Hause laufen, können Sie diese Einschränkung auch sein lassen.

Die Einstellung *recursion yes* ist in diesem Fall viel wichtiger. Mit ihr stellen Sie ein, dass der Nameserver auch rekursive Anfragen durchführen soll. Das heißt eigentlich nichts anderes, als dass er versucht, die Frage nach der IP-Adresse eines FQDN auch tatsächlich zu beantworten.

25.5.2 Caching und Primary Name Server

Dies ist eine Mischung aus der letzten und der nächsten Beispielkonfiguration. Der Nameserver ist sowohl ein Caching Name Server als auch ein Primary Name Server für eine Domain.

Diese Konfiguration würden Sie dann verwenden, wenn der Nameserver auch für Sie die DNS-Anfragen Ihrer Rechner beantworten soll. Gerade zum Testen und Ausprobieren ist diese Konfiguration wunderbar geeignet.

In der BIND-Konfigurationsdatei tragen Sie dazu folgende Einstellungen ein:

```
options {  
    ...  
    allow-query { any; };  
    recursion yes;  
    allow-recursion { 192.168.0.0/24; };  
    ...  
};
```

Mit der ersten Zeile erlauben Sie nun potentiell jedem, dass er auf Ihren Nameserver zugreifen kann. Die Aufgabe eines Primary Name Servers ist es ja eben, die Daten einer Domain öffentlich zu verwalten. Also muss man auch jeden an diese Daten lassen.

Mit *recursion yes* weisen Sie wieder den Nameserver an, auch rekursive Anfragen durchzuführen. Allerdings, und das definiert *allow-recursion*, sollen die rekursiven

Anfragen nur dann beantwortet werden, wenn die IP-Adresse des anfragenden Rechners mit 192.168.0 beginnt.

Etwas weiter unten in der BIND-Konfigurationsdatei tragen Sie die folgenden Zeilen ein (Achtung: Bei Debian und Ubuntu verwenden Sie in diesem Schritt bitte die Datei `/etc/bind/named.conf.local`):

```
zone "xampp-buch.de" IN {
    type master;
    file "/var/named/db.xampp-buch.de";
};
```

Mit diesen Zeilen machen Sie Ihren Nameserver zum »Master« (also Primary Name Server) für die Domain `xampp-buch.de` und sagen ihm, dass er das Zone File in der Datei `/var/named/db.xampp-buch.de` findet.

25.5.3 Primary Name Server

Um einen Primary Name Server zu konfigurieren, muss man nur die Caching-Funktion aus der letzten Konfiguration entfernen. Das macht man am besten indirekt, indem man mit *recursion no* rekursive Anfragen an den Nameserver verbietet:

```
options {
    ...
    allow-query { any; };
    recursion no;
    ...
};
```

Ansonsten ändert sich eigentlich nichts gegenüber der letzten Konfiguration. Die Konfiguration `allow-query { any; };` bleibt bestehen, da wir ja wollen, dass grundsätzlich jeder auf unseren Nameserver zugreifen können soll.

Auch die Definition der `xampp-buch.de`-Domain bleibt in diesem Fall gleich:

```
zone "xampp-buch.de" IN {
    type master;
    file "/var/named/db.xampp-buch.de";
}
```

25.5.4 Zone File

Nun müssen Sie noch das Zone File erstellen. Unter Windows verwenden Sie dazu das Verzeichnis `C:\WINDOWS\system32\dns\etc` und unter Linux das Verzeichnis `/var/named`. Aber Vorsicht: Nicht alle Linux-Distributionen verwenden `/var/named` zum Speichern der Zone Files. Wählen Sie ggf. einen anderen Pfad oder legen Sie dieses Verzeichnis einfach an.

In diese Datei schreiben Sie Folgendes:

Listing 25.1: Ein typisches Zone File

```
@          IN      SOA  ns.xampp-buch.de. oswald.xampp.org. (
                                2006013001
                                10800
                                3600
                                604800
                                86400 )

          IN      NS   ns.xampp-buch.de.

          IN      MX   10 mail.xampp-buch.de.

www       IN      A    127.0.0.1
test      IN      A    127.0.0.1
```

25.5.5 BIND starten

Nachdem Sie die Konfiguration erstellt haben, sollten Sie nun den BIND starten, um den ersten Test durchzuführen.

Unter Linux starten Sie den BIND mit dem folgenden Befehl:

```
/etc/init.d/named start
```

oder im Falle von Debian/Ubuntu:

```
/etc/init.d/bind start
```

Bei BIND (unter Windows) als Dienstinstallation erfolgt der Start in der Regel automatisch.

Start des Dienstes: `net start "ISC BIND"`

Beenden des Dienstes: `net stop "ISC BIND"`

Tipp



Fehlersuche bei Startproblemen

Wenn der BIND nicht starten mag, dann wird mit höchster Wahrscheinlichkeit irgendein Tippfehler dafür verantwortlich sein. Werfen Sie einfach einen Blick an das Ende der Datei `/var/log/messages` (oder bei Debian/Ubuntu `/var/log/syslog`). Dort werden die Fehlermeldungen von BIND protokolliert.

Korrigieren Sie ggf. Fehler in der Konfigurationsdatei oder im Zone File und starten Sie den BIND mit folgendem Befehl neu:

```
/etc/init.d/named restart
```

Oder im Falle von Debian/Ubuntu:

```
/etc/init.d/bind restart
```

25.5.6 Eigenen Nameserver verwenden

Linux

Zunächst zum Testen und langfristig, wenn Sie wirklich Ihren eigenen Nameserver verwenden wollen, stellen Sie in der Datei `/etc/resolv.conf` die IP-Adresse Ihres Nameservers ein.

Ihre `resolv.conf` könnte etwa wie folgt aussehen:

```
domain apachefriends.org
nameserver 213.133.98.98
nameserver 213.133.99.99
```

Kommentieren Sie die `nameserver`-Zeilen aus (zur Sicherheit, falls Sie sie später wieder brauchen) und fügen Sie eine Zeile mit dem Inhalt `nameserver 127.0.0.1` hinzu.

Etwa so:

```
domain apachefriends.org
#nameserver 213.133.98.98
#nameserver 213.133.99.99
nameserver 127.0.0.1
```

Damit weisen Sie Ihr System an, dass es nun nicht mehr die Nameserver Ihres Providers verwenden soll, sondern den eigenen Nameserver.

Windows

Um Ihren oder einen anderen DNS-Server für die IP-Auflösung zu benutzen, öffnen Sie die EIGENSCHAFTEN Ihrer AKTIVEN LAN-VERBINDUNG unter START > EINSTELLUNGEN > NETZWERKVERBINDUNGEN. Wählen Sie im zweiten Schritt die EIGENSCHAFTEN des Internetprotokolls TCP/IP aus. In diesem Eigenschaften-Kontext können Sie wahlweise die Serveradresse des DNS-Servers z.B. über einen DHCP-Server dynamisch zuordnen lassen. Oder Sie legen wie in diesem Beispiel eine Adresse für den Nameserver über die Option FOLGENDE DNS-SERVERADRESSEN VERWENDEN fest. Wenn der DNS-Server wie BIND auf demselben Rechner läuft, kann die Loopback-Adresse 127.0.0.1 eingegeben werden. Ansonsten benutzen Sie die IP-Adresse des jeweiligen Rechners. Im lokalen Netz wäre das in der Regel eine Adressierung aus dem A-, B- oder C-Netzwerkbereich.



Achtung

Nachdem Sie diese Einstellung vorgenommen haben, werden Sie unter Umständen keine Internetverbindungen mehr von Ihrem System aus öffnen können. Das liegt daran, dass Sie nun Ihren eigenen Nameserver verwenden und dieser evtl. nicht richtig funktionieren wird. D. h., Ihr System ist nun nicht mehr in der Lage, einen FQDN in eine IP-Adresse umzuwandeln. Erst wenn Ihr System wieder IP-Adressen zu einem FQDN herausfinden kann, können Sie wieder normal aufs Internet zugreifen.

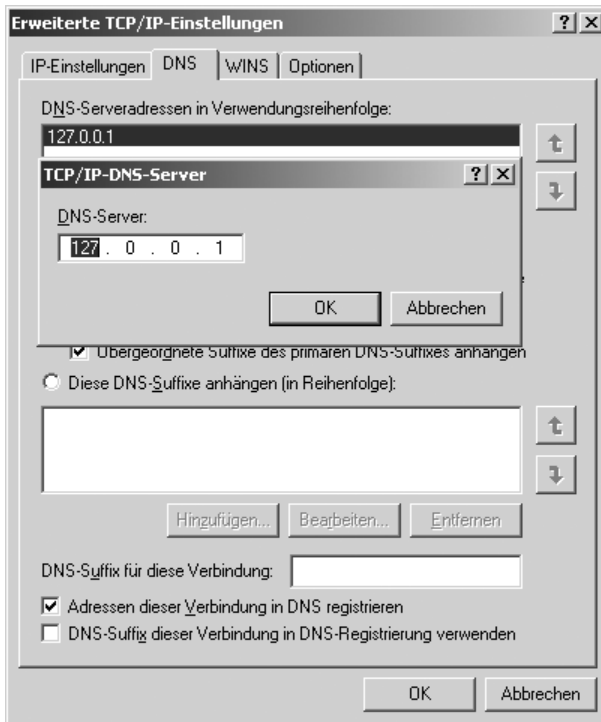


Abbildung 25.6: Verwenden Sie nun Ihren eigenen Nameserver

25.5.7 Testen

Wenn Sie sich an das Zone File-Beispiel aus diesem Buch gehalten haben, dann können Sie Ihren Nameserver nun testen, indem Sie die URL <http://test.xampp-buch.de> mal in Ihrem Browser aufrufen. Sie sollten nun die Seiten Ihres eigenen Webservers sehen – wenn dieser unter 127.0.0.1 erreichbar ist und wenn er gestartet ist.

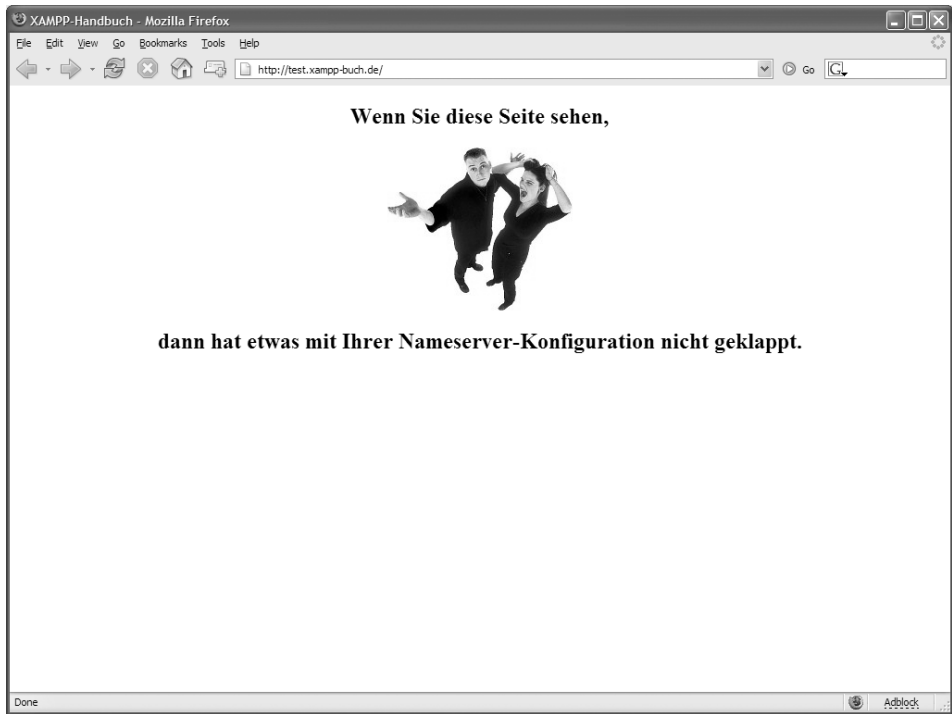


Abbildung 25.7: Wenn Ihre Nameserver-Konfiguration nicht geklappt hat, dann sehen Sie vielleicht diese Fehlerseite.

Wenn Sie allerdings die Seite in Abbildung 25.7 sehen, dann hat etwas mit Ihrer Konfiguration nicht geklappt. Was Sie dort nämlich sehen, ist eine Seite, die wir extra für dieses Kapitel ins Internet gestellt haben.



26 Eigenes XAMPP für Windows

Der XAMPP beruht in nunmehr klassischer Weise auf zwei Paketgruppen. Zu der ersten Gruppe gehören die komprimierten *ZIP- bzw. 7-zip-Pakete* ohne Installer. Die zweite Gruppe mit den *Installer-Versionen* beruht auf Übersetzungen mit dem freien *Nullsoft Scriptable Install System* (NSIS). Das jeweils aktuelle NSIS-Skript finden Sie in der Regel im Verzeichnis:

```
<xampp>\install\nsi
```

Die beiden nun folgenden Beispiele erhalten Sie auch als ZIP-Dateien unter der URL:
<http://www.apachefriends.org/files/misc/nsi/>

26.1 Der XAMPP NSIS Installer mit eigenem Programm

Ein paar grundlegende Sätze vorweg – als kleine Warnung vor einen unüberlegten Einsatz: Ein *Installer* packt nicht nur fertige binäre Systeme auf dem Ursprungssystem ein und auf einem Zielsystem aus. Ein Installer-System enthält auch viele Fähigkeiten der Systemanwendung und -manipulation. Wenn sinnvoll genutzt, ist eine solche Anwendung ein starkes Instrument für eine einfache Veröffentlichung eines Programms. Mutwillig verfälscht oder fahrlässig behandelt, kann ein solcher Installer durchaus zerstörerisch auf das Zielsystem wirken.

Das Ziel diese Kapitels: Sie sollen zum einen den XAMPP mit dem NSIS als Installer anfertigen können und zum anderen hier Ihre eigene Applikation hinzufügen. Außerdem sollen Sie den fertigen Installer zum Abschluss auf einer CD-ROM inklusive Autostart anbieten können.

Als zusätzliches Programm verwenden wir in diesem Beispiel *WordPress* (<http://wordpress.de>), ein *Weblog-System* mit *MySQL*-Anbindung. Sie können das Beispiel für Ihre konkreten Bedürfnisse quasi »geistig übersetzen«.

Aber bleiben wir erst einmal konkret. Installieren Sie zu Beginn das Nullsoft Scriptable Install System von der Homepage <http://nsis.sourceforge.net>. Wir benutzen hier die *NSIS-Version 2.14*. Darüber hinaus setzen wir den NSIS-Editor *HM NIS Edit* (<http://hmne.sourceforge.net>) in der Version 2.0.3 ein. HM NIS Edit stellt eine perfekte Komplettierung des ansonsten etwas schmucklosen NSIS-Installers dar.

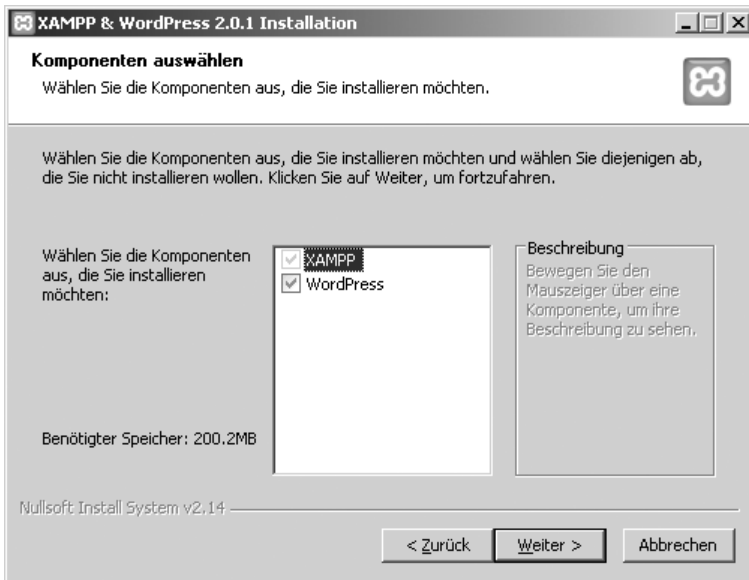


Abbildung 26.1: Der Nullsoft Installer mit Komponentenauswahl

Unser binäres Beispielprogramm für das NSIS besteht aus zwei Komponenten: dem XAMPP (Basispaket) und dem WordPress-Paket, dem PHP-Programm mit eigener MySQL-Datenbank.

Basisprogramm: `xampp-win32-1.5.1.exe` (7-zip)

Beispielprogramm: `wordpress-2.0.1.zip`

26.1.1 Vorbereitungen

Extrahieren Sie den XAMPP direkt auf eine Partition. In diesem Beispiel ist es Laufwerk D:\. Starten Sie in **keinem Fall** die `setup-xampp.bat`. In der obersten Verzeichnishierarchie können Sie nun den Apache sowie den XAMPP per Handstart (beispielsweise über die `xampp_start.exe`) starten. Sollte noch ein weiterer XAMPP auf Ihrem System laufen, beenden Sie vorher den dortigen Apache und MySQL.

Extrahieren Sie nun die `wordpress-x.y.z.zip` direkt in den `<xampp>\htdocs`-Ordner, damit beispielsweise `D:\xampp\htdocs\wordpress` entsteht. Dann legen Sie für WordPress in MySQL einen neuen Benutzer für die neue Datenbank an. Optional vergeben Sie auch ein Passwort für den MySQL-Administrator `root`.

```
D:\xampp\mysql\bin\mysql -h localhost -u root -p
mysql>CREATE DATABASE wordpress;
mysql>GRANT ALL PRIVILEGES ON wordpress.* TO 'blogadmin'@'localhost' IDENTIFIED BY
'987z654' WITH GRANT OPTION;
mysql>use mysql;
```

```
mysql>UPDATE user SET Password=PASSWORD('xampp45rt') WHERE user='root';
mysql>FLUSH PRIVILEGES;
mysql>quit
```

Sollten Sie sich für die Vergabe eines Passworts für den *root*-Administrator entscheiden, ändern Sie bitte auch die `config.inc.php` für *phpMyAdmin*. Ansonsten kann der zukünftige Benutzer Ihres Pakets nicht *phpMyAdmin* nutzen.

```
$cfg['Servers'][$i]['auth_type']    = 'config';
$cfg['Servers'][$i]['user']        = 'root';
$cfg['Servers'][$i]['password']    = 'xampp45rt';
```

Kommen wir nun aber wieder zurück zur Einrichtung von *WordPress*. Öffnen Sie hierfür die `<xampp>\htdocs\wordpress\wp-config-sample.php`, passen Sie die Konfiguration an und speichern Sie diese als `wp-config.php`. Beispiel:

Konfigurationsdatei: `<xampp>\htdocs\wordpress\wp-config.php`

```
<?php
define('DB_NAME', 'wordpress'); // The name of the database
define('DB_USER', 'blogadmin'); // Your MySQL username
define('DB_PASSWORD', '987z654'); // ...and password
define('DB_HOST', 'localhost'); // Host
$table_prefix = 'wp_'
define('WPLANG', 'de');
define('ABSPATH', dirname(__FILE__).'/');
require_once(ABSPATH.'wp-settings.php');
?>
```

Im nächsten Schritt initialisieren Sie die Tabellen für WordPress. Öffnen Sie hierfür die URL <http://localhost/wordpress/wp-admin/install.php> und vollziehen Sie die dort vorgegebenen Schritte. Danach können Sie sich mit dem bereits vorhandenen administrativen Standardbenutzer *admin* in das WordPress-System einloggen.

WordPress-Login: <http://localhost/wordpress/wp-login.php>

Benutzername: *admin*

Passwort: *f30cee*

Wenn Sie die weiterführende Einrichtung von WordPress zu Ihrer Zufriedenheit abgeschlossen haben, beenden Sie den Apache und MySQL im zukünftigen Installer-XAMPP. Die Pointe unseres Installers ist, dass wir hier WordPress als Option im Installer anbieten wollen. Der Benutzer kann also bei der Installation auswählen, ob mit XAMPP auch WordPress eingerichtet werden soll. Hierfür muss nun aber der gesamte WordPress-Bereich wieder ausgelagert werden. Für dieses Beispiel haben wir einen neuen Ordner mit dem Namen `nsi-wordpress` mit der folgenden Struktur eingerichtet:



Abbildung 26.2: Der Admin-Bereich von WordPress

```
D:\nsi-wordpress\nsi
D:\nsi-wordpress\xampp
D:\nsi-wordpress\xampp\htdocs\wordpress
D:\nsi-wordpress\xampp\install
D:\nsi-wordpress\xampp\mysql\data\wordpress
```

Das neue Verzeichnis besteht aus einem `nsi`- und einem `xampp`-Unterordner. Zu dem `nsi`-Bereich kommen wir gleich. Beginnen wir mit dem XAMPP-Ordner. Verschieben Sie das komplette WordPress-Verzeichnis von `<xampp>\htdocs` nach `D:\nsi-wordpress\xampp\htdocs\`. Verschieben Sie ebenfalls die WordPress-Datenbank von `<xampp>\mysql\data` nach `D:\nsi-wordpress\xampp\mysql\data`. Dem XAMPP fehlt damit der vollständige WordPress-Bereich. Und das ist auch gut so, möchten wir das Programm optional anbieten.

Zum Abschluss der Vorbereitungen wenden Sie sich den *Icons* zu. Das Icon mit der Endung *.ico* darf für das Nullsoft Install System die Größen 8×8 , 16×16 sowie 32×32 Pixel besitzen. Wir unterscheiden hier zwei Kategorien von Icons: zum einen interne Icons für den Installer und Uninstaller, (diese werden mit der Übersetzung des Installers automatisch implementiert), zum anderen Icons für z.B. eine Illustration der Programmeinträge unter `START > PROGRAMME`. Sie müssen auf dem zukünftigen Zielsystem vorgehalten werden.

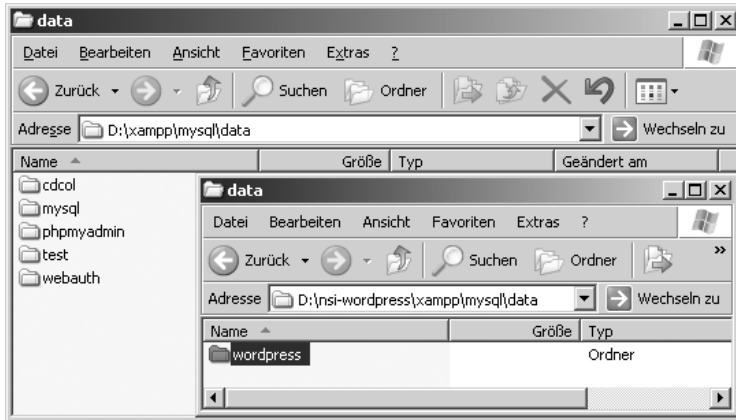


Abbildung 26.3: Auch die Datenbank wordpress wird verschoben

Als Icon-Ordner benutzen wir in diesem Beispiel das `<xampp>\install`-Verzeichnis:

```
D:\nsi-wordpress\xampp\install
[D:\nsi-wordpress\xampp\install\wordpress.ico]
[D:\nsi-wordpress\xampp\install\xampp-icon.ico]
[D:\nsi-wordpress\xampp\install\xampp-icon-uninstall.ico]
```

Die einzelnen hier verwendeten Icons erhalten Sie auch auf unseren Apache Friends-Seiten:

<http://www.apachefriends.org/files/misc/icons/>

Ein freier Bitmap-zu-Icon-Konverter und -Editor ist beispielsweise *@icon sushi* (<http://www.towofu.net/soft/e-index.php>).

26.1.2 Der Nullsoft Installer

Das Herzstück in diesem Kapitel ist sicherlich das eigentliche Installer-Skript. Die Nullsoft-Skripts haben standardgemäß die Endung `.nsi`. Sie legen allgemein die eigentlichen Attribute der Installation fest. Um das Beispiel auszutesten, benötigen Sie wie bereits erwähnt das Nullsoft Scriptable Install System. Vorweg noch ein kleiner Hinweis: Als Kommentarzeichen wird eine Raute (#) oder ein Semikolon (;) verwendet.

NSI-Quelldatei: `\nsi-wordpress\nsi\xamppwordpress.nsi`

Listing 26.1: NSIS-Skript für den XAMPP-Installer

```
SetCompressor /solid lzma # Starke Kompression mit LZMA
XPStyle on
# Produktinformationen
!define PRODUCT_NAME "XAMPP + WordPress" # Produktinformationen
!define PRODUCT_VERSION "2.0.1"
```

```

!define PRODUCT_PUBLISHER "Kay Vogelgesang, Kai 'Oswald' Seidler"
!define PRODUCT_WEB_SITE "http://www.apachefriends.org"
# Eingebettete Icons
!define MUI_ICON "D:\nsi-wordpress\xampp\install\xampp-icon.ico"
!define MUI_UNICON "D:\nsi-wordpress\xampp\install\xampp-icon-uninstall.ico"
# Produktinformationen plus Icons plus File-Name
Name "${PRODUCT_NAME} ${PRODUCT_VERSION}"
Icon "D:\nsi-wordpress\xampp\install\xampp-icon.ico"
OutFile "D:\nsi-wordpress\xampp\WordPress-installer.exe"
# Verzeichnis Programme wird vorgeschlagen
InstallDir "$PROGRAMFILES"
# Transparente Installation
ShowInstDetails show
ShowUninstDetails show
# Wir benutzen hier das Modern User Interface
!include "MUI.nsh"
# Verschiedene Macros z.B. für die Auswahl von Komponenten
!insertmacro MUI_PAGE_LICENSE "D:\nsi-wordpress\xampp\htdocs\wordpress\license.txt"
!insertmacro MUI_PAGE_COMPONENTS
!insertmacro MUI_PAGE_DIRECTORY
!insertmacro MUI_PAGE_INSTFILES
!insertmacro MUI_UNPAGE_CONFIRM
!insertmacro MUI_UNPAGE_INSTFILES
!insertmacro MUI_LANGUAGE "German" # Erste Wahl ist Deutsch
!insertmacro MUI_LANGUAGE "English"
!insertmacro MUI_RESERVEFILE_LANGDLL
# Diese Section wird auf jedem Fall ausgeführt, also Section ""
Section ""
ReadRegStr $RO HKLM "SOFTWARE\Microsoft\Windows NT\CurrentVersion" CurrentVersion
StrCmp $RO "" 0 NT
  MessageBox MB_OK "Die Installation benötigt ein Windows NT System! Abbruch."
  Abort # Kein NT dann ist Schluss
  NT:
# Erstellungen der Programmgruppe xamppWordPress
CreateDirectory "$SMPROGRAMS\xamppWordPress"
CreateShortCut "$SMPROGRAMS\xamppWordPress\XAMPP.lnk" "$INSTDIR\xampp\xampp-
control.exe" "" "$INSTDIR\xampp\install\xampp-icon.ico"
CreateShortCut "$SMPROGRAMS\xamppWordPress\UNINSTALL.lnk"
"$INSTDIR\xampp\uninstall.exe" "" "$INSTDIR\xampp\install\xampp-icon-uninstall.ico"
SectionEnd
# XAMPP wird angezeigt, darf aber nicht ausgewählt werden, also SectionIn RO
Section "XAMPP" SEC01
SectionIn RO
SetOutPath "$INSTDIR\xampp"
SetOverwrite ifnewer
File /r "D:\xampp\*.*)" # Alle XAMPP Dateien lokal
# Registry Einträge unter HKey Local Maschine
WriteRegStr HKLM "Software\xamppwordpress" "Install_Dir" "$INSTDIR\xampp"
WriteRegStr HKLM "Software\xamppwordpress" "wordpress" "0"

```

26.1 Der XAMPP NSIS Installer mit eigenem Programm

```

WriteUninstaller "$INSTDIR\xampp\Uninstall.exe"
# Registry-Einträge des Uninstallers unter HKey Local Maschine
WriteRegStr HKLM
"Software\Microsoft\Windows\CurrentVersion\Uninstall\xamppwordpress" "DisplayName"
"XAMPP + WordPress"
WriteRegStr HKLM
"Software\Microsoft\Windows\CurrentVersion\Uninstall\xamppwordpress"
"UninstallString" '"$INSTDIR\xampp\uninstall.exe"'
WriteRegDWORD HKLM
"Software\Microsoft\Windows\CurrentVersion\Uninstall\xamppwordpress" "NoModify" 1
WriteRegDWORD HKLM
"Software\Microsoft\Windows\CurrentVersion\Uninstall\xamppwordpress" "NoRepair" 1
SectionEnd
# Diese Section darf abgewählt werden
Section "WordPress" SEC02
SetOutPath "$INSTDIR\xampp"
SetOverwrite ifnewer
File /r "D:\nsi-wordpress\xampp\*.*" # Lokale Files im Ordner \nsi-wordpress\xampp
WriteRegStr HKLM "Software\xamppWordPress" "wordpress" "1" # Registry für WordPress
CreateShortCut "$SMPROGRAMS\xamppWordPress\WordPress.lnk" '"$PROGRAMFILES\Internet
Explorer\Iexplore.exe" "http://localhost/wordpress/"' ""
"$INSTDIR\xampp\install\wordpress.ico"
SectionEnd
# Ebenfalls eine obligatorische Section zum Abschluss der Installation
Section ""
##### Notwendig: Die Anpassung der XAMPP Pfade durch die install.php mit awk
#####
ExecWait '"$INSTDIR\xampp\php\php.exe" -n -d output_buffering=0
"$INSTDIR\xampp\install\install.php"' $4
# XAMPP Control Panel auf Desktop?
MessageBox MB_YESNO|MB_ICONQUESTION "Möchten Sie das XAMPP Kontrollzenter als Link
auf Ihrem Desktop anlegen?" IDNO NoDesktopIcon
CreateShortCut "$DESKTOP\XAMPP.lnk" "$INSTDIR\xampp\xampp-control.exe" ""
NoDesktopIcon:
SectionEnd
# Sprachauswahl
Function .onInit
!insertmacro MUI_LANGDLL_DISPLAY
FunctionEnd
# Section Uninstaller, obligatorisch
Section "Uninstall"
MessageBox MB_YESNO|MB_ICONQUESTION "Alle XAMPP Server sowie das Control Panel
müssen bei einer Deinstallation geschlossen sein. Sind Sie sich sicher?" IDNO
NoDeInstall
Abort
NoDeInstall:
RMDir /r "$INSTDIR"
DeleteRegKey HKLM "Software\xamppwordpress"
DeleteRegKey HKLM

```

```
"Software\Microsoft\Windows\CurrentVersion\Uninstall\xamppwordpress"
Delete "$SMPROGRAMS\xamppWordPress\*.*"
RMDir "$SMPROGRAMS\xamppWordPress"
Delete "$DESKTOP\XAMPP.lnk"
SectionEnd
```

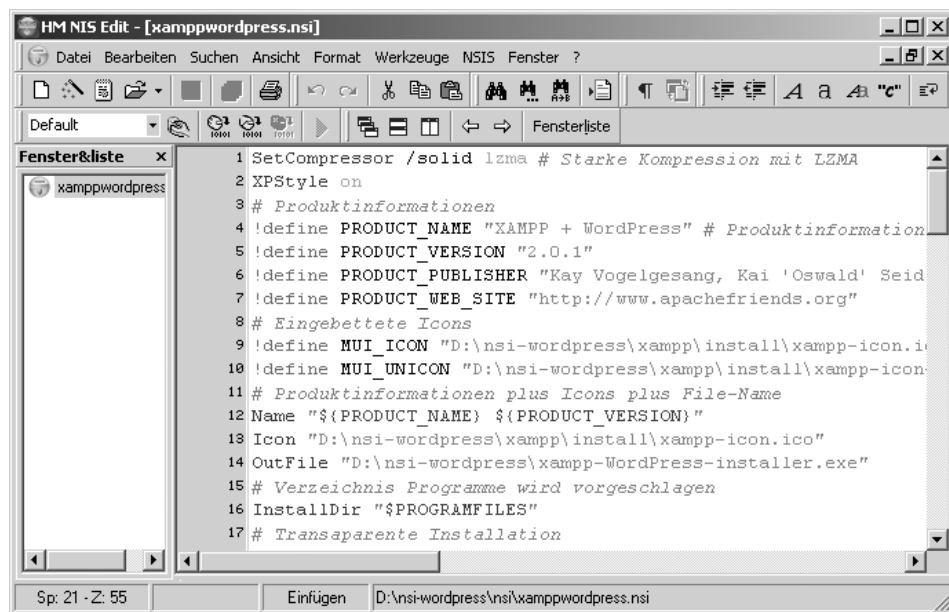


Abbildung 26.4: Mit dem HM NIS-Editor lassen sich NSI-Skripts besser bearbeiten

Hinweis



Wahrscheinlich haben Sie es schon gemerkt: Der XAMPP-Installer lässt sich *nicht* direkt auf C:\ installieren. Der Grund ist in der Architektur des XAMPP zu suchen. Denn ursprünglich gab es XAMPP nur als ZIP- bzw. 7-zip-Version, die immer einen gleichnamigen Hauptordner benötigte. Ohne dieses Stammverzeichnis kann die install.php nicht die absoluten Pfade für die Konfigurationsdateien einrichten. Deshalb also diese Definition von SetOutPath:

```
SetOutPath "$INSTDIR\xampp"
```

Für XAMPP 2.0 ist eine flexiblere Lösung angedacht.



Abbildung 26.5: XAMPP 1.5.1 lässt sich nur in Unterverzeichnisse installieren.

Nun kann dieses Kapitel keine Referenz für das NSIS bieten. Eine vorwiegend englischsprachige Dokumentation erhalten Sie unter <http://nsis.sourceforge.net/Docs/>. Trotzdem möchten wir hier ein paar kurze Anmerkungen geben.

Das Skript unterteilt sich in einen grundlegenden Abschnitt zu Beginn und mehrere darauf folgende sog. *Sections*. Die Sections ohne Attribut werden in jedem Fall ausgeführt (z.B. Section ""). Mit Attribut wird eine Section angezeigt und damit ist sie auch abwählbar.

```
Section "WordPress" SEC02
SectionEnd
```

Eine abwählbare, aber sichtbare Sektion ist die Section "XAMPP" mit SectionIn R0:

```
Section "XAMPP" SEC01
SectionIn R0
SectionEnd
```

Die Ausführung der `install.php` sollte immer gegen Ende der Installation erfolgen. Sie passt die Pfadangaben der Konfigurationsdateien an die Vorgaben des aktuellen Systems an.

```
ExecWait "$INSTDIR\xampp\php\php.exe" -n -d output_buffering=0
"$INSTDIR\xampp\install\install.php" $4
```

Damit die Deinstallationsroutine Ihres Nullsoft-Paketes wie bei jeder guten Software auch unter SYSTEMSTEUERUNG > SOFTWARE registriert ist, sollten die Einträge Ihres Uninstallers in der Windows Registry immer in dem folgenden Bereich liegen:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Uninstall
```

Mit den Einträgen wird unser Uninstaller unter SYSTEMSTEUERUNG > SOFTWARE registriert und kann darüber natürlich auch aufgerufen werden.



Tipp

Sie können selbstverständlich ein eigenes grafisches Interface konstruieren, um Ihre Komponenten und Bestandteile von XAMPP gezielt zu steuern. Wer hierbei nicht mit VC++ oder Visual Basic arbeiten möchte, sollte sich das *AutoIt-Projekt* näher anschauen: <http://www.autoitscript.com>. Mit der Basic ähnlichen Skriptsprache *AutoIt* in der Version 3 ist es möglich, auf relativ einfache Art und Weise auch komplexere *Graphical User Interfaces (GUIs)* zu erstellen.

26.1.3 Die Installer-CD-ROM

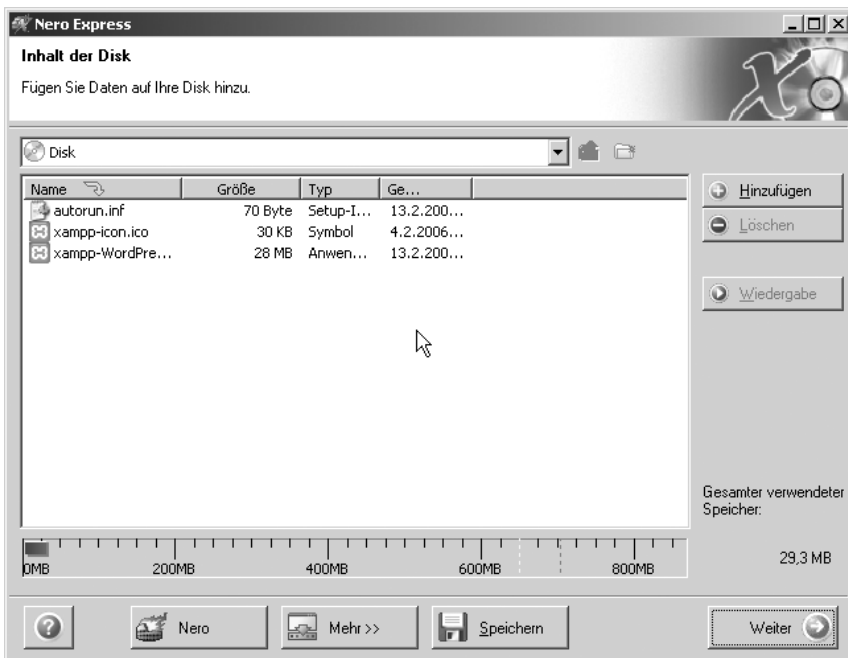


Abbildung 26.6: Eine XAMPP-Installer-CD-ROM mit dem Brennprogramm Nero erstellen

Wenn Sie nun Ihren Installer fertig gestellt haben, brauchen Sie eigentlich nur noch eine `autorun.inf` zu erstellen. Ein Beispiel:

```
[autorun]
icon=.\xampp-icon.ico
open=.\xampp-WordPress-installer.exe
```

Brennen Sie die drei Dateien `xampp-WordPress-installer.exe`, `xampp-icon.ico` und `autorun.inf` als Daten-Disk auf eine CD-ROM. Und vergessen Sie bitte danach das Testen nicht!

26.2 Das eigene XAMPP-Add-On

Während wir an diesem Buch schrieben, arbeiteten wir auch an einer Auffrischung des Installationsmechanismus für den *XAMPP 2.0*. In den 1er-Versionen des XAMPP sind *Add-Ons* nur ohne eine Anpassung von Pfadangaben in zusätzlichen Konfigurationsdateien möglich. Allerdings arbeiten wir an einem System, das die Funktionalität der `install.php` voll nutzt. Zudem können Add-Ons zukünftig automatisch als Link im HTML-Menü aufgenommen werden. Das Vorgehen für den Eigenbau dieser standardisierten Add-Ons wird mit der Etablierung des neuen Systems auf der Windows-Seite des XAMPP zeitnah veröffentlicht. Hier schildern wir Ihnen somit ein Add-On, das keine Pfadanpassungen in Konfigurationsdateien benötigt.

Als Beispielprogramm möchten wir ein *Wiki für Dokumentationen*, hier *DokuWiki* (<http://wiki.splitbrain.org/wiki:dokuwiki>) über ein Add-On für den XAMPP bereitstellen. Dafür sollen drei Pakete erstellt werden:

```
xampp-dokuwiki-addon-2005-09-19.zip
xampp-dokuwiki-addon-2005-09-19.exe
xampp-dokuwiki-addon-2005-09-19-installer.exe
```

Die Versionsnummer bezieht sich auf das aktuelle Paket von *DokuWiki*.

26.2.1 DokuWiki anpassen

Zz Beginn richten Sie die von Ihnen gewählte Applikation, hier *DokuWiki*, lokal auf dem XAMPP ein.

Eine Online-Dokumentation zu *DokuWiki* finden Sie unter der URL:

<http://wiki.splitbrain.org/wiki:DokuWiki>

1. Extrahieren Sie die `dokuwiki-x-y-z.tgz` direkt in das Verzeichnis `<xampp>\htdocs`. Benennen Sie dort den Ordner in `dokuwiki` um.
2. Erstellen Sie eine leere Datei mit dem Namen `changes.log` in dem `data`-Ordner:

```
<xampp>\htdocs\dokuwiki\data\changes.log
```

3. Editieren Sie die `local.php.dist` und speichern Sie diese als `local.php` ab:

```
<xampp>\htdocs\dokuwiki\conf\local.php
```

Beispiel:

```
$conf['title'] = 'XAMPP Wiki';
$conf['useacl'] = 1;
$conf['superuser'] = 'root';
```

4. Konfigurieren Sie die `dokuwiki.php` nach Ihren Vorstellungen.

```
<xampp>\htdocs\dokuwiki\conf\dokuwiki.php
```

5. Benennen Sie die `users.auth.php.dist` und `acl.auth.php.dist` in `users.auth.php` und `acl.auth.php` um. Passen Sie auch hier die Konfiguration bei Bedarf an.

```
<xampp>\htdocs\dokuwiki\conf\acl.auth.php
<xampp>\htdocs\dokuwiki\conf\users.auth.php
```

6. Rufen Sie die Seite `http://localhost/dokuwiki/` auf, um möglicherweise weitere Einstellungen vorzunehmen.



Abbildung 26.7: Das DokuWiki im XAMPP

26.2.2 Das Add-On als ZIP und 7-zip-exe

Um das Add-On nun als ZIP-Paket anfertigen zu können, richten Sie ein neues Arbeitsverzeichnis für das Add-On ein. Es könnte `d:\xampp-dokuwiki-addon` heißen. Darin erstellen Sie die folgenden Unterordner:

26.2 Das eigene XAMPP-Add-On

```
\xampp-dokuwiki-addon\htdocs
\xampp-dokuwiki-addon\licenses\dokuwiki
```

In das `htdocs`-Verzeichnis kopieren Sie nun Ihren bereits angepassten `dokuwiki`-Ordner. Die Lizenz kommt danach selbstverständlich in den `licenses\dokuwiki`-Unterordner. Zum Abschluss erstellen Sie noch im Arbeitsverzeichnis eine `Readme` (hier: `readme-addon-dokuwiki.txt`) mit wichtigen Angaben und Hinweisen zu Ihrem Add-On.

Damit haben Sie die Datei- und Verzeichnisstruktur bereits fertig gestellt. Jetzt packen Sie den Inhalt Ihres Add-On-Ordners mit 7-zip (<http://www.7-zip.org>) und erstellen so die Pakete. In diesem Beispiel:

```
xampp-dokuwiki-addon-2005-09-19.zip
xampp-dokuwiki-addon-2005-09-19.exe
```

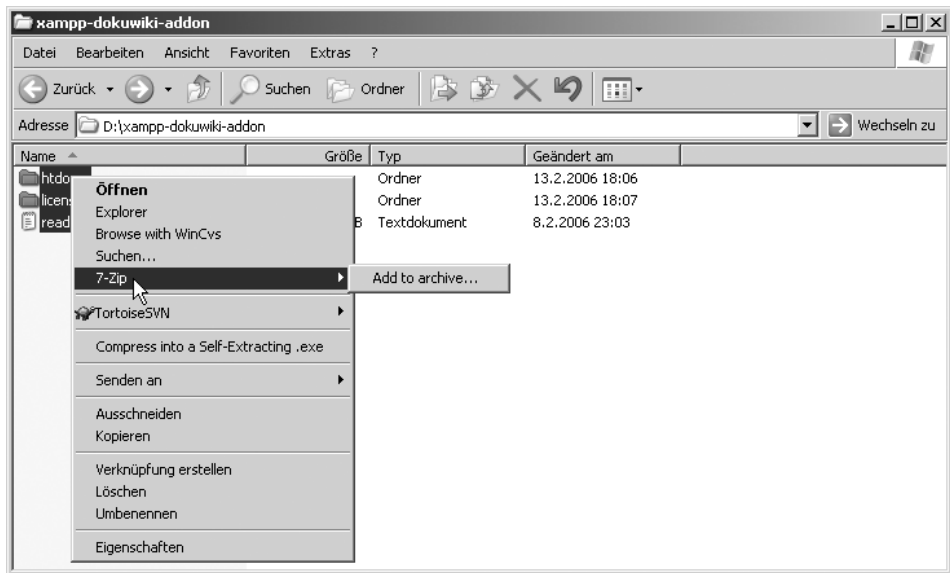


Abbildung 26.8: Den Inhalt des Add-Ons mit 7-zip zu einem neuen Archiv hinzufügen

Entfernen Sie die fertigen Pakete aus dem Arbeitsbereich und sichern Sie diese an einem anderen Ort, um nun den Installer einzurichten.

26.2.3 Das Add-On als Installer

Für die Installer-Version des Add-Ons benutzen wir die *Icons* aus Kapitel 26.1. Sie erhalten die XAMPP-Icons unter der folgenden URL:

<http://www.apachefriends.org/files/misc/icons>

Ein *Uninstaller* ist hier nicht notwendig, mithin auch keine zusätzlichen Einträge in der *Registry*. Die Entfernung des Add-Ons erfolgt automatisch mit der Deinstallation des XAMPP-Pakets. In diesem Fall kann der *dokuwiki*-Ordner aber auch problemlos über den *Explorer* gelöscht werden.

NSIS-Beispielskript: `xampp-addon-dokuwiki.nsi`

Listing 26.2: NSIS-Skript für einen Add-On-Installer

```
SetCompressor /solid lzma # Starke Kompression mit LZMA
XPStyle on
# Produktinformationen
!define PRODUCT_NAME "XAMPP DokuWiki Addon"
!define PRODUCT_VERSION "19-09.2005"
!define PRODUCT_PUBLISHER "Kay Vogelgesang, Kai 'Oswald' Seidler"
!define PRODUCT_WEB_SITE "http://www.apachefriends.org"
# Eingebettete Icons
!define MUI_ICON "D:\nsi-wordpress\xampp\install\xampp-icon.ico"
!define MUI_UNICON "D:\nsi-wordpress\xampp\install\xampp-icon-uninstall.ico"
# Produktinformationen plus Icons plus File-Name
Name "${PRODUCT_NAME} ${PRODUCT_VERSION}"
Icon "D:\nsi-wordpress\xampp\install\xampp-icon.ico"
OutFile "D:\nsi-wordpress\xampp-dokuwiki-addon-2005-09-19-installer.exe"
# Transparente Installation
ShowInstDetails show
ShowUninstDetails show
# Wir benutzen hier das Modern user interface
!include "MUI.nsh"
# Verschiedene Macros z.B. für die Auswahl von Komponenten
!insertmacro MUI_PAGE_LICENSE "D:\xampp-dokuwiki-addon\htdocs\dokuwiki\COPYING"
!insertmacro MUI_PAGE_COMPONENTS
!insertmacro MUI_PAGE_DIRECTORY
!insertmacro MUI_PAGE_INSTFILES
!insertmacro MUI_UNPAGE_CONFIRM
!insertmacro MUI_UNPAGE_INSTFILES
!insertmacro MUI_LANGUAGE "German" # Erste Wahl ist Deutsch
!insertmacro MUI_LANGUAGE "English"
!insertmacro MUI_RESERVEFILE_LANGDLL
# Zu Beginn wird geprüft, ob XAMPP Installer vorhanden
Function .onInit
!insertmacro MUI_LANGDLL_DISPLAY
ReadRegStr $INSTDIR HKLM "Software\xampp" "Install_Dir"
StrCmp $INSTDIR "" 0 NoAbort
MessageBox MB_OK "XAMPP win32 Installer nicht gefunden! Installation wird
abgebrochen."
Abort
NoAbort:
FunctionEnd
# Die DokuWiki wird angezeigt, darf aber nicht ausgewählt werden
```

```

Section "DokuWiki" SEC01
SectionIn R0
SetOutPath $INSTDIR
SetOverwrite ifnewer
File /r "D:\xampp-dokuwiki-addon\*.*" # Der gesamte DokuWiki Bereich wird gepackt
SectionEnd

```

Wichtig bei der Erstellung des Add-Ons ist die Suche nach XAMPP zu Beginn der Installation. Findet der Installer keine entsprechenden Werte in der Registry, bricht er die Installation ab. Bei Erfolg wird der Registry-Wert in `Install_Dir` als Zielverzeichnis zum XAMPP definiert.

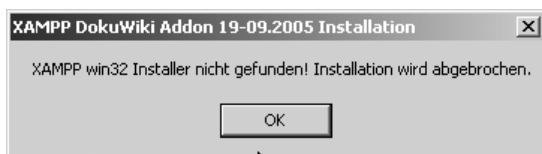


Abbildung 26.9: XAMPP nicht gefunden, also Abbruch der Installation

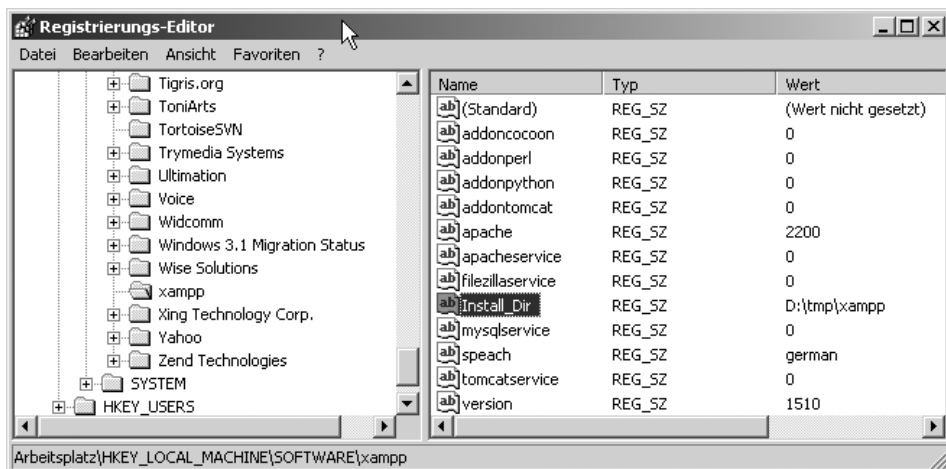


Abbildung 26.10: Die Windows-Registrierung durch den XAMPP



Stichwortverzeichnis

Symbole

@icon sushi 505

A

ab, ab.exe 191

Access 18, 179, 181f., 279

lesen 180

ADODB 181ff.

ADONewConnection() 182

Close() 183

Connect() 182

Datenbanktypen 181

ErrorMsg() 183

Execute() 183

MoveNext() 183

Anonymous FTP 18, 233

Apache 51

.htaccess 56f., 64, 199, 297

.htpasswd 56f.

access_log 195

Alias 66ff., 480

Allow 66f., 69

AllowOverride 55, 199f., 268

Apache Control Utility 83

AuthDigestDomain 68f.

Authentifikationsmethoden 64

AuthName 57, 66f., 69

AuthType Basic 57, 64ff., 69

AuthType Digest 64, 68f.

AuthUserFile 57, 66ff.

BrowserMatch 66ff.

CustomLog 195

Dav On 66f., 69

DavLockDB 66ff.

Deny 55

Directory-Container 55, 66ff., 268f.

DirectoryIndex 197f.

DocumentRoot 53, 55, 199

Einrichtung als Dienst 35

EnableMMap 196

EnableSendfile 196f.

error_log 195

ErrorLog 195

ExecCGI 216

FollowSymLinks 55, 202f.

HostnameLookups 195f.

htdigest-Befehl 64, 68

htpasswd-Befehl 56f., 65ff., 69

httpd-autoindex.conf 51

httpd-dav.conf 51, 65ff.

httpd-default.conf 51

httpd-info.conf 51

httpd-languages.conf 51

httpd-manual.conf 51

httpd-mpm.conf 51

httpd-multilang-errorlog.conf 51

httpd-ssl.conf 51, 58

httpd-userdir.conf 51, 268f.

httpd-vhosts.conf 51, 53, 56, 480

httpd-xampp.conf 51, 215, 480

httpd.conf 51ff., 56, 65f., 76, 80, 141, 186f.,
195, 203, 460, 480

IfModule-Container 67ff., 268, 414

Include 51, 53, 56, 65

Kompilieren (Linux) 382

Kompilieren (Windows) 366

Limit 57

Limit-Container 268

LimitExcept-Container 66ff.

LoadModule 186, 200f., 268, 414

mod_ssl.so 58

mod_userdir 268

MultiViews 268

NameVirtualHost 53f.

Options 55, 203, 216, 268

Order 55, 66f., 69

Proxy-HTTP-Modul 186

Proxy-Modul 186

require valid-user 57, 66ff.

Rewrite Engine 460

Rewrite-Modul 186

RewriteEngine 186

RewriteRule 179, 186f.
 ServerAdmin 54f.
 ServerName 53, 55
 ServerTokens 198f.
 SetHandler 216
 SymLinksIfOwnerMatch 202f., 268
 Verzeichnisschutz 56
 Virtual Hosts 55
 VirtualHost-Container 53ff., 480
 xampp-perl.conf 215
 apache_installservice.bat 76
 ApacheBench 189ff.
 APXS 399
 apxs 398
 AuthName 68
 Authoritative Name Server 486
 AutoIt 510
 autorun.inf 511

B

Backups
 Linux 229
 Windows 239
 Benchmark 189f., 207
 ApacheBench 189
 httpperf 190
 Web Application Stress Tool 190
 Benutzerauthentifizierung 134
 BIND 263
 named.conf 493f.
 allow-query 495f.
 allow-recursion 495
 options 493, 495f.
 recursion 495f.
 named.conf.local 494, 496
 named.conf.options 494
 Bluefish 85
 Borland 333

C

C++-Compiler 363
 C-Compiler 391
 Caching Name Server 494f.
 cds-fpdf.php 95
 cds.php 86
 cds.xml 102
 cds_auth.php 136, 138
 chgrp 225
 chmod 225, 458, 460f.
 chown 225, 336, 345, 412, 414, 420, 437, 445, 447, 471, 480
 Client-Server-Architektur 23
 Cocoon 39

COM-Anwendungen 179
 Component Object Model (COM) 180
 configure 380ff.
 Vorlagen 383ff.
 configures.tar.gz 383f.
 CPAN 403
 cpan 403, 406
 install 407
 quit 408
 Shell 406ff.
 Cross-Site Scripting (XSS) 244, 246

D

Daemon 217
 Debian 28, 411, 460, 491, 494, 496ff.
 DHCP 262, 498
 Dienst 217
 DNS 52, 167
 A Record 249, 487, 490, 497
 BIND siehe BIND
 CNAME Record 487
 DNS-Server 485
 Master 486
 MX Records 167, 249, 487f., 490, 497
 NS Record 488, 490, 497
 Resource Records 486
 Root Server 485
 Server 167
 Slave 486
 SOA Record 487ff., 497
 expire 489
 refresh 489
 retry 489
 serial 489
 TTL 490
 Zone 486
 Zone File 486, 490, 496ff.
 Document Object Model siehe DOM
 DokuWiki 511f.
 DOM 101f., 107ff.
 appendChild() 108ff.
 createElement() 108ff.
 createTextNode() 109ff.
 DOMDocument() 108, 111
 DOMDocument-Klasse 108
 PHP-Beispiel 107
 saveXML() 110f.
 setAttribute() 109, 111
 dom.php 107, 110f.
 Domain Name System siehe DNS
 Domain-Controller 262
 Dr. McCoy 18, 485
 Dreamweaver 64, 69

DynDNS 238, 249ff., 253ff.

ddclient 253, 256f.

ddclient.conf 256f.

Port-Forwarding 250, 252, 257

E

eAccelerator 206f.

EasyPHP 26

Excel 18, 179f., 183ff.

Datei 184f.

Extensible Markup Language siehe XML

F

Fake sendmail 174ff.

auth_password 175

auth_username 175

sendmail.exe 174

sendmail.ini 174ff.

smtp_server 175

FDB-Format 123

Fedora 226f., 491, 494

FFT-Format 123

FileZilla 151, 218

Client 151, 161f.

Einrichtung als Dienst 35, 151

FileZilla Server.exe 151f.

FileZilla Server.xml 151

filezilla_setup.bat 151f.

filezilla_start.bat 151

filezilla_stop.bat 151

Server 18, 23, 25, 151ff., 158, 160f., 218, 233, 242, 267, 483

Server Interface 153ff., 157, 160f.

SSL/TLS 160ff.

Virtual Hosts 154

Firebird 19, 333ff., 384

Benutzer einrichten 336

Datenbank einrichten 336

DBD-Treiber 340

gsec 333ff.

ibWebAdmin 344

Installation (Linux) 334

Intallation (Windows) 333

isql 335ff.

ODBC-Treiber (Windows) 340

FireFox Portable 38

Firewalls 226

Hardware-Firewall 226, 238

iptables 227f.

Kerio Personal Firewall 238

Personal Firewalls 226, 235f.

Schönherr Personal Firewall 238, 253

Software-Firewalls 226

Windows XP-Firewall 236

ZoneAlarm 238, 253

Flash 120ff., 129f., 132

on-the-fly erzeugen 121

Flash Gordon 121

FoxServ 26

FPDF 86, 95ff.

Add-Ons 100

AddPage() 97

Cell() 98f.

FPDF() 95f.

Ln() 98ff.

Maßeinheiten 96

PHP-Beispiel 95

Schriften 97

Seitengrößen 96

Seitenorientierung 96

SetFont() 97ff.

SetFontSize() 98

Write() 97f.

FreeTDS 307, 309

freetds.conf 307ff.

freetds.php 309

FTP 257

Server 141f., 146, 150f., 153f., 156, 159ff.

Fully Qualified Domain Name (FQDN) 167, 175, 483

G

Gallery 19, 457ff.

GCC 391, 406

Gnu-Plot 193

GoLive 64, 70

Google 117f., 120

API 117

API-Key 117f.

H

Hash-Algorithmen 139

hosts-Datei

Linux 53, 484

Windows 53, 484

HTTP-Auth 136, 139

httpperf 190

HyperText Transfer Protocol Secure (HTTPS) 58

I

ibWebAdmin 344f.

ifconfig 251

ImageMagick 443f., 448, 450

IMAP 218, 257, 464ff., 469

Server 165, 465, 467

Interbase 19, 333, 338, 340, 342f., 384
 interbase.php 338
 ipconfig 251, 263
 ISO 8859-1 108, 111, 113

J

Java 19, 333, 409ff., 415f., 418, 420f., 423ff., 427, 429, 431, 433
 Java Server Pages siehe Tomcat
 JavaScript 246f.
 JDK 39
 Joomla 19, 437ff.

K

Kaiser Ming 121
 kill.exe 31

L

LAMPP 30
 Latin1 113
 libaio 318
 localhost 53
 loopback 53

M

Mail Transport Agent 167
 makefdb 123
 MD5-Hash 33
 Mercury Mail Transport System
 165ff., 170f., 173
 Abwesenheitsbenachrichtigungen 173
 Autostart 166
 Benutzer 169
 CRAM-MD5 171
 IMAP-Postfach 166
 LOGIN 171
 mercury.exe 166
 mercury.ini 166
 Passwörter 170
 POP3-Postfach 166
 Postausgangsordner 166
 Postfächer 169
 queue 166
 Relaying 171
 SMTP-Authentifizierung 171f.
 SMTP-Autoresponder 173
 Ming 121ff., 130f., 133
 add() 125ff., 131ff.
 addEntry() 130, 133
 addFill() 130, 133
 addString() 125, 127, 131, 133
 Balkendiagramm 129
 drawLineTo() 130, 133

movePenTo() 130, 133
 moveTo() 124ff., 131ff.
 nextFrame() 126ff., 132, 134
 output() 100, 127f., 132, 134
 PHP-Beispiel 122
 rotateTo() 125, 127f.
 scale() 125, 127
 scaleTo() 130ff.
 setBackground() 123, 127, 130, 132
 setColor() 124, 127, 131, 133
 setDimension() 123, 127, 130, 132
 setFont() 124, 127, 131, 133
 setHeight() 124, 127, 131, 133
 setName() 133
 setRate() 123, 127, 130, 132
 setRightFill() 130, 133
 SWFAction() 132, 134
 SWFFont() 122, 124, 127, 130, 132
 SWFGradient() 130, 133
 SWFMovie() 123, 126f., 130, 132
 SWFShape() 130, 133
 SWFText() 124, 126f., 131, 133
 mingstats_swf.php 129, 132
 mingswf.php 122
 mod_jk 39
 mod_perl 30, 39
 modprobe 253
 ModSecurity 397
 Kompilieren (Linux) 397, 399
 Konfigurationsbeispiel 400
 MS SQL 19, 239, 299, 305, 307ff., 347, 434
 Beispieldatenbank einrichten 305
 FreeTDS 307
 Microsoft SQL Desktop Engine 299f., 304
 Microsoft SQL Server 2000 307
 Microsoft SQL Server 2005 Express
 Edition 299, 307
 MSDE 299
 Zugriff mit Perl 310
 mssql.php 310
 mssql.pl 313
 MTA 167
 MySQL 17ff., 23f., 27, 31, 35, 37, 41f., 44ff., 87, 101, 107f., 136f., 139, 179, 218, 220ff., 229ff., 236f., 240ff., 261f., 270, 275ff., 285ff., 296, 305, 344, 347, 352, 358, 363f., 368f., 371f., 374, 407, 426f., 430, 432, 438f., 447f., 457, 459f., 470ff., 501ff.
 BDB 211
 Benutzer einrichten 282
 Datenbank einrichten 281
 DBD-Treiber 285
 Einrichtung als Dienst 35
 InnoDB 208, 211

JDBC 425, 427
 key_buffer_size 212
 Kompilieren (Linux) 383
 Kompilieren (Windows) 368
 Locking-Methoden 211
 log_slow_queries 213
 long_query_time 213
 max_connections 210f.
 max_used_connections 210f.
 my-huge.cnf 208
 my-innodb-heavy-4G.cnf 208
 my-large.cnf 208
 my-medium.cnf 208
 my-small.cnf 207
 my.cnf 208
 my.ini 208
 MyISAM 211
 mysql 277
 MySQL Administrator 280
 MySQL Connector/J 5.0 425
 mysqladmin 278
 mysqlbinlog 278
 mysqlcheck 278
 mysqldump 278
 mysqldumpslow 213f.
 mysqlimport 278
 MySqlManager 278
 mysqlshow 279
 mysqltext 279
 ODBC-Connector 179
 open_tables 209f.
 opened_tables 210
 page-level locking 211
 PASSWORD() 139
 row-level locking 211
 Slow Query Log 213
 Starten und Stoppen (Linux) 275f.
 table-level locking 211
 table_cache 208ff.
 table_locks_immediate 211
 table_locks_waited 211
 Uptime 209
 Zugriff mit Perl 285
 Zugriff mit PHP 282

N

Netzwerkordner 71
 Nullsoft Scriptable Install System (NSIS) 501
 NuSphere 26

O

OpenSSL 364
 Kompilieren (Linux) 383
 openSUSE 28, 226, 350, 491

Oracle 17, 19, 218, 275, 279, 315ff., 323ff.,
 328ff., 347, 430ff., 460
 Beispielbenutzer aktivieren 322
 Beispieltabelle einrichten 322
 Connect Descriptor 325
 Connect String 325
 DBD-Treiber 329
 Easy Connect 326
 Express Edition 315ff.
 Fehlermeldungen 331
 Installation (Linux) 317
 Installation (Windows) 316
 Instant Client 323ff.
 JDBC 430
 Oracle Database 10g 315
 Oracle Express 321
 Zugriff mit Perl 329
 Zugriff mit PHP 325
 oracle.php 325, 327f.

P

Passwortschutz 56
 pdf class 86f., 91f., 95ff.
 Absatzformatierungen 91
 Cezpdf() 87, 89, 94
 ezStream() 94f.
 ezTable() 92ff.
 ezText() 90f., 94
 Schriften 90
 Seitengrößen 87
 Seitenorientierung 87
 selectFont() 89, 94
 Tabellen 92
 PDF generieren 85
 PEAR 387f., 391f., 403
 install 389
 list 388
 list-all 388
 list-upgrades 389
 pear 388ff.
 pear.bat 388
 uninstall 391
 upgrade 390
 upgrade-all 390f.
 XML_RSS 112
 XML_sql2xml 115
 PECL 387f., 391, 403
 install (Linux) 391
 install (Windows) 393
 pecl 391, 393
 uninstall 392
 Performance Tuning 189, 194, 215
 Caching 204
 CGI 214

- MySQL 207
- Perl 214
- PHP 204
- zweiter Server 204
- Perl 39, 403
 - ActivePerl 403
 - connect() 285, 287, 295, 313, 330, 341f., 357
 - CPAN 403
 - cpan 342
 - disconnect() 286, 288, 296, 313, 330, 341, 343, 357
 - execute() 286f., 295, 313, 330, 341f., 357
 - fetchrow_array() 286f., 296, 313, 330, 341, 343, 357
 - finish() 286, 288, 296, 330, 343, 357
 - FTP 159
 - mod_perl 215, 285
 - PerlModule 216
 - PerlOptions 216
 - PerlResponseHandler 216
 - PPM 403ff.
 - install 406
 - query 405
 - quit 406
 - remove 406
 - search 406
 - upgrade 406
 - prepare() 286f., 295, 313, 330, 341f., 357
- PHP
 - addslashes() 137, 246
 - deaktivieren 202
 - die() 245
 - Entwicklungsumgebungen 73
 - fclose() 105, 107
 - fopen() 105, 107
 - fread() 105, 107
 - ftp_connect() 158
 - ftp_delete() 159
 - ftp_get() 159
 - ftp_login() 158
 - ftp_nlist() 158
 - ftp_put() 158
 - ftp_quit() 159
 - ftp_size() 158
 - hash.so 392
 - header() 110ff., 116, 120, 127f., 132, 134, 137f., 245
 - htmlentities() 247, 327f.
 - ibase_errmsg() 339
 - ibase_fetch_assoc() 339
 - ibase_pconnect() 339
 - ibase_query() 339
 - IniFilePath 80f.
 - interbase.so 338
 - Kompilieren (Linux) 384
 - Kompilieren (Windows) 370
 - magic quotes 244
 - mail() 177
 - md5sum() 139
 - mssql_close() 309f.
 - mssql_connect() 309f.
 - mssql_fetch_array() 310
 - mssql_select_db() 310
 - mysql_connect() 86f., 94, 108, 111, 136ff., 245, 283f.
 - mysql_fetch_array() 92, 95, 99f., 109, 111, 283f.
 - mysql_num_rows() 245
 - mysql_query() 92, 95, 99, 108, 111, 137, 245, 247, 283
 - mysql_select_db() 86, 94, 108, 111, 136ff., 245, 283f.
 - ob_gzhandler 204f.
 - ob_start() 205
 - oci_connect() 329
 - oci_execute() 329
 - oci_fetch() 329
 - oci_parse() 329
 - oci_parse() 329
 - oci_result() 329
 - oci_result() 329
 - ocierror() 328
 - ociexecute() 328
 - ociexecute() 326
 - ocifetch() 327f.
 - ocilogon() 325, 328f.
 - ociparse() 326, 328f.
 - ocireult() 327
 - ocireult() 327ff.
 - ocitererror() 327
 - pg_connect() 355
 - pg_errormessage() 355
 - pg_exec() 355
 - pg_fetch_array() 355
 - pg_numrows() 355
 - php.ini 76f., 79f., 84, 176, 205, 309, 353, 392, 394, 466
 - extension 338, 353, 392, 394, 466
 - output_handler 205
 - sendmail 165
 - sendmail_path 176
 - SMTP 176
 - smtp_port 176
 - php_hash.dll 394
 - php_imap.dll 466
 - php_interbase.dll 338
 - php_mssql.dll 309
 - php_pgsql.dll 353

- sqlite_error_string() 293f.
- sqlite_fetch_array() 293f.
- sqlite_open() 292ff.
- sqlite_query() 293f.
- phpBB 19, 470f., 473, 476, 480
- phpdev 26
- phpMyAdmin 41, 134f., 221f., 231, 270, 279ff., 296, 344, 358, 439, 447, 459, 471, 473f., 503
 - Tabelle einrichten 134
- phpPgAdmin 358, 360
 - config.inc.php 358
- PHPTriad 26
- ping 53
- POP3 175, 218, 257, 464
 - Server 165
- Portnummer 218
- Ports 218
 - im XAMPP-Umfeld 218
- postgres.php 354
- postgres.pl 356
- PostgreSQL 218, 347ff., 356ff., 384, 429, 460
 - Benutzer einrichten 352
 - createdb 352
 - createuser 352
 - Datenbank einrichten 352
 - DBD-PgPP-Treiber 356
 - Installation (Linux) 350
 - Installation (Windows) 348
 - JDBC 429
 - Lizenz 348
 - PHP-Erweiterung 353
 - phpPgAdmin 358
 - psql 352
 - Zugriff mit Perl 356
 - Zugriff mit PHP 354
- PrcView 32
- Primary Name Server 486, 488f., 495f.
- ProFTPD 18, 23, 25, 44ff., 141, 143f., 146ff., 218, 220, 230, 267, 483
 - AllowOverwrite 142
 - Anonymous FTP-Server 141, 145ff., 150
 - Anonymous-Container 147f.
 - DefaultRoot 142, 144
 - DenyAll 147f., 150
 - Directory-Container 142, 148, 150
 - Group 147f.
 - Kompilieren (Linux) 383
 - Limit-Container 147f., 150
 - proftpd.conf 141f., 144, 147f., 150
 - User 147f.
 - UserAlias 147f.
 - UserPassword 142
 - Virtual Hosts 141, 143ff.
 - VirtualHost-Container 144

- Protokoll 218
- pv.exe 31

R

- RDF Site Summary siehe Really Simple Syndication
- Really Simple Syndication siehe RSS
- RedHat 28, 491, 494
- resolv.conf 498
- Reverse Proxy 186
- Rich Site Summary siehe Really Simple Syndication
- RSS 101, 111ff., 115, 117
 - Heise-News 112, 114

S

- SAX 101ff., 106f.
 - Callback-Funktionen 102ff.
 - characterData-Event 102ff.
 - endElement-Event 102ff.
 - PHP-Beispiel 102
 - startElement-Event 102ff.
 - xml_parse() 105, 107
 - xml_parser_create() 103, 106
 - xml_set_character_data_handler() 103f., 106
 - xml_set_element_handler() 103, 106
- sax.php 103, 106f.
- Secondary Name Server 486, 488f.
- Secure Socket Layer (SSL) 58, 200
 - CA-zertifiziertes Serverzertifikat 59, 61
 - Certificate Signing Request 62
 - Certification Authority 61
 - challenge password 62
 - Common Name 60, 62
 - Country Name 60, 62
 - Email Address 60, 62
 - Locality Name 60, 62
 - mod_ssl.so 201
 - openssl-Befehl 58f., 61f.
 - openssl.cnf 58f., 62
 - optional company name 62
 - Organization Name 60, 62
 - Organizational Unit Name 60, 62
 - selbstzertifiziertes Serverzertifikat 58, 60f., 64
 - server key 59, 61
 - server.crt 59, 64
 - server.csr 62
 - server.key 59, 62
 - Serverschlüssel 58ff.
 - Serverzertifikat 63f.
 - ssl.crt 58

- ssl.key 58
- State or Province Name 60, 62
- Testzertifikat 61, 64
- Zertifikate 58ff.
- Zertifizierungsstelle 61, 63
- sendmail.exe 174
- sendmail.ini 174
- Server 217f.
- Service 217
- Servlets siehe Tomcat
- Sicherheit 217, 219
 - Cross-Site Scripting (XSS) 244
 - moderne Gefahren 244
 - Sicherheitscheck 220
 - SQL-Injections 244
 - XAMPP unter Linux absichern 219
 - XAMPP unter Windows absichern 231
- Simple API for XML 101
- SMTP 175, 177, 257
 - Server 165, 175
- SOAP 117ff.
 - API 117
 - doGoogleSearch() 118, 120
 - PHP-Beispiel 117
 - SoapClient() 118, 120
- soap.php 118ff.
- Sokkit 26
- Source Forge 27
- Spreadsheet Excel Writer 185
- Spreadsheet Excel Reader 183
 - read() 184
 - setOutputEncoding() 184
 - Spreadsheet_Excel_Reader() 179, 184
- SQL-Injections 137, 244, 246f.
- sql2xml.php 115f.
- SQLite 19, 281, 289ff., 294ff., 347
 - Datenbank einrichten 289
 - DBD-Treiber 295
 - phpSQLiteAdmin 296f.
 - SQLiteCC 296
 - Tools 296
 - Zugriff mit Perl 295
 - Zugriff mit PHP 292
- sqlite.php 292, 294
- sqlite.pl 295
- SquirrelMail 19, 464f., 467ff.
- SWF 121
- T**
- Tomcat 19, 29, 39, 218, 409ff., 429, 431, 434
 - Admin-Bereich 421
 - Apache Connector 413
 - CATALINA_HOME 411f., 419ff., 425f., 429, 431, 434
 - J2SE Development Kit 410, 415
 - Java Server Pages 409, 419, 432
 - JAVA_HOME 411ff., 415f., 419
 - JDK 410, 415
 - jkMount 414, 420
 - jkWorkersFile 414
 - Manager-Bereich 421
 - Servlets 409, 419
 - workers.properties 414, 419
- TrueType 123
- ttf2fft 123
- Typo3 19, 443ff.
- U**
- Ubuntu 226, 491, 494, 496ff.
- Uniform Server 26
- Unterricht 261
- USB-Stick 30, 37f.
- UTF-8 112ff., 118, 120
- V**
- Virensscanner 31
- Virtual Hosts 17f., 52
 - CustomLog 195
 - IP-basierte 52, 54
 - namensbasierte 52ff.
- Visual C++ 364, 368, 370
- W**
- WAMPP 30
- Weaverslave 38, 85
- Web Application Stress Tool 190
- WebDAV 64f.
 - Beispielkonfiguration (Linux) 65
 - Beispielkonfiguration (Windows) 66
- Webordner 71
- WordPress 501, 503f.
- World Wide Web Consortium (W3C) 101
- WU-FTPD 141
- X**
- XAMPP
 - Add-Ons 29f., 38ff.
 - auf Wechseldatenträgern 37
 - Batch-Dateien 35
 - Control Panel 31, 33, 353
 - deinstallieren 37, 45
 - Entwicklungs-Paket 391, 397, 399
 - herunterladen 32, 42
 - Installation 29
 - Linux 42
 - Windows-Installer 33
 - Windows-ZIP 33

- Lite-Version 30, 41
- Perl-Add-On 30, 215, 285, 295, 311, 340, 403f.
- Sicherheit 47
- starten 33, 44
- stoppen 36, 44
- Systemvoraussetzungen 28
- testen 35, 44
- Upgrades 41, 46
- verschieben 36
- Zielgruppe 25
- XHTML 101
- XML 101ff., 107ff., 115f.
- XML_RSS 112ff.
 - getChannelInfo() 113, 115
 - parse() 113, 115
- XML_sql2xml 115
 - getxml() 116
 - xml_sql2xml() 115f.
- xmlrss.php 112, 114
- Y**
- YaST 226
- Z**
- Zend
 - Optimizer 74, 79
 - Studio 73ff., 79, 81ff.
 - Installation (Linux) 81
 - Installation (Windows) 75
- Zlib 365
- Zugriffsrechte unter Linux 223ff.



... aktuelles Fachwissen rund
um die Uhr – zum Probelesen,
Downloaden oder auch auf Papier.

www.InformIT.de

InformIT.de, Partner von **Addison-Wesley**, ist unsere Antwort auf alle Fragen der IT-Branche.

In Zusammenarbeit mit den Top-Autoren von Addison-Wesley, absoluten Spezialisten ihres Fachgebiets, bieten wir Ihnen ständig hochinteressante, brandaktuelle Informationen und kompetente Lösungen zu nahezu allen IT-Themen.

The screenshot displays the InformIT.de website interface. The top navigation bar includes links for Home, MyInformIT, eBooks, Bücher, angli. Bücher, Software, Service, Training, and eLearning. A search bar is prominently featured. The main content area is divided into several sections:

- Bücher bei InformIT**: A section for browsing books, with a sidebar listing categories like Computer, Betriebssysteme, Client/Server, and more. The main list includes titles like "Photoshop CS für digitale Fotografie" and "Delphi 2005".
- Download des Tages**: A featured book section highlighting "Das VB.NET Codebook".
- Englisch Book des Tages**: A section for English books, featuring "Oracle SQL by Example".
- Unsere eBook Empfehlung**: A section recommending eBooks, such as "Small Business Server 2003".
- eBooks bei InformIT**: A section dedicated to eBooks, showing titles like "Small Business Server 2003" and "Tomcat 5".

At the bottom, a banner reads: **wenn Sie mehr wissen wollen ... www.InformIT.de**

THE SIGN OF EXCELLENCE

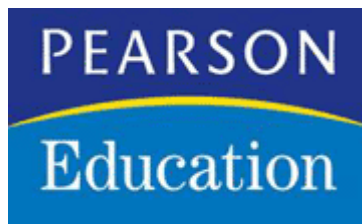
Sicher gegen Eingriffe von außen



Security-Fachmann Ralf Spenneberg führt mit diesem Buch tief in Konfiguration und Betrieb einer Linux-basierten Firewall ein. Nach einem Einstieg in die Technologie und Topologie zeigt er an einem einfachen Beispiel zunächst die Funktionsweise einer Firewall. Danach geht es in die Details: Firewalls in demilitarisierten Zonen (DMZ), Administrationshilfen, Filtern bestimmter Protokolle sowie die Einrichtung einer unsichtbaren Firewall.

Im Anhang beschreibt er Grundlagen der wichtigsten Protokolle IP, TCP, UDP und ICMP. Die CD enthält eine komplette Testumgebung zum einfachen Nachvollziehen der vielen Buch-Beispiele.

Ralf Spenneberg
3-8273-2136-0
59.95 EUR [D]



Copyright

Daten, Texte, Design und Grafiken dieses eBooks, sowie die eventuell angebotenen eBook-Zusatzdaten sind urheberrechtlich geschützt. Dieses eBook stellen wir lediglich als persönliche Einzelplatz-Lizenz zur Verfügung!

Jede andere Verwendung dieses eBooks oder zugehöriger Materialien und Informationen, einschliesslich

- der Reproduktion,
- der Weitergabe,
- des Weitervertriebs,
- der Platzierung im Internet, in Intranets, in Extranets,
- der Veränderung,
- des Weiterverkaufs
- und der Veröffentlichung

bedarf der schriftlichen Genehmigung des Verlags.

Insbesondere ist die Entfernung oder Änderung des vom Verlag vergebenen Passwortschutzes ausdrücklich untersagt!

Bei Fragen zu diesem Thema wenden Sie sich bitte an: info@pearson.de

Zusatzdaten

Möglicherweise liegt dem gedruckten Buch eine CD-ROM mit Zusatzdaten bei. Die Zurverfügungstellung dieser Daten auf unseren Websites ist eine freiwillige Leistung des Verlags. Der Rechtsweg ist ausgeschlossen.

Hinweis

Dieses und viele weitere eBooks können Sie rund um die Uhr und legal auf unserer Website



herunterladen