

Apache Compile HOWTO (Linux Edition) Deutsch

Table of Contents

Apache Compile HOWTO (Linux Edition) Deutsch	1
Luc de Louw	1
1. Einführung	2
1.1. Mitwirkende und Kontakte	2
1.2. Wieso dieses Dokument geschrieben wurde	3
1.3. Was dieses Dokument bezweckt	3
1.4. Was dieses Dokument nicht ist	3
1.5. Plattformen	3
1.6. Copyright Informationen	4
1.7. Haftungsausschlussklausel	4
1.8. Neuste Versionen dieses Dokuments	4
1.9. Credits	4
1.10. Feedback	4
1.11. Übersetzungen	4
1.12. Über den Autor	5
2. Voraussetzungen	5
2.1. Generell	5
2.2. OpenSSL	5
2.3. GNU Database System	6
2.4. MySQL	6
2.5. Kompilieren von mm	8
3. Downloaden, kompilieren und installieren von Apache und seinen wichtigsten Modulen	8
3.1. Download und auspacken des Apache Quellcodes	8
3.2. mod_ssl	9
3.3. mod_perl	10
3.4. Konfigurieren und Installieren von Apache	11
4. Zusätzliche Module	12
4.1. mod_dav	12
4.2. auth_ldap	12
4.3. mod_auth_mysql	13
4.4. mod_dynvhost	14
4.5. mod_roaming	15
5. Komprimierte Auslieferung	16
5.1. mod_gzip	16
5.2. mod_gunzip	17
6. mod_php seine Voraussetzungen	18
6.1. Was ist mod_php	18
6.2. Voraussetzungen	18
6.3. Kompilieren und installieren von PHP4	21
7. PHP Erweiterungen	22
7.1. APC (Alternative PHP-Cache)	22
7.2. Zend-Optimizer	22
8. Serverseitiges Java	23
8.1. mod_jserv	23
8.2. jakarta-tomcat	24
9. Weiterführende Informationen	26
9.1. Newsgroups	26
9.2. Mailing Listen	26
9.3. HOWTO	27
Notes	28
10. FAQ (Häufig gestellte Fragen)	29

Apache Compile HOWTO (Linux Edition) Deutsch

Luc de Louw

luc at delouw.ch

Revision History

Revision 1.9.16 2002-07-07

Software-Pakete auf den neusten Stand gebracht, LogFormat Konfiguration für mod_gzip hinzugefügt. gdbm zu den Voraussetzungen hinzugefügt. Verbesserungen im SGML Code (mehr Metadaten), FAQ Sektion neu gestaltet.

Revision 1.9.15 2002-06-24

Update auf Apache 1.3.26, ein update ist dringendst empfohlen (siehe CERT CA-2002-17), viele kleine Fehler beseitigt.

Revision 1.9.14 2002-06-16

Nicht in deutsch erschienen

Revision 1.9.13 2002-06-16

Software-Pakete auf den neusten Stand gebracht, und viele kleine Fehler beseitigt. Dokument Struktur verbessert, Module mod_gzip und mod_gunzip hinzugefügt. Weiterführende Quellen wurden hinzugefügt, FAQ (Fragen und Antworten) Sektion wurde hinzugefügt

Revision 1.9.8-1.9.12 2002-02-08

Nicht in deutsch erschienen. Eigentlich wollte ich dieses Dokument dem »Deutsches Linux HOWTO Projekt« zur Verfügung stellen, und wurde trotz Riesenaufwand meinerseits abgelehnt. Seitens des elitären Projekt Managers erlebte ich eine nie erlebte Ignoranz und Arroganz. Es hat leider eine ganze Weile gedauert bis ich wieder up-to-date war, sorry.

Revision 1.9.7 2001-12-27

Software-Pakete auf den neusten Stand gebracht, Test der Anleitung auf Linux S/390 (IBM Mainframes zSeries). Siehe "Plattformen" für mehr Infos, Kurzanleitung für Tomcat wurde hinzugefügt (nur Binär-Installation)

Revision 1.9.6 2001-10-27

Erste deutsche Fassung

Dieses Dokument beschreibt wie man den Apache Webserver und die wichtigsten dazugehörigen Module wie mod_perl, mod_dav, mod_auth_ldap, mod_dynvhost, mod_roaming, mod_jserv, und mod_php kompiliert.

Table of Contents

1. [Einführung](#)

- 1.1. [Mitwirkende und Kontakte](#)
- 1.2. [Wieso dieses Dokument geschrieben wurde](#)
- 1.3. [Was dieses Dokument bezweckt](#)
- 1.4. [Was dieses Dokument nicht ist](#)
- 1.5. [Plattformen](#)
- 1.6. [Copyright Informationen](#)
- 1.7. [Haftungsausschlussklausel](#)
- 1.8. [Neuste Versionen dieses Dokuments](#)
- 1.9. [Credits](#)
- 1.10. [Feedback](#)
- 1.11. [Übersetzungen](#)

- 1.12. [Über den Autor](#)
2. [Voraussetzungen](#)
 - 2.1. [Generell](#)
 - 2.2. [OpenSSL](#)
 - 2.3. [GNU Database System](#)
 - 2.4. [MySQL](#)
 - 2.5. [Kompilieren von mm](#)
3. [Downloaden, kompilieren und installieren von Apache und seinen wichtigsten Modulen](#)
 - 3.1. [Download und auspacken des Apache Quellcodes](#)
 - 3.2. [mod_ssl](#)
 - 3.3. [mod_perl](#)
 - 3.4. [Konfigurieren und Installieren von Apache](#)
4. [Zusätzliche Module](#)
 - 4.1. [mod_dav](#)
 - 4.2. [auth_ldap](#)
 - 4.3. [mod_auth_mysql](#)
 - 4.4. [mod_dynvhost](#)
 - 4.5. [mod_roaming](#)
5. [Komprimierte Auslieferung](#)
 - 5.1. [mod_gzip](#)
 - 5.2. [mod_gunzip](#)
6. [mod_php seine Voraussetzungen](#)
 - 6.1. [Was ist mod_php](#)
 - 6.2. [Voraussetzungen](#)
 - 6.3. [Kompilieren und installieren von PHP4](#)
7. [PHP Erweiterungen](#)
 - 7.1. [APC \(Alternative PHP-Cache\)](#)
 - 7.2. [Zend-Optimizer](#)
8. [Serverseitiges Java](#)
 - 8.1. [mod_jserv](#)
 - 8.2. [jakarta-tomcat](#)
9. [Weiterführende Informationen](#)
 - 9.1. [Newsgroups](#)
 - 9.2. [Mailing Listen](#)
 - 9.3. [HOWTO](#)
10. [FAQ \(Häufig gestellte Fragen\)](#)



Sicherheitslücke in Apache kleiner als 1.3.26

Benutzen Sie KEINE Version von Apache älter als 1.3.26. Siehe <http://www.cert.org/advisories/CA-2002-17.html> für mehr Informationen.

1. Einführung

1.1. Mitwirkende und Kontakte

Zuerst möchte ich allen Leuten danken, welche Fragen und Vorschläge sandten. Dies ermöglichte es mir, dieses HOWTO weiter zu entwickeln. Daher fordere ich alle auf, weiterhin kritische Fragen und Vorschläge einzusenden. Email genügt: <luc_at_delouw.ch>

1.2. Wieso dieses Dokument geschrieben wurde

Alle Distributionen welche von mir getestet wurden, haben eine schlechte Apache-Installation. Des weiteren sind viele dieser Versionen veraltet.

Dazu kommt, dass die meisten kommerziellen Unix Betriebssysteme selten mit einem vorinstallierten Apache geliefert werden.

Mein Job ist es auf verschiedenen Unix-Plattformen Webserver zu implementieren. Da ich etwas bequem bin, und auch niemand die ganzen configure-Parameter auswendig wissen kann, schrieb ich als erstes ein Plaintext Dokument, zum bequemen copy-pasten. Ein Freund postete dann irgendwann die URL in eine Newsgroup, und es kamen die ersten Anfragen ... Dies ermutigte mich dann mehr aus dem Dokument zu machen.

Besonders viele Leute fragten nach einem »offiziellen« HOWTO im SGML Format. Nun, hier ist es...

1.3. Was dieses Dokument bezweckt

Wie bereits erwähnt, niemand kann die alle Konfigurationsoptionen auswendig kennen, welche nötig sind einen Apache mit allen Features zu kompilieren. Dieses Dokument hält die Konfigurationsschritte zum *copy-paste* bereit.

Das zweite Ziel ist es, dass die Leute lernen wie man sich selber zu einem Apache verhilft. Dadurch wird die Abhängigkeit von Distributoren etwas geringer werden.

1.4. Was dieses Dokument nicht ist

Es ist ganz einfach nur ein Dokument, kein Script, welches einem die Arbeit abnimmt. Alle Schritte müssen vom Benutzer selber ausgeführt werden.

1.5. Plattformen

Das ursprüngliche Dokument war für alle wichtigen Un*x Plattformen gedacht, beim aktuellen HOWTO ist das etwas anders, es wird für jede Plattform ein eigenes geben, der Anfang ist Linux.

- Linux (Alle Prozessoren)
- IBM AIX 4.3 und 5.1L
- Sun Solaris 6/7/8
- Hewlett-Packard HP-UX 11
- {Free|Net|Open}-BSD

Wichtige Information für Benutzer welche Linux auf IBM S/390 Mainframes (zSeries) betreiben: PostgreSQL und jserv lassen sich auf diesen Systemen nicht kompilieren. Die übrigen Anleitungen laufen allerdings perfekt

Andere Unix-Plattformen: Ich bin sehr dankbar für einen Gast-Zugang um das ganze auszuprobieren.

Windows: Nein danke, ich bin noch viel zu jung für einen Herzinfarkt :-)

1.6. Copyright Informationen

Dieses Dokument ist Copyright (c) 2000,2001,2002 by Luc de Louw und ist unter den Bedingungen der GNU Free Dokument License veröffentlicht. Dies bedeutet, dass dieser Text sowohl in elektronischer wie auch physikalischem Weg verbreitet werden darf, Solange der Copyright Vermerk nicht entfernt wird. Eine kommerzielle Verbreitung ist erlaubt und erwünscht. Bei Veröffentlichung ist dies dem Autor mitzuteilen.

1.7. Haftungsausschlussklausel

Der Autor lehnt jegliche Haftung für das Resultat der Anwendung dieses Dokumentes ab. Dies schliesst auch Datenverluste infolge von eventuellen Fehlern in diesem Dokument mit ein. Die Anwendung erfolgt auch eigenes Risiko und Gefahr.

Es wird dringend angeraten die aufgeführten Prozeduren erst auch einem Testsystem auszuprobieren um die gewünschte Funktionalität zu prüfen

1.8. Neuste Versionen dieses Dokuments

Die neuste Version ist jeweils auf <http://www.delouw.ch/linux/apache.phtml> zu finden.

Neue Versionen werden auf <http://freshmeat.net/apacompile> angekündigt.

1.9. Credits

An dieser Stelle möchte ich mich bei all den vielen Leuten bedanken welche mir kritische Fragen und Fehlerberichtigungen geschickt haben.

Speziell danken möchte ich den Leuten der TLDP *The Linux Documentation Project* Mailingliste <discuss@linuxdoc.org> welche mich vor allem in SGML Fragen unterstützt haben.

1.10. Feedback

Feedback ist immer herzlich willkommen. Wenn Sie Anregungen und Wünsche betreffend diesem Dokument haben, schreiben Sie mir einfach ein Email an <luc@delouw.ch>

1.11. Übersetzungen

Im Moment ist dieses Dokument in folgenden Sprachen verfügbar:

- [Deutsch](#)
- [Französisch](#)
- [Englisch \(Original\)](#)

Übersetzungen in andere Sprachen sind immer willkommen. Bitte lassen Sie es mich wissen wenn Sie dieses Dokument übersetzt haben, damit hier ein Link eingefügt werden kann.

Ich bitte Sie wenn möglich die englische Version im SGML Quellcode als Grundlage zu verwenden

1.12. Über den Autor

Luc ist 29 Jahre alt, und spielt mit Computern seit etwa 20 Jahren rum. Zur Zeit ist er bei einer IT-Firma in Kloten (Zürich), Schweiz als Unix System Engineer angestellt. Hauptbeschäftigung ist alle möglichen innovativen Sachen zu auf Linux (und anderen Un*xes) zum Laufen zu bringen. Weiter landet alles "unmögliche" bei ihm auf dem Schreibtisch. (Ja, es macht Spass!)

2. Voraussetzungen

2.1. Generell

- flex 2.54
- bison 1.28
- autoconf 2.52
- automake 1.4
- libtool 1.4
- yacc-91.7.30
- freetype2-devel [\[1\]](#)
- re2c [\[2\]](#)

Fortsetzung folgt

2.2. OpenSSL

2.2.1. Was ist OpenSSL

Das OpenSSL Projekt hat ein gemeinschaftliches Ziel, eine robuste, einem kommerziellen Produkt in nichts nachstehende vollständige Secure Sockets Layer (SSL v2/v3) und Transport Layer Security (TLS v1) Implementierung auf OpenSource Basis zu entwickeln. Dazu gehört ebenfalls eine universelle Kryptografische Bibliothek. Das Projekt wird durch eine weltweite Gemeinschaft von Volutären unterhalten welches das Internet für Kommunikation, Planung und Entwicklung benutzt.

OpenSSL basiert auf der exzellenten SSLeay Bibliothek von Eric A. Young und Tim J. Hudson. OpenSSL hat eine Apache-ähnliche Lizenz, welche grundlegend besagt: Freie Benutzung für kommerzielle und nicht-kommerzielle Anwendungen.

--www.openssl.org

Aus meiner Sicht ist es die Basis um einem sicheren Webserver einzurichten. OpenSSL ist die OpenSource Basis für sehr viele Applikationen welche sichere Verschlüsselung gebrauchen (Wie zum Beispiel OpenSSH etc)

2.2.2. Download des Quellcodes

Haupt-Seiten <http://www.openssl.org>

2.2.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf openssl-0.9.6d.tar.gz

cd openssl-0.9.6d
./config
```

```
make
make test
make install
```

2.3. GNU Database System

2.3.1. Was ist gdbm

GNU dbm ist ein Datenbank System das erweiterbare »hashing«. Es arbeitet ähnlich dem standart UNIX dbp Programm.

—www.gnu.org/software/gdbm

Das GNU Datenbank Sytem ist eine sehr wichtige Applikation, und wird von fast allen Distributoren eingesetzt. Es ist standartmässig bei allen Distributionen eingesetzt, welche ich getestet habe.

Ziemlich sicher ist jedoch, dass die die header Dateien nicht standartmässig installiert werden. Diese werden allerdings zwingend benötigt, um Apache mit mod_rewrite zu kompilieren, und werden auch von PHP gebraucht. Bitte installieren Sie die benötigten Dateien von der CD/DVD Ihres Distributors nach (Version kann abweichen).

```
rpm -i gdbm-devel-1.8.0-546
```

Diese Instruktion funktioniert auf SuSE und Redhat Systemen. Andere RPM-basierende Systeme wie Mandrake: bitte bestätigen.

Benutzer von Debian basierenden System können das Paket folgendermassen installieren:

```
apt-get install libgdbmgl-dev
```

2.3.2. Selber kompilieren und installieren

Im kaum anzunehmenden Fall, dass Ihre Distribution gdbm nicht enthält, hier die Instruktionen wie Sie das Programm selber kompilieren.

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

2.4. MySQL

2.4.1. Was ist MySQL

MySQL ist eine sehr schnelle, performante und einfach zu handhabende Datenbank.

Speziell für Webapplikationen die häufige Lesezugriffe tätigen, und relativ seltene Schreibzugriffe benötigen ist MySQL die beste Wahl. Wenn Transaktionen benötigt werden ist PostgreSQL allerdings die bessere Wahl, da Transaktionen bei MySQL noch in den Kinderschuhen stecken.

Im Bezug zu Apache, hier wird das C-API von MySQL gebraucht, um PHP und mod_authmysql zu kompilieren, Siehe [Section 4.3](#)

2.4.2. Download

Hauptserver: <http://www.mysql.com/downloads/>

2.4.3. Kompilieren und Installieren

```
cd /usr/local
tar -xvzf mysql-3.23.51.tar.gz
cd mysql-3.23.51

./configure
--prefix=/usr/local/mysql \
--enable-assembly \
--with-innodb

make
make install

/usr/local/mysql/bin/mysql_install_db
echo /usr/local/mysql/lib/mysql >> /etc/ld.so.conf
ldconfig
```

Um die Sicherheit zu erhöhen, fügen Sie Ihrem System einen MySQL-Benutzer hinzu. Danach muss der Besitzer des Datenverzeichnisses angepasst werden:

```
chown -R mysql /usr/local/mysql/var
```

Wahrscheinlich ist ein automatischer Start von MySQL beim Systemstart gewünscht, dazu kopieren Sie die Datei `/usr/local/mysql/share/mysql/mysql.server` nach `/etc/init.d/` (oder wo immer die rc-scripts auf dem entsprechenden System zu finden sind). Auch müssen die entsprechenden symbolischen Links in den entsprechenden *Runleveldirectories* erstellt werden.

```
cp /usr/local/mysql/share/mysql/mysql.server /etc/init.d/

ln -s /etc/init.d/mysql.server /etc/init.d/rc3.d/S20mysql
ln -s /etc/init.d/mysql.server /etc/init.d/rc3.d/K20mysql
```

Dieses Beispiel funktioniert bei SuSE ab Version 7.0

2.4.4. Sicherheit erhöhen

Dieser Teil ist nicht unbedingt nötig aber dringend empfohlen.

Wenn nicht anders benötigt sollte MySQL nur auf das loopback-interface 127.0.0.1 gebunden werden. Dies stellt sicher, dass niemand über das Netzwerk auf Ihren MySQL Server zugreifen kann. Dies macht natürlich nur Sinn, wenn MySQL auf dem gleichen Rechner wie der Webserver läuft.

Editieren Sie `/etc/init.d/mysql.server` etwa auf Linie 107 wie folgt:

Original Linie:

```
$bindir/safe_mysqld --datadir=$datadir --pid-file=$pid_file&
```

Neu:

```
$bindir/safe_mysqld --datadir=$datadir --pid-file=$pid_file \
--bind-address=127.0.0.1& ❶
```



Hier wird die Adresse an welche MySQL gebunden werden soll definiert.

oder alternativ kann die ganze Netzwerkfunktionalität abgeschaltet werden:

```
$bindir/safe_mysqld --datadir=$datadir --pid-file=$pid_file \  
--skip-networking &
```

2.5. Kompilieren von mm

2.5.1. Was ist mm

Die mm Bibliothek ist eine 2-Schichten Abstraktions-Bibliothek, welche die Verwendung von Shared Memory zwischen Prozessen auf Un*x Plattformen erleichtert. Die erste Schicht versteckt die Plattformabhängigkeit der verschiedenen Implementierungen (Handling und Locking) Die zweite Schicht bietet ein high-level malloc(3)-style API für eine bequeme und gut dokumentierte Arbeit innerhalb der Shared Memory Segmenten.

--www.engelschall.com

MM ist es eine allgemeine Bibliothek für shm (Shared Memmory) handling welche hier in diesem Zusammenhang für mod_ssl und PHP gebraucht wird.

2.5.2. Download

HauptServer : <http://www.engelschall.com/sw/mm/mm-1.1.3.tar.gz>

2.5.3. Kompilieren und Installieren

```
cd /usr/local  
tar -xvzf mm-1.1.3.tar.gz  
cd mm-1.1.3  
./configure  
make  
make test  
make install
```

3. Downloaden, kompilieren und installieren von Apache und seinen wichtigsten Modulen

3.1. Download und auspacken des Apache Quellcodes

3.1.1. Was ist Apache

Das Apache-Projekt hat zum Ziel in einer gemeinschaftlichen Entwicklung einen robusten, einem kommerziellen Produkt in nichts nachstehenden und mit freiem Quellcode verfügbarer http (Web)Server zu bauen. Das Projekt wird unterhalten von einer weltweiten Gruppe aus Volutären, welche das Internet als Medium für Kommunikation, Planung und Entwicklung benutzen. Diese Volutäre sind bekannt als

"The Apache Group". Zusätzlich hunderte von Benutzern senden Ideen, Quellcode und Dokumentationen für das Projekt.

--www.apache.org

Apache ist ganz einfach der beste Webserver, sehr flexibel, ein den jeweiligen Ansprüchen anzupassender Webserver. Auch ist er E-X-T-R-E-M stabil. Ich konnte noch nie einen Absturz in einer Produktions-Umgebung miterleben.

3.1.2. Download des Quellcodes

Hauptserver: <http://www.apache.org/dist/httpd/>

```
cd /usr/local/  
tar -xvzf apache_1.3.26.tar.gz
```

3.1.3. Patch für Grosse Serverumgebungen

Download von: http://www.delouw.ch/linux/apache-patch_HARD_SERVER_LIMIT.txt

```
--- httpd.h      Thu Mar 21 18:07:34 2002  
+++ httpd.h-new Sun Apr  7 13:34:11 2002  
@@ -320,7 +320,7 @@  
 #elif defined(NETWARE)  
 #define HARD_SERVER_LIMIT 2048  
 #else  
-#define HARD_SERVER_LIMIT 256  
+#define HARD_SERVER_LIMIT 512  
 #endif  
 #endif
```

Dieser Patch erhöht die maximale Anzahl laufender Prozesse, und damit die Anzahl maximal gleichzeitig zugreifender Benutzer auf 512. Man kann diese Zahl durchaus weiter erhöhen, man muss dabei allerdings an die Kernel-Limits und an `/etc/security/limits.conf` denken.

Sicherstellung von genügend Ressourcen

Mit falschen Einstellungen in `/etc/security/limits.conf` und/oder falsch konfiguriertem Kernel kann man sich einen »*Self-Denial-Of-Service-Attack*« bescheren. Stellen Sie sicher, dass immer genügend Tasks for root exklusiv zur Verfügung stehen. Bei Linux Kernel 2.2.x müssen ebenfalls die Kernel Quellcodes angepasst werden

Um den patch anzuwenden:

```
cd /usr/local/apache_1.3.26/src/include  
patch -p0 < apache-patch_HARD_SERVER_LIMIT.txt
```

3.2. mod_ssl

3.2.1. Was ist mod_ssl

Dieses Module gibt die Möglichkeit starke Verschlüsselung mit dem Apache 1.3.x einzusetzen. Das verwendete Protokoll ist Secure Socket Layer (SSL v2/v3) und Transport Layer Security (TLS v1). Dafür greift mod_ssl auf die OpenSource Bibliothek OpenSSL, welche auch SSLey von Eric A. Young und Tim J. Hudson basiert zurück.

--www.modssl.org

Dieses Modul wird benötigt um dem Apache die Möglichkeit zu geben, Daten verschlüsselt über das Internet zum Client zu transportieren. Es ist ein Patch für den Apache-Quellcode und erweitert dessen API(Application Programming Interface). Das Resultat nennt sich EAPI (Extended Application Programming Interface).

Falls mod_ssl eingesetzt wird, ist es sehr wichtig, alle anderen Module mit dem Compiler-Flag `-DEAPI` zu kompilieren, sonst könnte es sein, das Apache einfach abstürzt, oder gar nicht erst läuft. Fast alle Module erkennen dies allerdings selbstständig, eine Ausnahme dabei ist mod_jserv und mod_jk

3.2.2. Download des Quellcodes

Hauptserver:<http://www.modssl.org>

3.2.3. Anwenden des Patches

```
cd /usr/local/
tar -xvzf mod_ssl-2.8.10-1.3.26.tar.gz
cd mod_ssl-2.8.10-1.3.26/
./configure \
--with-apache=../apache_1.3.26 \
```

3.3. mod_perl

3.3.1. Was ist mod_perl

Mit mod_perl ist es möglich, Apache-Module komplett in Perl zu entwickeln. Da der Interpreter in den Apache integriert wird, muss im Gegensatz zu CGI nicht bei jedem Request ein neuer Prozess gestartet werden. Dies spart sehr viel Overhead, und macht die Applikationen schneller.

--perl.apache.org

Mod_perl ist eine Art Ersatz von CGIs. CGI's starten üblicherweise bei jedem Request einen neuen Prozess, welches die Performance natürlich in Mitleidenschaft zieht. Mod_perl verhindert genau dies.

3.3.2. Download des Quellcodes

Hauptserver: <http://www.apache.org/dist/perl>

3.3.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf mod_perl-1.27.tar.gz
```

```
cd mod_perl-1.27

perl Makefile.PL \
EVERYTHING=1 \
APACHE_SRC=../apache_1.3.26/src \
USE_APACI=1 \
PREP_HTTPD=1 \
DO_HTTPD=1

make
make install
```

Kompilieren Sie mod_perl nicht als DSO (Dynamic Shared Object)

Mod_perl kann nicht als DSO kompiliert werden

Kompilieren Sie mod_perl nicht als DSO (Dynamic Shared Object)! Aufgrund von Meldungen soll Apache nicht mehr lauffähig sein.

3.4. Konfigurieren und Installieren von Apache

Nun sind die beiden statischen Module mod_ssl und mod_perl dem Apache-Quellcode zugefügt worden, der Apache ist jetzt bereit zur Konfiguration.

3.4.1. Kompilieren und installieren

```
EAPI_MM="/usr/local/mm-1.1.3" SSL_BASE="/usr/local/ssl" \
./configure \
--enable-module=unique_id \
--enable-module=rewrite \
--enable-module=speling \
--enable-module=expires \
--enable-module=info \
--enable-module=log_agent \
--enable-module=log_referer \
--enable-module=usertrack \
--enable-module=proxy \
--enable-module=userdir \
--enable-module=so \
--enable-shared=ssl \
--enable-module=ssl \
--activate-module=src/modules/perl/libperl.a \
--enable-module=perl

make
make install
```

3.4.2. Erstellen eines selbst signierten SSL-Zertifikates

```
cd /usr/local/ssl/bin

./openssl req -new > new.cert.csr
./openssl rsa -in privkey.pem -out new.cert.key
./openssl x509 -in new.cert.csr \
-out new.cert.cert -req -signkey new.cert.key \
-days 999

cp new.cert.key /usr/local/apache/conf/ssl.key/server.key
cp new.cert.cert /usr/local/apache/conf/ssl.crt/server.crt
```

 **Common name**

OpenSSL fragt nach verschiedenen Dingen. Ein oft gemachter Fehler ist das Missverstehen von »*common name*« Hier ist der »*Fully Qualified Hostname*« gemeint, also zum Beispiel www.foo.org

4. Zusätzliche Module

4.1. mod_dav

4.1.1. Was ist mod_dav

mod_dav ist ein Apache Module das DAV-Funktionalität gemäss RFC 2518 zur Verfügung stellt. Es ist ein Opensource Module, freigegeben unter einer Apache-ähnlichen Lizenz.

--www.webdav.org

DAV heisst: *Distributed authoring and Versioning* oder in Deutsch: »Verteilte Versionenbereinigte Verwaltung«. Es erlaubt eine Webseiten ähnlich wie im lokalen Dateisystem zu verwalten. Es ist ebenfalls als Ersatz für FTP-Transfers gedacht. DAV ist ein anerkannter Standard.

DAV wird von allen wichtigen Web-Entwicklungs-Werkzeugen unterstützt. (Inkl. M\$ Frontpage)

4.1.2. Download des Quellcodes

Hauptserver: http://www.webdav.org/mod_dav/

4.1.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf mod_dav-1.0.3-1.3.6.tar.gz
cd mod_dav-1.0.3-1.3.6
./configure --with-apxs=/usr/local/apache/bin/apxs
make
make install
```

Verwirrender Dateiname

Der Dateiname `mod_dav-1.0.3-1.3.6` suggeriert, dass das Module nur mit Apache 1.3.6 läuft, soll aber heissen, es läuft mit Apache $\geq 1.3.6$

4.2. auth_ldap

4.2.1. Was ist auth_ldap

auth_ldap ist ein LDAP Authentifizierungs-Modul für Apache, dem am meisten eingesetzten Webserver. auth_ldap hat eine exzellente Performance, und unterstützt Apache sowohl unter Unix als auch Windows NT. auth_ldap unterstützt ebenfalls LDAP over SSL, und ein Modus welcher Frontpage Clients erlaubt ihre Web Permissions mit LDAP zu benutzen.

--www.rudedog.org

LDAP (*Lightweight Directory Access Protocol*) ist ein wunderbarer Weg Login-Facilities zu konsolidieren, das heisst den Benutzern in einer Firma ein einheitlicher Benutzername/Passwort zu geben, welches auf

verschiedenen Diensten wie FTP, Windows Login und so weiter benutzt werden kann. LDAP ist ein offener Standard welcher sehr breit unterstützt wird.

4.2.2. Download des Quellcodes

Hauptserver: http://www.rudedog.org/auth_ldap/

4.2.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf auth_ldap-1.6.0.tar.gz
cd auth_ldap-1.6.0
./configure --with-apxs=/usr/local/apache/bin/apxs \
--with-sdk=openldap
make
make install
```

4.3. mod_auth_mysql

4.3.1. Was ist mod_auth_mysql

mod_auth_mysql ist ein http-Basic Authentication Modul. Es erlaubt Benutzerdaten bequem in einer MySQL-Datenbank abzuspeichern und zu unterhalten.

4.3.2. Download des Quellcodes

Hauptserver: ftp://ftp.kciLink.com/pub/mod_auth_mysql.c.gz

4.3.3. Kompilieren und installieren

```
/usr/local/apache/bin/apxs \
-c -I/usr/local/mysql/include \
-L/usr/local/mysql/lib/mysql \
-lmysqlclient -lm mod_auth_mysql.c
cp mod_auth_mysql.so /usr/local/apache/libexec/
```

Folgendes muss in httpd.conf zugefügt werden:

```
LoadModule mysql_auth_module libexec/mod_auth_mysql.so
```

Und dort wo auch die anderen module geladen werden:

```
AddModule mod_auth_mysql.c
```

Es ist darauf zu achten, das die MySQL Libs and Includes richtig angegeben wurden!

Library Pfad

Es ist sicherzustellen, das der Pfad /usr/local/mysql/lib/mysql in /etc/ld.so.conf steht bevor das Modul kompiliert wird

4.3.4. Beispiel Konfiguration

Example 1. /usr/local/apache/conf/httpd.conf

```
<location /manual/>
  AuthType Basic
  AuthUserfile /dev/null
  AuthName Testing
  AuthGroupFile /dev/null
  AuthMySQLHost localhost
  AuthMySQLCryptedPasswords Off
  AuthMySQLUser root
  AuthMySQLDB users
  AuthMySQLUserTable user_info
  <Limit GET POST>
    require valid-user
  </limit>
</location>
```

4.3.4.1. Script um die MySQL-Datenbank zu erstellen

Zuerst muss die Datei `authmysql.sql` mit folgendem Inhalt erstellt werden:

```
create database http_users;
connect http_users;

CREATE TABLE user_info (
  user_name CHAR(30) NOT NULL,
  user_passwd CHAR(20) NOT NULL,
  user_group CHAR(10),
  PRIMARY KEY (user_name);
```

Dannach kann MySQL angewiesen werden, dies Datei zu verarbeiten

```
mysql < authmysql.sql
```

4.4. mod_dynvhost

4.4.1. Was ist mod_dynvhost

`mod_dynvhost` ist ein kleines Modul welches erlaubt neue VirtualHosts zu definieren und in Betrieb zu nehmen, ohne das der Apache neu gestartet werden muss. Dazu muss nur ein Ordner erstellt werden, fertig.

Ist eine gute Lösung für Massen-Virtual-hosting.

4.4.2. Download des Quellcodes

Hauptserver: http://funkcity.com/0101/projects/dynvhost/mod_dynvhost.tar.gz

4.4.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf mod_dynvhost.tar.gz
```



```
cd dynvhost/
/usr/local/apache/bin/apxs -i -a -c mod_dynvhost.c
```



Überprüfung von httpd.conf

Ein Blick in httpd.conf ob mod_dynvhost.so beim Starten geladen wird ist sinnvoll:

```
LoadModule dynvhost_module libexec/mod_dynvhost.so
```

4.4.4. Beispielkonfiguration

Example 2. /usr/local/apache/conf/httpd.conf

```
<DynamicVirtualHost /usr/local/apache/htdocs/vhosts/>
  HomeDir /
</DynamicVirtualHost>
```

Nun muss für jeden gewünschten Virtualhost in /usr/local/apache/htdocs/vhosts/ ein Subverzeichnis erstellt werden

Zum Beispiel

```
/usr/local/apache/htdocs/vhosts/foo.bar.org
```

Der Webserver muss NICHT neu gestartet werden, die vhosts sind sofort aktiv.

4.5. mod_roaming

4.5.1. Was ist mod_roaming

Mittels mod_roaming ist es möglich den Apache Webserver als Netscape Roaming Access Server zu benutzen. Dies erlaubt es Netscape 4.x Einstellungen, Bookmarks, Adressbücher, Cookies usw. zentral zu speichern und an verschiedenen Orten zu benutzen.

—www.klomp.org/mod_roaming/

Das ist wirklich cool. Leider funktioniert es nicht über Proxy-Verbindungen. Auch andere Browser wie Netscape 6.1, Mozilla oder Opera haben diese Möglichkeit nicht vorgesehen.

4.5.2. Download des Quellcodes

Hauptserver: http://www.klomp.org/mod_roaming/

4.5.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf mod_roaming-1.0.2.tar.gz
cd mod_roaming-1.0.2
/usr/local/apache/bin/apxs -i -a -c mod_roaming.c
```



Überprüfung von `httpd.conf`

Überprüfung von `httpd.conf` ob `mod_roaming` beim Starten geladen wird:

```
LoadModule roaming_module libexec/mod_roaming.so
```

4.5.4. Beispielkonfiguration

Example 3. `/usr/local/apache/conf/httpd.conf`

```
RoamingAlias /roaming /usr/local/apache/roaming
<Directory /usr/local/apache/roaming>
  AuthUserFile /usr/local/apache/conf/roaming-htpasswd
  AuthType Basic
  AuthName "Roaming Access"
  <Limit GET PUT MOVE DELETE>
    require valid-user
  </Limit>
</Directory>
```

5. Komprimierte Auslieferung

Es sind zwei Module zur Komprimierung der Auslieferung verfügbar: `mod_gzip` und `mod_gunzip`. Diese verwenden verschiedene Methoden um dieses Ziel zu erreichen: Reduktion der verwendeten Bandbreite.

`mod_gunzip` erwartet komprimierte Dateien im Dateisystem, und dekomprimiert diese, falls der Browser keine komprimierten Daten verarbeiten kann. Der Vorteil ist dass nur wenig CPU Ressourcen gebraucht werden weil fast alle Browser in der Lage sind komprimierte Dateien zu verarbeiten. Auf der anderen Seite werden heutzutage der grösste Teil der Webseiten dynamisch erstellt, zum Beispiel mit PHP, und diese Inhalte werden dann unkomprimiert ausgeliefert.

`mod_gzip` braucht keine komprimierten Dateien auf dem System, alle zuvor definierten Inhalte werden automatisch komprimiert bevor diese ausgeliefert werden. Der Vorteil davon ist, dass auch dynamisch erzeugte Inhalte komprimiert werden. Auf der anderen Seite wird recht viel CPU Ressourcen verbraucht, weil dann bei jeder Anfrage das Ergebnis on-the-fly komprimiert wird. `mod_gzip` kann allerdings auch so konfiguriert werden, dass bereits vorkomprimierte Inhalte direkt ausgeliefert werden können.

Schlussfolgerung: Sie müssen sorgfältig abwägen welche der beiden Methoden die Richtige ist für Ihre Anwendung und Ihren Server. Wenn für Sie CPU-Ressourcen kein Thema sind, dann wird `mod_gzip` die richtige Wahl. Wenn Sie auf sehr schnelle Antwortzeiten angewiesen sind, ist `mod_gunzip` die bessere Wahl.

Eine gute Seite (in Englisch), welche Ihnen weiterhelfen kann bei Ihrer Entscheidung ist Martin Kiff's Dokument über `mod_gunzip` <http://www.innerjoin.org/apache-compression/howto.html>

5.1. `mod_gzip`

5.1.1. Download

Origin-Site: http://www.ehyperspace.com/htmlonly/products/mod_gzip.html

5.1.2. Kompilieren und installieren

```
/usr/local/apache/bin/apxs -i -a -c -lz mod_gzip.c
```

5.1.3. Beispielkonfiguration

Fügen sie folgendes in `/usr/local/apache/conf/httpd.conf` ein:

Example 4. `/usr/local/apache/conf/httpd.conf`

```
mod_gzip_on                Yes
mod_gzip_can_negotiate     Yes
mod_gzip_dechunk           Yes
mod_gzip_minimum_file_size 600
mod_gzip_maximum_file_size 0
mod_gzip_maximum_inmem_size 100000
mod_gzip_keep_workfiles    No
mod_gzip_temp_dir          /usr/local/apache/gzip
mod_gzip_item_include      file \.html$
mod_gzip_item_include      file \.txt$
mod_gzip_item_include      file \.jsp$
mod_gzip_item_include      file \.php$
mod_gzip_item_include      file \.pl$
mod_gzip_item_include      mime ^text/.
mod_gzip_item_include      mime ^application/x-httpd-php
mod_gzip_item_include      mime ^httpd/unix-directory$
mod_gzip_item_include      handler ^perl-script$
mod_gzip_item_include      handler ^server-status$
mod_gzip_item_include      handler ^server-info$
mod_gzip_item_exclude     file \.css$
mod_gzip_item_exclude     file \.js$
mod_gzip_item_exclude     mime ^image/.*
```

Wenn Sie die Resultate der Kompression in der Logdatei speichern wollen, kann dies mit einer Änderung in der »LogFormat« Direktive in `/usr/local/apache/conf/httpd.conf` angepasst werden

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\""
mod_gzip: %{mod_gzip_compression_ratio}npt." combined
```

5.2. mod_gunzip

5.2.1. Download

Origin-Site: http://www.oldach.net/mod_gunzip.tar.gz

5.2.2. Kompilieren und installieren

```
tar -xvzf mod_gunzip.tar.gz
cd mod_gunzip-2

/usr/local/apache/bin/apxs -i -a -c -lz mod_gunzip.c
```

5.2.3. Beispielkonfiguration

Fügen Sie folgendes in `/usr/local/apache/conf/httpd.conf` ein:

```
AddType text/html .htmz
AddHandler send-gunzipped .htmz
```

Sie können nun mit **gzip** Ihre Html-Dateien komprimieren und umbenennen:

```
gzip index.html
mv index.html.gz index.htmz
```

Natürlich müssen alle Links auf den Seiten entsprechend geändert werden, also zum Beispiel: `Eine Seite`

6. mod_php seine Voraussetzungen

6.1. Was ist mod_php

PHP ist eine serverseitige, plattformübergreifende, HTML-embedded Scriptsprache.

Am Anfang war es ein einfaches Gästebuch-Verarbeitungsprogramm, und es wuchs und wuchs. Seit der Version 3 ist PHP eine mächtige Web-Entwicklungs-Sprache

--www.php.net

Seit Version 4 ist PHP fähig und robust genug für unternehmenskritische Applikationen. PHP ist mächtig, und unterstützt die meisten wichtigen Datenbanken von Hause aus, oder via ODBC. PHP auf Apache ist auf derselben Hardware um einiges schneller als IIS mit ASP aus dem Hause Microsoft.

Da sich die Syntax stark an Java, C und Perl orientiert kann man PHP sehr schnell erlernen.

Es existieren Erweiterungen wie APC (Alternative PHP Cache) welche die Geschwindigkeit um weitere 50-400 % erhöhen (Kommt auf die Anwendung an)

6.2. Voraussetzungen

Abhängig von den gewünschten Fähigkeiten welche PHP haben soll, muss erst noch einiges an Software installiert werden. MySQL ist bereits installiert um mod_auth_mysql kompilieren zu können.

6.2.1. IMAP client

6.2.1.1. Was ist IMAP

IMAP heisst »Internet Mail Application Protocol« und ist ein Protokoll als Nachfolger von POP3. Es erlaubt Emails in verschiedenen Verzeichnissen auf dem Server aufzubewahren (welcher hoffentlich eine regelmässige Datensicherung erfährt – Nie mehr verlorene Mails weil die lokale Festplatte zerstört wurde).

6.2.2. Download des Quellcodes

Hauptserver: <http://www.washington.edu/imap/>

6.2.3. Kompilieren und installieren

```
cd /usr/local
tar -xvfz imap.tar.Z
cd imap
```

```
make slx SSLTYPE=unix ❶
```



Hier wird SSLTYPE=unix angegeben, falls SSL Unterstützung benötigt wird.

❶ Dateiname zum downloaden

imap.tar.Z ist ein symbolischer Link zur letzten verfügbaren Version, also zum Beispiel
imap-2001a.tar.Z

6.2.4. PostgreSQL

6.2.4.1. Was ist PostgreSQL

PostgreSQL ist eine sehr mächtige Datenbank mit langer Geschichte

PostgreSQL ist wie MySQL sehr gut geeignet für Webapplikationen, aber aus meiner Sicht weniger einfach zu handhaben.

Wenn die Applikation sehr viel Schreibzugriffe tätigt, oder auf Transaktionen angewiesen ist, dann ist PostgreSQL eine gute Wahl.

6.2.4.2. Download des Quellcodes

Hauptserver: <http://www.postgresql.org>

6.2.4.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf postgresql-7.2.1.tar.gz
cd postgresql-7.2.1
./configure \
--with-perl \
--with-odbc \
make
make install
echo /usr/local/pgsql/lib >> /etc/ld.so.conf
ldconfig
```

6.2.5. PDFlib

6.2.5.1. Was ist PDFlib

PDFlib: Eine Bibliothek welche es erlaubt PDF dynamisch zu erzeugen. PDFlib ist die erste Wahl wenn PDF dynamisch auf dem Server erzeugt werden soll. PDF konvertiert Text und Grafiken, oder implementiert PDF in eigenentwickelten Produkten.

—www.pdflib.com

PDFlib ist eine kommerzielle Software, die Prüfung ob eine Lizenz benötigt wird oder nicht ist dringend anzuraten!

6.2.5.2. Download des Quellcodes

Hauptserver: <http://www.pdflib.com/pdflib/download/pdflib-4.0.3.tar.gz>

6.2.5.3. Kompilieren und installieren

```
cd /usr/local/  
tar -xvzf pdflib-4.0.3.tar.gz  
  
cd pdflib-4.0.3  
  
./configure --enable-shared-pdflib --enable-cxx  
  
make  
make install  
  
ldconfig
```

6.2.6. gettext

6.2.6.1. Was ist gettext

gettext ist eine Bibliothek welche für die Internationalisierung von Applikationen verwendet werden kann.

6.2.6.2. Download des Quellcodes

Hauptserver: <ftp://ftp.gnu.org/gnu/gettext>

6.2.6.3. Kompilieren und installieren

```
cd /usr/local  
  
tar -xvzf gettext-0.11.2.tar.gz  
  
cd gettext-0.11.2  
  
./configure  
  
make  
make check  
make install
```

6.2.7. zlib

6.2.7.1. Was is zlib

zlib ist eine Bibliothek welche eine verlustfreie Kompression ermöglicht. zlib läuft auf praktisch jeder Hardware und Betriebssystem.

6.2.7.2. Download des Quellcodes

Origin-Site: <ftp://ftp.info-zip.org/pub/infozip/zlib/zlib.tar.gz>

6.2.7.3. Kompilieren und installieren

```
cd /usr/local  
  
tar -xvzf zlib.tar.gz
```

```
cd zlib-1.1.4/

./configure

make
make check
make install

ldconfig
```

6.3. Kompilieren und installieren von PHP4

```
cd /usr/local

tar -xvzf php-4.2.1.tar.gz

cd php-4.2.1

./configure \
--with-apxs=/usr/local/apache/bin/apxs \
--with-mysql=/usr/local/mysql \
--with-pgsql=/usr/local/pgsql \
--enable-track-vars \
--with-openssl=/usr/local/ssl \
--with-ldap=/usr/local/imap-2001a \
--with-gd --with-ldap \
--enable-ftp \
--enable-sysvsem \
--enable-sysvshm \
--enable-sockets \
--with-pdflib=/usr/local \
--with-gettext \
--with-mm=/usr/local/mm-1.1.3 \
--with-jpeg-dir=/usr/lib \
--with-zlib-dir=/usr/local \

make
make install
```

Kopieren Sie die mitgelieferte Beispiel `php.ini-dist` nach `/usr/local/lib/php.ini`

```
cp /usr/local/php-4.2.1/php.ini-dist /usr/local/lib/php.ini
```

Entfernen Sie die `#` an Anfang der folgenden Linien in `/usr/local/apache/conf/httpd.conf`

Die derzeitige Apache Version enthält die folgenden Zeilen nicht, und müssen eingefügt werden.

```
AddType application/x-httpd-php .php
AddType application/x-httpd-php .phtml
AddType application/x-httpd-php .php3

AddType application/x-httpd-php-source .phps ❶
```



Diese Zeile brauchen Sie nur einzufügen, falls Sie PHP Quellcodes im Browser anzeigen lassen wollen. Diese Dateien müssen die Endung `.phps` besitzen

Der Apache muss nun neu gestartet werden:

```
/usr/local/apache/bin/apachectl restart
```

7. PHP Erweiterungen

Es existieren verschiedene Erweiterungen (*extensions*) für PHP, welche in der `php.ini` eingetragen werden können.

7.1. APC (Alternative PHP-Cache)

7.1.1. Was ist APC

APC heisst »Alternative PHP Cache«. APC wurde erdacht um einen Weg zu Performancesteigerung auf hoch belasteten Servern zu erzielen. Dies wird erreicht durch das Zwischenspeichern der `php`-Seiten in ihrem kompilierten Zustand, so das der Overhead des Parsens und Kompilieren nahezu eliminiert wird. Es existieren kommerzielle Produkte mit derselben Funktionalität, aber diese sind weder OpenSource noch frei verfügbar.

--www.apc.communityconnect.com/

Ich machte einige Performancetests mit `apc` und es war wirklich eine Überraschung. Eine PHP-Webpage mit verschachtelten MySQL-Abfragen (total 10 Abfragen) wurde mehr als 50% schneller.

7.1.2. Download des Quellcodes

Hauptserver: <http://apc.communityconnect.com/sources/apc-cvs.tar.gz>

7.1.3. Kompilieren und installieren

```
cd /usr/local
tar -xvzf apc-cvs.tar.gz
cd apc
./configure --enable-apc --with-php-config=/usr/local/bin/php-config
make
make install
cp modules/php_apc.so /usr/local/lib/php/extensions
echo 'zend_extension="/usr/local/lib/php/extensions/php_apc.so"' >> /usr/local/lib/php.ini
echo apc.mode = shm >> apc.mode = shm
```

Apache muss nun neu gestartet werden. Durch das Erstellen eines PHP-Dokumentes mit folgendem Inhalt kann die Funktion des APC überprüft werden:

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

7.2. Zend-Optimizer

7.2.1. Was ist Zend-optimizer

Zitat www.zend.com

Der Zend-Optimizer optimiert den Code weitergehend als die Zend-Engine, und erreicht durch dies eine Performancesteigerung.

--www.zend.com

Bei eigenen Performancetests habe ich einen Performance-Verlust von rund 5% gesehen? Ich stelle die Frage in den Raum, für was diese close source-Applikation den nützlich sein soll?

Auf jeden Fall sind eigene Test angebracht um zu sehen in welchen spezifischen Fällen APC oder Zend-Optimizer das bessere Ergebnis produziert.

Was leider nicht funktioniert, ist APC und Zend-Optimizer zusammen lauffenzulassen, Apache startet erst gar nicht

7.2.2. Download des Programms

Hauptserver: https://www.zend.com/store/free_download.php?pid=13

Um das Programm downloaden zu können muss man sich erst bei zend.com registrieren, meine Erfahrung war, das kein SPAM verschickt wurde aufgrund dieser Registration.

7.2.3. Installieren

Das das Produkt closedsource ist, gibts nichts zu kompilieren.

```
cd /usr/local
tar -xvzf ZendOptimizer-1.3.1-Linux_glibc21-i386.tar.gz
cd ZendOptimizer-1.3.1-Linux_glibc21-i386
./install.sh
```

Das Installations-Script sollte selbsterklärend sein. Wenn Sie Apache und PHP so kompiliert haben, wie in diesem Dokument beschrieben, genügt es bei den Fragen **ENTER** zu drücken.

8. Serverseitiges Java

8.1. mod_jserv

8.1.1. Was ist mod_jserv

Apache JServ ist eine 100% pure Java Servlet Engine welche voll den Anforderungen von JavaSofts Java Servlet API 2.0 entspricht.

--java.apache.org

mod_jserv ist erstaunlich einfach zu handhaben, was für Java-Applikationen an ein Wunder grenzt.

Unglücklicherweise wird mod_jserv nicht mehr länger aktiv weiterentwickelt, sondern nur noch allfällige schwerwiegende Fehler repariert.

An Stelle von mod_jserv ist Tomcat getreten, welches ungleich schwieriger zu handhaben und installieren ist. Auch JSDK 2.0 ist nicht aktuell. Sobald ich genügend Erfahrungen mit Tomcat habe, wird das HOWTO entsprechend erweitert.

8.1.2. Voraussetzungen

8.1.2.1. JSDK 2.0

8.1.2.1.1. Was ist JSDK

Es ist das API von Sun Microsystems um Servlets entwickeln zu können.

8.1.2.1.2. Download

Hauptserver: <http://java.sun.com/products/servlet/archive.html>

8.1.2.1.3. Installieren

JSDK ist ein Java-Binary, einfach installieren genügt

Verwirrender Dateiname

Der Dateiname `jsdk20-solaris2-sparc.tar.Z` suggeriert das die Datei nur für die SPARC Plattform ist, funktioniert aber auf allen anderen Plattformen ebenfalls.

```
cd /usr/local/  
uncompress jsdk20-solaris2-sparc.tar.Z  
tar -xvf jsdk20-solaris2-sparc.tar
```

8.1.3. Download des Quellcodes

Hauptserver: <http://java.apache.org/jserv/dist/>

8.1.4. Kompilieren und installieren

```
cd /usr/local  
  
tar -xvzf ApacheJServ-1.1.2.tar.gz  
cd ApacheJServ-1.1.2  
  
./configure \  
--prefix=/usr/local/jserv \  
--with-apxs=/usr/local/apache/bin/apxs \  
--with-JSDK=/usr/local/JSDK2.0 \  
--enable-EAPI \  
  
make  
make install
```

Extended API

Hier ist es sehr wichtig das Compilerflag `--enable-EAPI` zu benutzen, falls Apache mit `mod_ssl` kompiliert wurde

8.2. jakarta-tomcat

8.2.1. Was ist Tomcat

Tomcat ist ein *servlet container* der in der offiziellen Referenzimplementierung für die Java Servlet und JavaServer Pages Technologien benutzt wird. Die Java Servlet und

JavaServer Pages Spezifikationen sind von Sun im *Java Community Process* entwickelt.

--jakarta.apache.org

Tomcat ist der Nachfolger von jserv das nicht mehr weiterentwickelt wird. Tomcat unterstützt die aktuellsten jsp und Servlet-API's die Sun definiert hat. Leider is Tomcat sehr aufwändig zu kompilieren, weil es ein total proprietäres Building-System benutzt (Ant). Die Voraussetzungen dazu enthält eine elend lange Liste:

<http://jakarta.apache.org/tomcat/tomcat-4.0-doc/BUILDING.txt> Viel Glück, und Feedback ist willkommen. In der Zwischenzeit wird an dieser Stelle eine Installation der Binär-Distribution beschrieben.

Ich suche Voluntäre welche mit der kompilierung der Quellcodes Erfahrungen haben und die Erfahrungen mitteilen.

8.2.2. Voraussetzungen

8.2.2.1. Java2

8.2.2.1.1. Was ist Java2

Zitat java.sun.com

Kein Zitat, das dieses HOWTO sonst doppelt so lange würde. Hier der Link zu der entsprechenden Seite bei Sun: <http://java.sun.com/j2se/1.3/docs/relnotes/features.html>

8.2.2.2. Download der Binär-Distribution

[http://java.sun.com/j2se/1.3/\[3\]](http://java.sun.com/j2se/1.3/[3]) ,wählen Sie die von Ihnen eingesetzte Plattform und folgen Sie den Instruktionen auf der Webseite.

8.2.2.3. Installation der Binär-Distribution

Dazu muss die Datei ausgeführt werden:

```
chown +x j2sdk-1_3_1_02-linux-i386.bin
./j2sdk-1_3_1_02-linux-i386.bin
```

Nach dem akzeptieren der Lizenz wird ausgepackt, danach kann man das Verzeichnis nach `/usr/lib` verschieben.

8.2.3. Download der Binärdistribution

Hauptserver:

<http://jakarta.apache.org/builds/jakarta-tomcat-4.0/release/v4.0.3/bin/jakarta-tomcat-4.0.3.tar.gz>

8.2.4. Installation der Binär-Distribution

```
cd /usr/local
tar -xvzf jakarta-tomcat-4.0.3.tar.gz
cd jakarta-tomcat-4.0.3
cd bin
rm *.bat
echo export JAVA_HOME=/usr/lib/java/ >> /etc/profile
. /etc/profile
```

Jetzt sollte es möglich sein Tomcat zu starten:

```
/usr/local/apache/jakarta-tomcat-4.0.3/bin/startup.sh
```

Falls der Start erfolgreich war, sollte man jetzt Tomcat erreichen können: <http://localhost:8080/index.html>

Für die weiteren Schritte wie installieren der Servlets und jsp Dateien ist jeder selber verantwortlich...

Umgebungsvariablen

Sollte Tomcat und/oder Ihre Servlets nicht ausgeführt werden können, liegt meinstens ein Problem mit der Umgebungsvariable CLASSPATH vor. Stellen Sie sicher das alle von Ihren Servlets benötigten Klassen in in dieser Variable drin stehen.

9. Weiterführende Informationen

Hier finden Sie Quellen für weiterführende Informationen zum den besprochenen Themen

9.1. Newsgroups

Einige interessante (englische) Newsgroups:

- [alt.apache.configuration](#)
- [comp.infosystems.www.servers.unix](#)
- [alt.comp.lang.php](#)
- [alt.php](#)
- [comp.databases](#)

Daneben gibts es einige Länderspezifische Newsgroups, welche ebenfalls in Sachen Apache helfen können. Ein Beispiel ist: [ch.comp.os.linux](#)

9.2. Mailing Listen

Für einige Themen existieren Mailinglisten, allerdings sind alle welche ich kenne in Englisch. Diesen Mailinglisten sollte man nur in englisch schreiben. Wenn jemand diese Sprache nicht so beherrscht, macht nichts, Rechtschreibfehler werden verziehen.

9.2.1. <users@httpd.apache.org>

Senden Sie ein leeres Email an <users-subscribe@httpd.apache.org> um in der Mailingliste aufgenommen zu werden.

Bevor Sie Fragen an eine Mailingliste schicken konsultieren Sie bitte das entsprechende Archiv. Im Falle von Apache is dieses an folgender Adresse zu finden: <http://marc.theaimsgroup.com/?l=apache-httpd-users>

9.2.2. <modperl@apache.org>

Senden Sie ein Email mit dem Inhalt (nicht Betreff) »subscribe modperl" an <modperl@apache.org>

Das Archiv dieser Liste kann hier gefunden werden: <http://outside.organic.com/mail-archives/modperl/>

9.2.3. <openssl-users@openssl.org>

Senden Sie ein Email mit dem Inhalt (nicht Betreff) »subscribe openssl-users" an <majordomo@openssl.org>

Das Archiv dieser Liste kann hier gefunden werden:
<http://www.mail-archive.com/openssl-users@openssl.org/>

9.2.4. <modssl-users@modssl.org>

Senden Sie ein Email mit dem Inhalt (nicht Betreff) »subscribe modssl-users" an <majordomo@modssl.org>

Das Archiv dieser Liste kann hier gefunden werden:
<http://www.mail-archive.com/modssl-users@modssl.org/>

9.2.5. <mysql@lists.mysql.com>

Senden Sie ein leeres Email an <mysql-subscribe@lists.mysql.com>

Das Archiv dieser Liste kann hier gefunden werden: <http://lists.mysql.com/cgi-ez/ezmlm-cgi/>

9.2.6. <pgsql-general@postgres.org>

Füllen Sie das Formular auf <http://developer.postgresql.org/mailsub.php>

Das Archiv dieser Liste kann hier gefunden werden: <http://archives.postgresql.org/pgsql-general/>

9.2.7. PHP Mailinglisten

Hier gibt es diverse Mailinglisten, einfach das Formular auf folgender Webseite ausfüllen:
<http://www.php.net/mailling-lists.php>

Auf dieser Seite gibts ebenfalls Links zu den entsprechenden Archiven

9.2.8. <apc-cache@lists.communityconnect.com>

Sende ein Email mit dem Inhalt (nicht Betreff) »subscribe« an <apc-cache-request@lists.communityconnect.com>

9.3. HOWTO

Leider sind zu verwandten Thememen keine deutschsprachigen HOWTO's vorhanden. Dies liegt wohl daran, dass die Qualitätsansprüche des »Deutschen Linux HOWTO Projektes« schlichtweg zu hoch sind. Bei diesem Projekt wird auf maximale sprachliche Qualität Wert gelegt. Da selten ein Hacker Germanistik studiert hat, ist die Auswahl entsprechend sehr klein.

Ich spiele mit dem Gedanken, ein alternatives Projekt zu gründen, welches mehr Wert auf den technischen Inhalt legt, mit dem Vorbild des *Linux Documentation Project*

Alleine ist der Arbeitsaufwand allerdings kaum zu bewältigen. Daher müsste ein solches Projekt »Mitreiter« haben, welche die Arbeit gemeinsam erledigen.

Folgende Kenntnisse sind sinnvoll:

- SGML Kenntnisse, speziell Erfahrungen mit den Docbook 3.1 und 4.1 DTD's
- XML Kenntnisse, speziell Erfahrungen mit Docbook 4.1 XML DTD
- CVS Kenntnisse, um einen CVS Server zu pflegen

Web und CVS Server sowie 512kbps Leitung kann ich zumindest vorläufig zur Verfügung stellen.

Viel zu bieten hat ein solches Projekt den Teilnehmern kaum, ausser ein gutes Gefühl, etwas für die Allgemeinheit getan zu haben. Ausserdem macht sich eine Publikation immer gut auf einer Stellenbewerbung :-)

10. FAQ (Häufig gestellte Fragen)

Hier beantworte ich die am häufigsten gestellten Fragen

1. [FAQ](#)

10.1.1. [Frage: Gibt es eine Version dieses Dokumentes für Apache 2.0?](#)

10.1.2. [Frage: Wieso gibt es in diesem Dokument keine Beschreibung für mod_xyz?](#)

10.1.3. [Frage: Meine Kunden erhalten beim Aufruf von <https://myserver.org> eine Fehlermeldung wegen einem ungültigen Zertifikat. Was läuft falsch?](#)

10.1.4. [Frage: Wenn ich eine PHP Seite aufrufe will der Browser die Seite downloaden oder zeigt mir den Quellcode der PHP-Seite an. Was ist hier falsch gelaufen?](#)

10.1.5. [Frage: Was ist mit Anderen Plattformen? ganz oben kann ich lesen es soll eine AIX, HP-UX usw. Version geben?](#)

1. FAQ

10.1.1. Frage: Gibt es eine Version dieses Dokumentes für Apache 2.0?

Antwort: Noch nicht. Das liegt daran, das PHP 4.2.1 das Apache2 API zur Zeit nur experimentell unterstützt. Dadurch ist die Stabilität des ganzen sehr in Frage gestellt. Ausserdem habe ich bei ersten Tests massive Geschwindigkeitsprobleme festgestellt. Apache 2.0 ist ein ganz gewaltiger Fortschritt in Sachen Funktionalität und Geschwindigkeit (ohne PHP). Deshalb wird ganz sicher ein entsprechendes HOWTO geschrieben. Zur Zeit bin ich am Sammeln von Vorschlägen, was von den Benutzern denn so gewünscht wird. Für Vorschläge bitte ein Email senden an: [<luc_at_delouw.ch>](mailto:luc_at_delouw.ch)

10.1.2. Frage: Wieso gibt es in diesem Dokument keine Beschreibung für mod_xyz?

Antwort: Weil ich das entsprechende Modul nicht kenne, und weil niemand danach gefragt hatte. Wenn ich genügend Anfragen für die Unterstützung eines Modules erhalte, und ich das Modul entsprechend getestet habe, werde ich das Dokument entsprechend ergänzen.

10.1.3. Frage: Meine Kunden erhalten beim Aufruf von <https://myserver.org> eine Fehlermeldung wegen einem ungültigen Zertifikat. Was läuft falsch?

Antwort: Das selber erstellte Zertifikat wie beschrieben in diesem Dokument is selbst-zertifiziert. Nun kennen die Browser der Kunden dieses Zertifizierungsstelle (*Certification Authority*, CA) nicht. Im Falle eines Einsatzes in einem Intranet ist der einfachste Weg, das Zertifikat auf den Browsern zu installieren. Für kommerzielle Webserver im Internet für Zahlungsverkehr oder ähnliches ist aber dringend empfohlen das Zertifikat von entsprechender Stelle autorisieren zu lassen. Ein solcher Vorgang kostet leider einige hundert US Dollars, zum Beispiel bei <http://www.verisign.com>.

10.1.4. Frage: Wenn ich eine PHP Seite aufrufe will der Browser die Seite downloaden oder zeigt mir den Quellcode der PHP-Seite an. Was ist hier falsch gelaufen?

Antwort: Sie haben vergessen dem Apache mitzuteilen was mit den Dateien anzufangen ist. So sind die Dateien eben nicht von der Zend Engine ausgeführt worden. Um dies zu berichtigen lesen Sie bitte [Section 6.3](#) durch.

10.1.5. Frage: Was ist mit Anderen Plattformen? ganz oben kann ich lesen es soll eine AIX, HP-UX usw. Version geben?

Antwort: Nun, sobald ich entsprechend Zeit und Maschinen zu Verfügung habe werde ich mich dem Thema wieder annehmen. Zur Zeit bin ich in der Lage sagen zu können, das Solaris fast 1:1 übernommen werden kann, und das es auf AIX unmöglich ist ohne brutale Quellcode Hackery PHP kompilieren zu können. Auf HP-UX habe ich PHP zum laufen gebracht, aber nicht als DSO, sondern nur statisch (php configure --with-apache=/. anstelle --with-apxs=/.)

Notes

- [1] Dieses RPM enthält die header Dateien welche für PHP gebraucht werden
- [2] Wird nur benötigt, wenn PHP von den Quellen aus dem CVS tree kompiliert werden soll
- [3] Es existiert auch die Version 1.4.0 von Java. Wenn Sie keine Bedenken bezüglich Stabilität haben, können Sie auch diese Version downloaden