

Apache + mod_perl (част 1)

Един от начините да генерирате динамични страниците на вашия сайт е CGI скриптове написани на Perl. Но често проблема който възниква е как да ги накараме да се изпълняват по-бързо.

CGI (Common Gateway Interface) се е наложило като стандарт сред сървърите. Предимството на CGI е че всеки web сървър го поддържа и програмите за него могат да се напишат на различни програмни езици и голямо количество платформи.

Недостатък на CGI е неговата неефективност. Всяко извикване на CGI програми създава нов процес на сървъра. Ако напишете CGI на Perl то тогава всеки път при извикване на програмата се зарежда и стартира и ново копие на Perl интерпретатора. Това води до използване на още памет и процесорно време. Естествено е да си зададем въпроса дали няма по-ефективен начин да се изпълняват програмите на Perl без да претърпят големи корекции.

Отговорът на този въпрос е mod_perl – модул за известния Apache Web Server. Този модул зарежда само веднъж копие на Perl интерпретатора в Apache, като по този начин не само ускорява изпълнението на perl скриптовите но и добавя нови възможности към сървъра.

Може би най-трудния момент при използването на mod_perl е неговото инсталиране и конфигуриране. За целта първо трябва да си свалите от интернет последните версии на Apache и mod_perl. Можете да ги откриете на следните адреси:

След като сте ги свалили ги копирайте в една директория – например /usr/local/download/ и напишете:

```
cd /usr/local/downloads
```

Линукс. Apache+MOD_PERL

Написано от
Понеделник, 20 Февруари 2012 09:28 -

```
tar -zxvf apache_1.3.11.tar.gz  
tar -zxvf mod_perl-1.2.tar.gz
```

Тези команди ще разархивират файловете и ще създадат две директории със сорс кодовете на Apache и mod_perl.

Сега е време да конфигурира Apache за първи път. Ако не сте правили това преди напишете следните редове:

```
cd apache-1.3.11/  
./configure --prefix=/www
```

Където /www може да замените с директорията където искате да инсталирате вашия web сървър.

Следващите команди които трябва да напише са следните:

```
cd ../mod_perl-1.2/
```

Ако искате да разберете кои настройки на mod_perl да използвате прочетете README и INSTALL файла преди да напишете тази команда.

```
perl Makefile.PL
```

Ако искате да се активират всички директиви на mod_perl напишете:

```
perl Makefile.PL EVERYTHING=1
```

Линукс. Apache+MOD_PERL

Написано от
Понеделник, 20 Февруари 2012 09:28 -

При изпълнение на тази команда инсталационната програма ще ви зададе два въпроса.

```
Configure mod_perl with ../apache-1.3.11/src ? [y]  
Shall I build httpd in ../apache_1.3.11/src for you? [y]
```

Отговорете и на двата с "y" или просто натиснете клавиша ENTER.

Командите които остава да напишете са

```
make  
make install
```

Преди да напишете обаче make install спрете всички работещи сървъри с командата killall httpd

Ако до тук сте изпълнявали правилно командите трябва да сте готови да пуснете вашият нов сървър. За целта пишете:

```
/www/bin/apachectl start
```

Ако искате да се уверите, че работи вашия нов сървър напишете следните команди:

```
telnet localhost 80
```

и след това

HEAD / HTTP/1.0

Резултата от тази команда трябва да съдържа и следния ред
Server: Apache/1.3.11 mod_perl/1.20

Което ще рече че вече имате работещ сървър на порт 80 с mod_perl модул.

За да използвате възможностите на mod_perl ще трябва да добавите следните редове в httpd.conf файла(той се намира в под директория conf на главната директорията на сървъра):

```
SetHandler perl-script  
PerlHandler Apache::Registry  
Options ExecCGI
```

Чрез и тагове показваме, че искаме тези настройки да важат само за /perl-bin директорията отколкото за целия сървър. Това ще рече че само файловете в директорията /perl-bin ще се третират единствено като perl скриптове и ще се изпълняват от заредения вече в Apache Perl интерпретатор.

Добавете също така и следните директиви в httpd.conf

```
PerlSendHeader On  
PerlWarn On  
(ако искате предупрежденията при изпълнение на програмата да се показват, отколкото да се записват в error.log файла)  
PerlTaintCheck On  
(ако искате да добавите допълнителна сигурност към скриптовете си).
```

След което запишете файла и проверете новата конфигурация със следната команда:

```
/www/bin/apachectl configtest
```

Apache + mod_perl (част 2)

Време е за нашата първа програма. Запишете я под името test.pl в /perl-bin директорията.

```
use strict;
print "Content-type: text/html\n";
foreach my $key (sort keys %ENV)
{
print ""$key" = "$ENV{$key}"
n";
}
}
```

За да я изпълните пишете във вашият браузър

```
http://localhost/perl-bin/test.pl
```

Тази команда трябва да ви покаже Environment обвивката на вашата система.

При писането на програми в mod_perl трябва да се съобразявате със следните неща

1. когато се изпълнява програма тя се компилира и се пази в паметта. При извикване на командата exit вие изтривате програмата от паметта което води до обратен ефект. Затова използвайте винаги командата Apache::exit вместо exit

2. Ако искате програма ви да работи коректно трябва винаги да инициализирате променливите. Вземете за пример следната програма

Написано от
Понеделник, 20 Февруари 2012 09:28 -

```
#!/usr/local/bin/perl  
print "Content-type: text/html\n";  
++$i;  
print $i;
```

Ако използвате стандартният Perl програма винаги ще връща 1. Но при mod_perl всеки път когато извикаме програмата ще получаваме число по-голямо от предишното с единица – 1, 2, 3, ... Това е така защото стойностите на променливите от програмата се пазят в паметта и при всяко следващо извикване променливата \$i не се занулява.

1. Всяко използване на командата fork води до създаване на ново копие на сървъра в паметта. Това допълнително натоварва системата и забавя бързодействието

За да разберете как да използвате mod_perl максимално ефективно прочетете информацията на следния адрес: <http://perl.apache.org/tuning/>

С mod_perl не само ускорявате изпълнението на Perl CGI скриптовете но и добавяте нови възможности към сървъра.

Например ако искате да получите информация за състоянието на сървъра, за броя на компилираните perl скриптове и допълнителни технически подробности трябва да добавите следните редове в конфигурационния файл:

```
SetHandler perl-script  
PerlHandler Apache::Status
```

Тук /perl-status не е физическа директория на вашият сървър. Ако искате само локално да виждате тази информация пишете:

Линукс. Apache+MOD_PERL

Написано от

Понеделник, 20 Февруари 2012 09:28 -

```
SetHandler perl-script
PerlHandler Apache::Status
order deny,allow
deny from all
allow from 127.0.0.1
```

В заключение ще кажа, че mod_perl не е панацея. Бързодействието което постигате е за сметка на повече използвана памет, тъй като всеки скрипт който се изпълнява се кешира.

Ако паметта не е проблема за вас и единственото ви желание е по-бърз сайт то тогава комбинирането на мощта на Apache и mod_perl е за вас.