Apache 2 und libapache2-mod-fcgid

Die Einrichtung von mod-fcgid für Apache 2.x unter Debian ist denkbar einfach, man muss nur einmal herausgefunden haben, wie es geht. Hier eine kurze aber vollständige Zusammenfassung:

```
apt-get install libapache2-mod-fcgid
```

Es empfiehlt sich, die neuere Version 2.2 von mod-fcgid zu installieren, muss also auf unstable oder testing ausweichen. Die alte Version 1.x hat laut diversen Quellen im Internet diverse unschöne Probleme.

Die Datei */etc/apache2/mods-available/fcgid.conf* folgendermaßen anpassen:

fcgid.conf

```
<IfModule mod_fcgid.c>
   # IdleTimeout n (300 seconds)
   # An idle fastcgi application will be terminated after IdleTimeout
seconds.
   IdleTimeout 600

   # IdleScanInterval n (120 seconds)
   # The scan interval for idle fastcgi applications.
   IdleScanInterval 240

   # BusyTimeout n (300 seconds)
   # A fastcgi application will be terminated if handing a single
request
   # longer than busy timeout.
   BusyTimeout 300

   # BusyScanInterval n (120 seconds)
   # The scan interval for busy timeout fastcgi applications.
   BusyScanInterval 120

   # ErrorScanInterval n (3 seconds)
   # The scan interval for exit pending fastcgi applications. fastcgi
   # applications will be terminated within this scanning.
   ErrorScanInterval 6

   # ZombieScanInterval n (3 seconds)
   # The scan interval for zombie process.
   ZombieScanInterval 6

   # ProcessLifeTime n (3600 seconds)
   # A fastcgi application will be terminated if lifetime expired,
   # even no error is detected.
   ProcessLifeTime 3600
```

```
   # SocketPath path (logs/fcgidsock)
   # The directory to put the UNIX domain socket. (UNIX only)
   # This directory should be writable only by apache user
   SocketPath /var/lib/apache2/fcgid/sock

   #SharememPath path (logs/fcgid_shm)
   #The share memory file path. (UNIX only) (version >= 2.1 only)
   SharememPath /var/lib/apache2/fcgid/shm

   # SpawnScoreUpLimit n (10)
   # The spawn-speed control score up water limit. Score increases
while
   # a process is spawned or terminated, and decreases as time
progresses;
   # while the score is higher than SpawnScoreUpLimit, the spawning
will be
   # held for a while. The higher this number is, the higher speed of
the
   # spawning can be.
   SpawnScoreUpLimit 10

   # SpawnScore n (1)
   # The weight of spawning. This weight will be plused to the spawn-
control
   # score on every spawn. The higher this number is, the lower speed
of
   # spawning can be.
   SpawnScore 1

   # TerminationScore n (2)
   # The weight of termination. This weight will be plused to the score
while
   # fastcgi process terminates. The higher this number is, the lower
speed
   # of spawning can be.
   TerminationScore 2

   # MaxProcessCount n (1000)
   # The max count of total fastcgi process count.
   MaxProcessCount 750

   # DefaultMaxClassProcessCount n (100)
   # The maximum number of fastcgi application instances allowed to run
for
   # particular one fastcgi application.
   DefaultMaxClassProcessCount 15

   # DefaultMinClassProcessCount n (3)
   # The minimum number of fastcgi application instances for any one
fastcgi
```

```
    # application.
    # Idle fastcgi will not be killed if their count is less than n
    # Set this to 0, and tweak IdleTimeout
    DefaultMinClassProcessCount 0

    # DefaultInitEnv env_name env_value
    # The default environment variables before a fastcgi application
    # is spawned. You can set this configuration more than once.

    # IPCConnectTimeout n (3 seconds)
    # The connect timeout to a fastcgi application.
    IPCConnectTimeout 30

    # IPCCommTimeout n (20 seconds)
    # The communication timeout to a fastcgi application. Please
increase this
    # value if your CGI have a slow initialization or slow respond.
    IPCCommTimeout 120

    # OutputBufferSize n (64k bytes)
    # CGI output cache buffer size.

    # PHP_Fix_Pathinfo_Enable n(n=0/1, default 0)
    # If you are using PHP and set cgi.fix_pathinfo=1 in php.ini, set
PHP_Fix_Pathinfo_Enable 1.
    # From php.ini:
    # cgi.fix_pathinfo provides *real* PATH_INFO/PATH_TRANSLATED support
for CGI. PHP's
    # previous behaviour was to set PATH_TRANSLATED to SCRIPT_FILENAME,
and to not grok
    # what PATH_INFO is. For more information on PATH_INFO, see the cgi
specs. Setting
    # this to 1 will cause PHP CGI to fix it's paths to conform to the
spec. A setting
    # of zero causes PHP to behave as before. Default is zero. You
should fix your scripts
    # to use SCRIPT_FILENAME rather than PATH_TRANSLATED.
    # cgi.fix_pathinfo=1
    PHP_Fix_Pathinfo_Enable 1

    MaxRequestsPerProcess 500

    AddHandler fcgid-script .php
    AddHandler fcgid-script .fcgi

    AddType application/x-httpd-php .php

    <Directory "/var/www/php-fcgid-scripts">
        SetHandler fcgid-script
        Options +ExecCGI
        Order allow,deny
```

```
        Allow from all
    </Directory>
</IfModule>
```

Das Modul, falls nicht schon durch die Installation erledigt, aktivieren durch manuelle Anlage der Symlinks in */etc/apache2/mods-enabled* oder durch Benutzung des Befehls *a2enmod*. Darauf achten, dass andere fastcgi-Module wie libapache2-mod-fastcgi (die beiden Dateien heißen *fastcgi.conf* und *fastcgi.load*) deaktiviert sind.

Pro VHost eine Starter-Datei *php-fcgid-starter* anlegen. Die Verzeichnisstruktur kann beispielsweise folgendermaßen aussehen:

```
root:root     /var/www/php-fcgid-scripts
web001:web001 /var/www/php-fcgid-scripts/www.domain1.tld
web001:web001 /var/www/php-fcgid-scripts/www.domain1.tld/php-fcgid-starter
web002:web002 /var/www/php-fcgid-scripts/www.domain2.tld
web002:web002 /var/www/php-fcgid-scripts/www.domain2.tld/php-fcgid-starter
```

Vorangestellt hier die Besitzer und Besitzergruppen. Zusätzlich zu mod-fcgid empfiehlt sich die Verwendung von suExec, warum hier jetzt aber nicht näher eingegangen wird.

Das Verzeichnis */var/www/php-fcgid-scripts* wurde oben in der *fcgid.conf* schon entsprechend berechtigt.

Inhalt der Starter-Datei:

php-fcgid-starter

```
#!/bin/sh
PHPRC="/etc/php5/cgi/www.domain1.tld"
export PHPRC
FCGI_SERVER_MAX_STDERR_LINE_LEN=2047
export FCGI_SERVER_MAX_STDERR_LINE_LEN
exec /usr/bin/php5-cgi
```

Im Verzeichnis */etc/php5/cgi/www.domain1.tld* liegt die php.ini für diesen VHost. Manche HowTo's gehen davon aus, dass alle VHosts die gleiche php.ini verwenden, wir halten das nicht für praktikabel. Aber auch auf das Thema php.ini wollen wir hier nicht näher eingehen.

FCGI_SERVER_MAX_STDERR_LINE_LEN wurde deswegen hier auf 2047 gestellt weil der Defaultwert auf 1023 liegt und es mit dem Defaultwert ab und an Probleme gab.

Anpassung der VHosts: Innerhalb von *<VirtualHost> </VirtualHost>* folgendes eintragen:

```
<Directory /var/www/www.domain1.tld>
  FCGIWrapper /var/www/php-fcgid-scripts/www.domain1.tld/php-fcgid-starter
.php
  Options +ExecCGI
```

```
</Directory>
```

Das DocRoot des VHosts liegt in diesem Fall unter */var/www/www.domain1.tld*.

Damit sollte alles konfiguriert sein. Die Config des Apachen reloaden mit */etc/init.d/apache2 reload* und die neue Konfiguration testen. Es empfiehlt sich natürlich, dies nicht auf einem Produktivsystem zu testen sondern sich zuerst einmal an einer Labormaschine zu versuchen.

Probleme mit 500er Fehlern nach 40 Sekunden:

Auf der Webseite erscheint ein Error 500, im error.log des VHosts tauchen z.B. folgende Fehler auf:

```
mod_fcgid: read data timeout in 40 seconds
```

In der hot-chilli.net Umgebung überschreiben die oben angegebenen Werte nicht die Defaultwerte aus dem Paket! Leider half weder dieser noch dieser Hinweis, das Problem in den Griff zu bekommen. Die Werte wurden weiterhin ignoriert. Abhilfe schaffte bei uns nur eine Neukompilierung des Paketes, dabei wurden o.g. Werte als Defaultwerte in den Sourcecode eingetragen. Aus diesem Paket wurde dann die fcgid.so entnommen und unter /usr/lib/apache2/modules eingespielt. Dieses Paket stellen wir auf Anfrage gerne zur Verfügung, Kontaktdaten siehe Impressum auf unserer Webseite.

Hinweis: mod-fcgid hat gegenüber mod-fastcgi den gewaltigen Vorzug, dass die PHP-Childs dynamisch verwaltet werden. Dies gilt sowohl für den Start wie das das Beenden der Childs.

Weitere Infos:

Warum PHP via FastCGI und nicht via mod-php oder suPHP?

Thread "Erfahrungen apache2 mit PHP 5 als FastCGI" aus debianform.de

PS: An dieser Stelle auch herzlichen Dank an cirox aus dem Rootforum. 

From:
https://wiki.hot-chilli.net/ - **Hot-Chilli Wiki**

Permanent link:
https://wiki.hot-chilli.net/allgemeine-howtos/apache/php/apache-2-und-libapache2-mod-fcgid

Last update: **2016/01/20 11:11**