

I Zusammenspiel des Apache-Servers und Perl

Um⁴⁷ den Apache-Server als Common Gateway Interface zu nutzen, muss das Modul *mod_cgi* installiert sein. Dieses Modul wird bei der Standardinstallation mit dazugenommen. Der Apache-Server wurde von mir in das Verzeichnis */usr/local/apache* installiert. Bei allen nachfolgenden Beispielen wird dieses Verzeichnis als Installationsverzeichnis angenommen. In der Konfigurationsdatei des Apache-Servers */usr/local/apache/conf/httpd.conf* befindet sich deshalb die Zeile:

```
ServerRoot "/usr/local/apache"
```

Das Verzeichnis, in dem die CGI-Skripten stehen, könnte in der Konfigurationsdatei *httpd.conf*, wie folgt eingebracht worden sein:

```
#
# ScriptAlias: This controls which directories contain server
# scripts.
# ScriptAliases are essentially the same as Aliases, except that
# documents in the realname directory are treated as applications
# and run by the server when requested rather than as documents
# sent to the client.
# The same rules about trailing "/" apply to ScriptAlias directives
# as to Alias.
#
ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"

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

Damit wird zum Beispiel das CGI-Skript */usr/local/apache/cgi-bin/datei1* durch die Eingabe von *http://hostname.domain/cgi-bin/datei1* im Browser gestartet. Oder es wird durch eine POST- oder GET-Methode, die */cgi-bin/datei1* für den Parameter *action* verwendet, ausgeführt. In dem Verzeichnis */usr/local/apache/cgi-bin* können Perl-, Shell-Skripte, C-Programme usw. als CGI-Programme eingesetzt werden. Es könnte zum Beispiel ein Verzeichnis *perl* unter dem Verzeichnis *cgi-bin* angelegt werden, um alle Perl-Skripten in einem separaten Verzeichnis unterzubringen. Schöner ist es, in der Datei *httpd.conf* einen zweiten Alias hinzuzufügen, um ein separates Verzeichnis, in dem sich eine große Anzahl von Perl-Skripten befindet, kenntlich zu machen.

```
# Meine Ergänzungen

ScriptAlias /perls/ "/usr/local/apache/perls/"

<Directory "/usr/local/apache/perls">
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
</Directory>
```

47 Die angeführten Beispiele wurden mit dem Apache-Server in der Version 1.3.6 ausprobiert.

Apache mit mod_perl

Jedes Mal, wenn ein Perl-CGI-Script aufgerufen wird, muss der Perl-Interpreter gestartet werden, um das Skript abzuarbeiten. Die Ladezeit ist im Verhältnis zur Ausführungszeit zu groß. Dieses wird als Startup-Penalty (penalty=Strafe) des Web-Servers bezeichnet. Das Modul *mod_perl* für den Apache-Server löst dieses Problem. Durch das Modul *mod_perl* kann ein Perl-Interpreter direkt in den Apache eingebaut werden. Der ständig aktive Perl-Interpreter hält CGI-Skripte im Speicher und ruft diese bei Bedarf als Unterprogramm auf. *Lars Eilebrecht* bezeichnet in seinem Buch *Apache Web-Server* das Modul *mod_perl* als ein Muss für Perl-Programmierer.

Installation von mod_perl

Der Link [mod_perl](http://perl.apache.org) auf der Home-Page von Apache führt zu der Seite <http://perl.apache.org>. Dort befinden sich alle weiteren Informationen zu dem Modul.

Eine kurze und gute Anleitung zur Installation von *mod_perl* befindet sich in der Datei *README.configure* des Apache-Server. Das tar-File des Apache-Servers habe ich unter dem Verzeichnis `/users/root/apache` ausgepackt. Damit ist der absolute Pfadname der Datei *README.configure*: `/users/root/apache/apache_1.3.6/README.configure`. Der folgende Ausschnitt aus der Datei beschreibt die Installation des Moduls *mod_perl*:

```
Apache and mod_perl
```

```
-----  
The Apache/Perl integration project (http://perl.apache.org/) from Doug MacEachern <dougmac@perl.apache.org> is a very powerful approach to integrate a Perl 5 interpreter into the Apache HTTP server both for running Perl programs and for programming Apache modules in Perl. The distribution mod_perl-1.XX.tar.gz can be found on http://perl.apache.org/src/. Here is how you can build and install Apache with mod_perl:
```

```
$ gunzip <apache_1.3.X.tar.gz | tar xvf -  
$ gunzip <mod_perl-1.XX.tar.gz | tar xvf -  
$ cd mod_perl-1.XX  
$ perl Makefile.PL APACHE_SRC=../apache_1.3.X/src \  
                  DO_HTTPD=1 USE_APACI=1 \  
                  [EVERYTHING=1 ...]  
  
$ make  
$ make install  
  
[optionally you now have the chance to prepare or add more  
third-party modules to the Apache source tree]  
  
$ cd ../apache_1.3.X  
$ ./configure --prefix=/path/to/apache \  
              --activate-module=src/modules/perl/libperl.a \  
              [--enable-shared=perl]  
  
$ make  
$ make install
```

Somit sah der Aufruf des Kommandos aus dem Ausschnitt

```
$ perl Makefile.PL ...
```

bei mir so aus:

```
> perl Makefile.PL APACHE_SRC=/users/root/apache/apache_1.3.6/src DO_HTTPD=1  
USE_APACI=1 EVERYTHING=1
```

Nach dem Übersetzen und Installieren des Moduls *mod_perl* mit

```
> make; make install
```

muss der Apache-Server übersetzt werden.

Damit ist das Modul *mod_perl* zwar installiert, wird aber noch nicht benutzt. Dafür sind Änderungen in *httpd.conf* vorzunehmen. Eine Beschreibung, welche Einträge in der Konfigurations-datei des Apache-Servers notwendig sind, ist mit der URL http://perl.apache.org/dist/cgi_to_mod_perl.html zu finden. Nachfolgendes Konfigurationsbeispiel ordnet dem Verzeichnis */perldir* den "Handler" *perl-script* zu. Handler ist ein Begriff beim Apache-Server, mit dem eine Funktion (-alität) zugeordnet werden kann.

```
ScriptAlias /perldir/ /usr/local/apache/perldir/

<Location /perldir>
  SetHandler perl-script
  #
  # Apache-Modul mit dem Namen: Apache::Registry
  PerlHandler Apache::Registry
  PerlModule CGI
  Options ExecCGI
</Location>
```

Diese Konfiguration wurde mit der Version 1.21 des Moduls *mod_perl* ausprobiert. Das Ergebnis macht Spaß: Eine Web-Seite, ausgegeben durch ein CGI-Perl-Skript, erscheint sofort auf dem Bildschirm. Die erste Zeile in einem CGI-Skript braucht und sollte nicht den expliziten Aufruf des Perl-Interpreters enthalten. Das heißt, eine Zeile wie: `#!/usr/bin/perl` sollte nicht in dem Skript stehen, da dadurch ein neuer Prozess gestartet wird.