

## MySQL en Perl, het verstandshuwelijk



door Georges Tarbouriech  
<georges.t/at/linuxfocus.org>

### *Over de auteur:*

Georges een lange-tijd Unix gebruiker. Hij houdt van produkten die meedragen aan het verspreiden van vrije software oplossingen in het professionele gebied.

*Vertaald naar het Nederlands door:*

Guus Snijders  
<ghs(at)linuxfocus.org>



### *Kort:*

MySQL en Perl bestaan reeds langere tijd. Ze worden nog steeds veel gebruikt, ondanks dat de "mode" veranderlijk is. Dit artikel gaat over samenwerking tussen deze 2 produkten, of het nu op het Internet is of op je lokale netwerk. Het gegeven voorbeeld is voor Unix systemen, vrij of niet, ook al kan het aangepast worden aan andere wijdverspreide "systemen".

Wat dit artikel is: een klein voorbeeld van wat je kunt doen met dit paar, het laten zien van eenvoudig gebruik, snelheid, betrouwbaarheid, veiligheid...

Wat dit artikel niet is: geen MySQL tutorial of Perl tutorial; geen MySQL of Perl review.

Aansluitend, we zullen MySQL aan het werk zien in combinatie met Perl, zonder te vergeten dat "er meer dan een manier is om dingen gedaan te krijgen".

---

## Over dit mooie paar

**MySQL** is een Relational DataBase Management System (RDBMS) en is beschikbaar op <http://www.mysql.com>. Het wordt verspreid onder de GNU GPL, vrij, afhankelijk van waarvoor het gebruikt wordt. Bekijk de licentie politiek op MySQL's website. Het werkt als server of client op vele platformen. Enkele andere RDBMS'en bestaan als vrije software en we zullen geen verglijkingen maken, daar de keuze van MySQL voor dit artikel een voorwaarde was. Ook zullen we niet kijken naar de grote "commerciële" jongens zoals Informix, Oracle, Sybase... Het volstaat te vermelden dat MySQL waarschijnlijk het meest gebruikte RDBM op het Internet is. Voor dit artikel zullen we gebruik maken van 3.23.36 (weer, voorwaardelijk). Op moment van schrijven is de huidige stabiele versie 3.23.46 en de experimentele is de lang verwachte versie 4.0. Deze kunnen gedownload worden als broncode om te compileren of als pakketten.

Om MySQL in combinatie met Perl te gebruiken, heb je nog iets meer nodig: de Perl DBI modules. Op

z'n minst kun je downloaden DBI, Mysql-Mysql-modules, Data-Dumper en Data-ShowTable. De installatie ervan komt hier verder niet ter sprake, daar deze niet moeilijk is en pakketten voorzien je van alles wat je moet weten.

**Perl** staat voor Practical Extraction and Report Language. In het begin was het bedoelt voor het manipuleren van documenten (parsen, uitpakken...) maar al snel werd het meer dan dit. Je kunt bijna alles doen met Perl. Van administratieve zaken tot cgi scripts via echte applicaties en natuurlijk, database interfaces.

Perl is onderdeel van vele (zo niet alle) Unix distributies of ze nu vrij zijn of niet. De huidige stabiele versie is 5.6.1 en de experimentele is 5.7.2 op moment van schrijven. Voor het artikel zullen we een goede oude gebruiken, 5.005\_03. Voor het geval dat je Perl niet geïnstalleerd hebt (hoe is dit mogelijk?) kun je het krijgen op <http://www.perl.com>. Perl voorziet je van tonnen modules voor bijna alles. Je kunt ze vinden op het CPAN gedeelte van deze website: een ware goudmijn!

Ten slotte, om met beide tools te werken, verassing: je hebt een webserver nodig! Apache lijkt de juiste keus, daar het onderdeel is van veel Unix distributies, vrij of niet, zoals gebruikelijk. Voor het geval je hem niet hebt (waar heb je die distro gevonden?), hij is beschikbaar op <http://www.apache.org>.

## Het voorbeeld dat we zullen gebruiken

Je hebt waarschijnlijk al gemerkt dat LinuxFocus een meertalige magazine is. Dat betekent dat, als je een redacteur bent, je de status van veel artikelen moet managen, zoals de vertaling ervan. In andere woorden, wie doet wat, wanneer... Op het moment zijn er zo'n 200 artikelen beschikbaar, gemiddeld in zo'n 5 talen. Dat maakt samen zo'n 1000 artikelen (hoe slim ben ik?) en de teller loopt nog steeds! Dit spul moet worden opgeslagen, geformatteerd, samengevat... Hoe denk je dat dit management word gedaan? Met Perl, natuurlijk.

Onze hoofdredacteur, Guido Socher heeft veel Perl programma's geschreven om het werk eenvoudiger te maken. Hij schreef ook een driedelige Perl tutorial en een Perl boek bespreking. Zie de referentie sectie aan het eind van het artikel.

Javi, de Spaanse redacteur, schreef een programma om de status van de vertalingen te managen... in Perl.

Atif, een van onze ster auteurs, komt uit Perl land, dat is waarom zijn Perl zijn moedertaal is. Ook heeft hij bijgedragen aan MySQL, met het verbeteren van een web administratie tool. Wederom, zie de referenties sectie.

Al dit om te zeggen dat als je op zoek bent naar een Perl paradijs... kom bij Linuxfocus.

Daar ik een van de Franse Linuxfocus redacteurs ben... en nogal lui, creëerde ik mijn eigen LinuxFocus database, met behulp van, raad eens: MySQL en Perl!

## De database creëren

Dit gaat ervan uit dat MySQL correct geïnstalleerd is, users aangemaakt en beschermt met passwords. De installatie valt buiten het bereik van dit artikel en de geweldige documentatie die met MySQL wordt bijgeleverd zou je alles moeten vertellen.

Start de MySQL server met het *mysql.server* script, daar deze de *safe\_mysqld* daemon aanroept en je zo opties kunt doorgeven aan deze daemon.

Maak contact met de server met

```
mysql -h host -u user -p
```

Als de server op je locale computer draait, hoef je *-h host* niet te gebruiken.

Na het ingeven van het password ben je verbonden met de server (wel, dat zou je moeten!). En dan kun je nu je database opzetten.

Op de mysql prompt, type

```
CREATE DATABASE lf;
```

Dit is ons voorbeeld (lf staat voor LinuxFocus) en vanzelfsprekend kun je hier iedere naam opgeven die je wilt. Vervolgens is het tijd om de toegestane users permissies toe te kennen, hier wordt aangenomen dat je het recht hebt om dat te doen (oftewel, dat je administratie rechten hebt als de verbonden user).

Als je een user in staat wilt stellen de database te managen, kun je hem de permissies geven met

```
GRANT ALL ON lf.* TO username;
```

Selecteer de database die je zojuist hebt aangemaakt met

```
USE lf
```

Maak een tabel aan die voldoet aan jouw eisen. In ons voorbeeld maken we een tabel aan genaamd trissue:

```
CREATE TABLE trissue (num INTEGER UNSIGNED, category VARCHAR(25), title VARCHAR(40), author VARCHAR(20), en VARCHAR(20), es VARCHAR(20), fr VARCHAR(20), de VARCHAR(20), nl VARCHAR(20), ru VARCHAR(20), tk VARCHAR(20), issue VARCHAR(20));
```

controleer of hij is aangemaakt zoals verwacht met:

```
USE lf  
SHOW TABLES;  
DESCRIBE trissue;
```

Dat is alles

Nu hebben we nog data nodig. Om data in een lege tabel te laden is het het eenvoudigst om gebruik te maken van een tekst bestand met tab scheidingstekens. Als je text klaar is, type:

```
LOAD DATA LOCAL INFILE "maindb.txt" INTO TABLE trissue;
```

Als je text bestand correct is, is de tabel nu gevuld. Je kunt het controleren met:

```
SELECT * FROM trissue;
```

Dit zou een lange lijst moeten laten zien. Nu ben je in staat om ieder type data op te halen met behulp van queries.

So far, so good.

Tot nu toe, hebben alleen gebruik gemaakt van MySQL en zijn we in staat om er alles mee te doen. Dus wat is het punt met Perl?

## Perl aan het werk

Perl zal ons helpen om de queries te automatiseren, de resultaten in een web browser weer te geven, enz. Wederom, dit gaat ervan uit dat de Perl modules juist geïnstalleerd zijn om MySQL te gebruiken in combinatie met Perl.

We zullen nu Perl scripts schrijven om te gebruiken als cgi scripts. Deze stellen ons in staat om Perl en HTML te mixen om de database te ondervragen en de output te op te maken (te formatteren).

We zullen een eenvoudig voorbeeld script gebruiken, welke ons toestaat alle artikelen te zoeken die geschreven zijn door dezelfde auteur. We zullen de artikel nummers weergeven, de category, de titel, de namen van de vertalers voor de verschillende talen (alleen volledig werkende projecten) en de uitgave van wanneer de artikelen gepubliceerd zijn.

Je kunt dit script gebruiken als een model voor eigen gebruik, maar pas op dat dit een niet erg veilig programma is. Een versie voorzien van meer commentaar kun je =>hier<= downloaden.

```
#!/usr/bin/perl -Tw
# Eerst zeggen we dat dit een "Tainted" Perl script is.
#
# Dit is commentaar
# db consult
#
# We zullen de Perl DBI module gebruiken
use DBI;

# Als cgi :
use CGI qw(param());

print <<END_of_start;

Content-type: text/html

<html>
<title>LFAuthors main db</title>

<center><TABLE>
<TR VALIGN=TOP>
<TD><form action="/cgi-bin/lf.cgi" method="get">

# Hier komt de titel van de knop van de startpagina
<input type="submit" value=" LFAuth ">
</form>
</TD>
</TR>
</TABLE>
```

Nu vertellen we het script om de database te ondervragen

```
<center><H2>Search by author</H2></center>
```

```
<form action="/cgi-bin/lf.cgi" method="get">Author name : <input  
type="text" size="30" name="author"><input type="submit"  
value="Search..."></form></center>
```

END\_of\_start

```
if (param("author") ne '') {  
$author = param("author");
```

```
$autsrch.='';  
$autsrch=$author;  
$autsrch.='';
```

```
# We melden ons bij de database genaamd lf aan als gebruiker doe
```

```
$dbh = DBI->connect("DBI:mysql:lf","doe",'');
```

```
$sth = $dbh->prepare("  
select *  
from trissue  
where  
author = $autsrch  
");
```

```
$sth->execute;
```

Nu vragen we het script om de resultaten van de query klaar te maken en weer te geven. Ofwel we gaven geen criteria, en het zal de volledige inhoud van de database weergeven, of we gaven een naam op en het zal alle artikelen die corresponderen met de naam van de auteur. Indien je duizenden records in je database hebt, raad ik je niet aan om de volledige inhoud weer te laten geven!

```
print <<END_suite;
```

```
<center>  
<TABLE BORDER=>  
<tr bgcolor=#A1C4EE>  
<th width=60 align=CENTER><font color=#000000> Num </font></th>  
<th width=110 align=CENTER><font color=#000000> Category </font></th>  
<th width=110 align=CENTER><font color=#000000> Title </font></th>  
<th width=110 align=CENTER><font color=#000000> Author </font></th>  
<th width=110 align=CENTER><font color=#000000> En </font></th>
```

```

<th width=110 align=CENTER><font color=#000000> Es </font></th>
<th width=110 align=CENTER><font color=#000000> Fr </font></th>
<th width=110 align=CENTER><font color=#000000> De </font></th>
<th width=110 align=CENTER><font color=#000000> Nl </font></th>
<th width=110 align=CENTER><font color=#000000> Ru </font></th>
<th width=110 align=CENTER><font color=#000000> Tk </font></th>
<th width=110 align=CENTER><font color=#000000> Issue </font></th>
</tr>

```

END\_suite

```

while( ($num,$category,$title,$author,$en,$es,$fr,$de,$nl,$ru,$tk,$issue) =$sth->fetchrow() ) {
print "<tr>";
print "<td width=60 bgcolor=#FFFFFFE8 align=center> $num</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $category</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $title</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $author</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $en</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $es</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $fr</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $de</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $nl</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $ru</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $tk</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $issue</td>";
print "</tr>";

}
print "</TABLE>";
print "<BR>";
print "<BR>";
print "<br>";

```

} else {

# DB Connect

```
$dbh = DBI->connect("DBI:mysql:lf","doe","");
```

# Zoek

```

$sth = $dbh->prepare("
select *
from trissue
");

```

```
$sth->execute;
```

```
# Display result
```

```
print <<SUITE;
```

```
<center>
<TABLE BORDER=>
<tr bgcolor=#A1C4EE>
<th width=60 align=CENTER><font color=#000000> Num </font></th>
<th width=110 align=CENTER><font color=#000000> Category </font></th>
<th width=110 align=CENTER><font color=#000000> Title </font></th>
<th width=110 align=CENTER><font color=#000000> Author </font></th>
<th width=110 align=CENTER><font color=#000000> En </font></th>
<th width=110 align=CENTER><font color=#000000> Es </font></th>
<th width=110 align=CENTER><font color=#000000> Fr </font></th>
<th width=110 align=CENTER><font color=#000000> De </font></th>
<th width=110 align=CENTER><font color=#000000> Nl </font></th>
<th width=110 align=CENTER><font color=#000000> Ru </font></th>
<th width=110 align=CENTER><font color=#000000> Tk </font></th>
<th width=110 align=CENTER><font color=#000000> Issue </font></th>
</tr>
```

```
SUITE
```

```
while( ($num,$category,$title,$author,$en,$es,$fr,$de,$nl,$ru,$tk,$issue) =$sth->fetchrow() ) {
print "<tr>";
print "<td width=60 bgcolor=#FFFFFFE8 align=center> $num</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $category</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $title</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $author</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $en</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $es</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $fr</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $de</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $nl</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $ru</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $tk</td>";
print "<td width=110 bgcolor=#FFFFFFE8 align=left> $issue</td>";
print "</tr>";

}
print "</TABLE>";
print "<BR>";

}
```

```
print end_html;
$sth->finish;
```

```
# Verbreek de verbinding
```

```
$dbh->disconnect;
```

```
exit;
```

Hier is de query output in een web browser::

Item	Category	Title	Author	En	Et	Fr	De	It	Ru	Tr	Issue
53	UNIX Basics	Regular Expressions	G.Socher	G.Socher	F.Gliero	J.D.Giard	G.Socher	F.Lambrechts	Xiv	unk	July1990
64	UNIX Basics	Finding Files	G.Socher	G.Socher	R.Sobko	J.Per	G.Socher	unk	unk	Triban	September1990
77	UNIX Basics	File Access Permissions	G.Socher	G.Socher	M.Morano	J.Per	G.Socher	E.Nikishagan	Aun	unk	January1995
114	Software Development	Part I	G.Socher	G.Socher	O.Abdelra	V.Hirou	K.Socher	F.Lambrechts	unk	unk	September1995
126	Software Development	Part II	G.Socher	G.Socher	H.Rodriguez	G.Tarbouriech	K.Socher	F.Lambrechts	nt	G.Aky	November1995
134	System Administration	Home Networking	G.Socher	G.Socher	V.Roselli	J.Peyralout	H.Radke	T.Ujbert	nt	nt	January2000
138	Software Development	Perl part III	G.Socher	G.Socher	J.R.Garcia	G.Tarbouriech	K.Socher	F.Lambrechts	nt	S.Denc	January2000
148	Software Development	SNMP for Linux	G.Socher	G.Socher	J.Palacios	E.Cappaso	nt	M.Reiners	nt	P.Karlu	March2000
151	System Administration	Setting up IP-Masquerading	G.Socher	G.Socher	P.Vega	J.Peyralout	M.Blank	T.Ujbert	K.Paukhikoz	K.Ozbak	May2000
185	Hardware	Using Serial Line LCD	G.Socher	G.Socher	J.Nieto	J.Per	J.Link	nt	K.Paukhikoz	Z.Gul	July2000
186	Hardware	A serial line shutdown button	G.Socher	G.Socher	nt	J.Peyralout	G.Socher	T.Ujbert	K.Paukhikoz	H.Kaya	January2001
182	System Administration	Using different ISPs	G.Socher	G.Socher	R.Velasco	J.Peyralout	G.Socher	nt	K.Paukhikoz	S.Balucu	March2001
204	Software Development	Backtracing Professional Perl Programers	G.Socher	G.Socher	A.Pardo	P.Lecade	P.Thalson	A.van Eijkelensborg	K.Paukhikoz	E.Mulla	July2001
215	System Administration	E-mail over UUCP	G.Socher	G.Socher	R.Hernando	G.Tarbouriech	S.Swin	T.Ujbert	E.Saenko	E.Mulla	September2001
220	Hardware	Alien Super Max Optical Mouse	G.Socher	G.Socher	nt	P.Tebich	G.Socher	nt	nt	E.Mulla	November2001
222	UNIX Basics	Running applications remotely with X11	G.Socher	G.Socher	nt	nt	nt	nt	nt	nt	nt

Dat is alles!

## De beveiligings kant

Vanzelfsprekend, als je een database service op je website wilt aanbieden, moet je het geheel beveiligen. Uiteraard gaan we geen stap-voor-stap manier geven om een website of database server te beveiligen. Echter, het is belangrijk om in te gaan op de basis.

In het kort, als je services gaat aanbieden op het web, is het eerste dat je moet beveiligen, je web server. Dit gaat ver voorbij dit artikel. als je meer wilt leren over het onderwerp, is er zat documentatie beschikbaar. Een goede plaats om te beginnen is het A nice place to start is het Linux Documentatie Project.

De volgende stap is de database server. als je een tool als MySQL installeert, vergeet niet om het beveiligings deel van de manual te lezen. Weer, de basis gaat over user passwords: laat nooit een account



achter zonder password, in het bijzonder de root account van de database (welke niet dezelfde zou moeten zijn de machine's root account). Een ander belangrijk punt is permissies: geef niet iedereen volledige rechten. Het lijkt logisch... en dat is waarom veel mensen het vergeten!

Om een stap verder te gaan, waarom de database niet chroot draaien? zie Mark's artikel "Alle diensten in Linux onder Chroot draaien" in deze uitgave. Hij spreekt over een andere database, maar wat hij zegt kan ook wordentoegepast op MySQL.

Een andere beveiligings maatregel gaat over de data circulatie. Het is geen slecht idee om de data te zenden en ontvangen door een tunnel. Je kunt het artikel Through the tunnel lezen voor meer informatie. Last but not least, veilig programmeren is een van de sleutels. Perl is een prachtige taal, maar het is vrij makkelijk om programmeer fouten ermee te maken. Een ander LinuxFocus artikel zal je leren wat te doen, vooral met Perl. Werp een blik hier. Het is het laatste artikel van een Secure Programming serie en behelst in het bijzonder cgi scripts. Een "must read"!

Let wel, dit gaat er vanuit dat je reeds een gehard systeem hebt zonder bekende beveiligingsgaten, met de nieuwste patches, en vele verplichte beveiligings tools met een NIDS (Network Intrusion Detection System) zoals snort (van <http://www.snort.org>), firewall, poort en beveiligings scanners (nmap, nessus), enz.

Als je het je kunt veroorloven kun je ook een aparte server voor iedere geleverde server inzetten: een web server, een database server... en de mirrors voor hoge beschikbaarheid. En zo verder! Er komt nooit een eind aan, daar veiligheid nooit volledig bereikt is. Je probeert alleen de risico's te reduceren... en ze worden iedere dag erger. Je bent gewaarschuwd.

## Wat anders?

Daar er meer dan een manier is, kun je je eigen weg kiezen. Er zijn vele RDBMS en vele talen om ermee te communiceren. Het idee achter het schrijven van dit artikel was om te laten zien hoe MySQL en Perl goed kunnen samenwerken.

Uiteraard was de keuze nogal subjectief: Ik hou van MySQL omdat het nogal klein in grootte is, werkt met vele besturingssystemen, het is snel, betrouwbaar... Ook heb ik veel waardering voor het werk gedaan door het MySQL team, zonder de vele deelnemers te vergeten. En wat ik het mooiste vind: die mensen probeerden niet het wiel opnieuw uit te vinden. Ze hielden de dingen eenvoudig.

Wat betreft Perl, alles is er reeds over gezegd: wat zou ik nog kunnen toevoegen? Ikzelf ben van mening dat je niet zonder kunt, of je nu een netwerkbeheerder bent of een ontwikkelaar, of wat dan ook. De Perl gemeenschap is een van de grootste bronnen van kennisdeling daar buiten. Er is ook een tijdschrift beschikbaar, genaamd Perl Journal, deze is nu opgenomen in SysAdmin magazine en verschijnt iedere 2 uitgaven. Als je je wilt inschrijven, ga naar <http://www.samag.com>.

Nu we het hebben over goed werk, hier komt het gebruikelijke off-topic gedeelte. Onze LinuxFocus lezers hebben waarschijnlijk niet het kleine aantal personen opgemerkt die betrokken zijn bij LinuxFocus. Desalniettemin kun je het lezen in veel verschillende talen. Heb je ooit opgemerkt dat sommige teams werken met slechts een of twee personen die al het werk doen? Zij zijn webmasters, vertalers, enz. Zie het Russische team, het Turkse Team: Je zult zien dat de meeste artikelen zijn vertaald door Kirill of Erdal. Zie de projecten in ontwikkeling zoals Portugees of Arabisch: zelfde resultaat! Hierbij wil ik ze alle feliciteren voor het goede werk dat ze doen. Allen bedankt: de vrije software gemeenschap heeft veel aan jullie te danken.

Sorry voor de afleiding, maar ik geloof dat gezegd moest worden.

Terug naar het onderwerp, laten we afsluiten met een paar woorden over vrije software. De mensen van MySQL of Perl verdienen veel dankjes. Ze voorzien ons van fantastische tools, en meestal voor niks.

Deze tools zijn echter even goed als echte commerciële producten (zo niet beter), worden vaak geüpdate, zeer goed gedocumenteerd en je kunt ze gebruiken op vrijwel alle Unix systemen. Kun je een vergelijkbaar produkt ergens anders vinden? Ik ben bang van niet!  
Dit artikel zal je waarschijnlijk niet veel leren, echter als het je stimuleert te kijken naar die producten, is het niet voor niks.  
Leven we niet in een prachtige tijd?

## References

Perl.org  
cpan.org, het perl archief

Guido's Perl tutorial:

Perl I  
Perl II  
Perl III

Professional Perl Programming boek bespreking:

Perl Programming

Atif's bijdrage aan MySQL.

Een LinuxFocus MySQL bespreking: een oud artikel, nog altijd up-to-date:

MySQL

Een oude LinuxFocus tweedelige SQL tutorial:

SQL Part I  
SQL Part II

---

Site onderhouden door het LinuxFocus editors team © Georges Tarbouriech "some rights reserved" see <a href="http://linuxfocus.org/license/">linuxfocus.org/license/</a> <a href="http://www.LinuxFocus.org">http://www.LinuxFocus.org</a>	Vertaling info: en --> -- : Georges Tarbouriech < <a href="mailto:georges.t@linuxfocus.org">georges.t@linuxfocus.org</a> > en --> nl: Guus Snijders < <a href="mailto:ghs(at)linuxfocus.org">ghs(at)linuxfocus.org</a> >
---	--