

Huiswerk Linux: Installatie MySQL

Zoals in de les aangekondigd, gaan we ons deze lesperiode bezighouden met een LAMP installatie. De afkorting staat voor Linux, Apache, MySQL en PHP. Deze combinatie is zeer populair omdat je daarmee Content Management Systemen (CMS), zoals WordPress kunt draaien. De opdracht voor deze week is: installeer de MySQL database server.

Over MySQL

MySQL is een database server, geschreven door Michael Widenius en David Axmark en vergelijkbaar met Microsoft SQL Server. De afkorting SQL staat voor *Structured Query Language* en is een soort protocol (taal) om met databases te praten. *My* is de naam van één de dochters van Michael Widenius.

MySQL is wat men noemt een *back-end* programma. Meestal zit er een applicatie voor welke gegevens uit de database haalt. Dit noemen we de *front-end* applicatie. MySQL kan worden gebruikt door verschillende front-end toepassingen (*WordPress*, *GitHub*, *Drupal*, *Joomla*, en gewone Linux gebruikers via het *mysql* client-programma).



MySQL installeren

Het compileren en installeren van software doen we namens de *root* gebruiker. Hierdoor worden automatisch de juiste toegangsrechten op de bestanden toegepast.

<h3>Cygwin</h3> <p>Sudo voor Cygwin gebruikers: rechts-klik op het icoon van de Cygwin terminal, en kies voor Als administrator uitvoeren. Installeer de <i>mysql</i> (client-programma's) en het <i>mysqld</i> (server) pakket:</p> <pre>apt install mysql mysqld</pre>	<h3>Debian</h3> <p>Gebruikers van Debian distributies (<i>Ubuntu</i>, <i>Knoppix</i>, <i>Kali</i>) starten een terminal met de toetsencombinatie <Ctrl><Alt>-T. We gebruiken het commando <i>sudo</i> om in te loggen met het <i>su</i> (become Super User) commando. Daardoor blijven we ingelogd:</p> <pre>sudo su</pre> <p>Installeer het <i>mysql-server</i> pakket:</p> <pre>apt install mariadb-server</pre> <p>Let op: je wordt soms gevraagd om een wachtwoord op te geven voor de <i>root@localhost</i> gebruiker¹. Geef dan het wachtwoord '0000'. Geef geen bestaande wachtwoorden!! Dit is zeer onveilig.</p>	<h3>Slackware</h3> <p>Op Slackware distributies, zoals Suse Linux, heet de terminal <i>Konsole</i> en gebruiken we de <i>zypper</i> installer:</p> <pre>zypper install mariadb zypper install mariadb-tools</pre> <p>Zorg er ook voor dat de service start als je je machine opnieuw opstart:</p> <pre>systemctl enable mysql</pre>
---	---	--

¹ De MySQL *root* gebruiker is niet gelijk aan de Linux *root* gebruiker. Dit is zeer verwarrend. De maintainers van MySQL hadden beter een andere gebruikersnaam kunnen verzinnen, bijvoorbeeld *admin*.

Unprivileged user aanmaken

De MySQL daemon werkt via een *unprivileged user* (zie ook: *Apache Daemon*). Deze gebruiker heet *mysql* en bestaat op sommige Linux systemen nog niet. We moeten hem dus eerst aanmaken. Dit doen we met het commando *useradd*:

```
useradd -M -d /var/empty mysql
```

Via optie *M* geef je aan dat er geen *home* directory moet worden aangemaakt. Met optie *d* (directory) geven we aan dat */var/empty* de home-directory van de nieuwe gebruiker is. Ook ontnemen we de nieuwe gebruiker de login-rechten. Dit doen we met het commando *usermod*:

```
usermod -L mysql
```

Via optie *L* (lock) worden de nieuwe gebruiker de login-rechten ontnomen. Dit noemen we een *unprivileged user*.

Stelsel-databases aanmaken

Voordat we MySQL kunnen starten, moeten we eerst de map voor de databases aanmaken. Ook heeft MySQL een aantal stelseldatabases nodig. Bovendien moet de Daemon gestart worden namens een *unprivileged user*. De bestanden moeten dus toegankelijk zijn voor de nieuwe gebruiker.

Aanmaken database-bestanden

Als eerste gaan we de map voor de MySQL databases aanmaken. Deze bestaat op sommige systemen nog niet. We maken de map aan met het commando *mkdir* (make directory):

```
mkdir /var/lib/mysql
```

Cygwin gebruikers

Nu gaan we de map vullen met de stelsel-databases. Dit wordt automatisch gedaan met het MySQL programma *install_db*:

```
mysql_install_db
```

Andere Linux gebruikers

Gebruikers van andere Linux distributies (*Ubuntu*, *Debian*, *Suze*, etc.) gebruiken de nieuwe optie van de MySQL Daemon:

```
mysqld --initialize --user=mysql
```

De (ingekorte) output van dit commando ziet er op een *Cygwin* systeem als volgt uit:

```
Installing MySQL system tables...
140518 16:53:04 [Warning] Setting lower_case_table_names=2 because file system for /var/mysql5/ is
case insensitive
OK
Filling help tables...
140518 16:53:05 [Warning] Setting lower_case_table_names=2 because file system for /var/mysql5/ is
case insensitive
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:
/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h boland-b3d79829 password 'new-password'
```

Let op: in bovenstaande melding wordt aangegeven dat je de *root* gebruiker een wachtwoord moet geven. Wij kiezen ervoor om dit niet te doen. De MySQL daemon wordt namelijk alleen gebruikt door Apache/PHP. Gebruikers van buitenaf krijgen geen toegang tot de database.

Wijzigen eigenaar database-bestanden

Zorg ervoor dat de *mysql* gebruiker in de database mag schrijven. Dit doen we door *mysql* de eigenaar van de directory te maken:

```
chown -R mysql /var/lib/mysql
```

Via de optie *R* (recursive) geven we aan dat ook alle onderliggende bestanden en mappen moeten worden gewijzigd.

Het hoera-moment

Allereerst moeten we de de MySQL Service starten. We gebruiken daarvoor het *service* programma:

```
service mysql start
```

Op de meeste systemen krijg je geen output te zien. Op systemen met output ziet een succesvol gestarte *mysql* daemon er als volgt uit. Dit is de output op een *Cygwin* systeem. Als je geen foutmeldingen krijgt, is MySQL succesvol gestart:

```
150811 13:35:04 [Warning] Setting lower_case_table_names=2 because file system for /var/lib/mysql/ is
case insensitive
InnoDB: The first specified data file ./ibdata1 did not exist:
InnoDB: a new database to be created!
150811 13:35:04 InnoDB: Setting file ./ibdata1 size to 10 MB
InnoDB: Database physically writes the file full: wait...
150811 13:35:05 InnoDB: Log file ./ib_logfile0 did not exist: new to be created
InnoDB: Setting log file ./ib_logfile0 size to 5 MB
InnoDB: Database physically writes the file full: wait...
150811 13:35:05 InnoDB: Log file ./ib_logfile1 did not exist: new to be created
InnoDB: Setting log file ./ib_logfile1 size to 5 MB
InnoDB: Database physically writes the file full: wait...
InnoDB: Doublewrite buffer not found: creating new
/etc/init.d/mysql start: mysqld started
```

Tijd voor het hoera-moment. Niet echt spectaculair, maar we kunnen in ieder geval kijken of MySQL is gestart. Dit doen we met het *ps* (process status) commando:

```
ps -ef | grep mysql
```

Het resultaat zou er ongeveer als volgt uit moeten zien. Onderstaand de output van een *Cygwin* systeem:

```
UID      PID    PPID  TTY          STIME COMMAND
mysql    476     488  ?           09:01:11 /usr/sbin/mysqld
```

Tot slot kunnen we kijken of de server reageert op een client-request. Dit doen we met het *mysqladmin* programma:

```
mysqladmin -u root ping
```

Met het *ping* commando sturen we een test-request naar de Daemon. Met de optie *u* (user) geven we aan namens welke gebruiker het ping request gestuurd moet worden. Je zou het volgende antwoord moeten krijgen:

```
mysqld is alive
```

Huiswerk opsturen

Zoals ik in de les heb uitgelegd, kunnen je de opdracht aftekenen door een e-mail met daarin de output van het *mysqladmin* commando te versturen. Dit doen we met een pijpleiding tussen de commando's *mysqladmin* en *email*. Zorg dat het onderwerp van de e-mail tussen aanhalingstekens staat:

```
mysqladmin -u root ping | email -s "Installatie MySQL" docent@localhost
```

Als je geen foutmelding ziet is de e-mail succesvol verstuurd.

Administrator uitloggen

We hebben de MySQL daemon geïnstalleerd, geconfigureerd en getest, maar we zijn nog steeds ingelogd als Administrator (*root*).

Geef tot slot **<Ctrl>-D** om de Administrator uit te loggen.

Problemen oplossen

Zoals altijd kun je het log raadplegen om te kijken of er foutmeldingen zijn:

```
tail /var/log/mysql/mysqld.log
```

Hieronder de meest voorkomende problemen en hun oplossingen.

Socket bestaat niet

Het kan zijn dat je de volgende foutmelding krijgt:

```
error: 'Can't connect to local MySQL server through socket '/tmp/mysql.sock' (2)'  
Check that mysqld is running and that the socket: '/tmp/mysql.sock' exists!
```

Dit betekent dat het *mysqladmin* client-programma de *socket*² van MySQL niet kan vinden. Meestal is dan de server nog niet gestart. Start de server met het service commando:

```
service mysql start
```

Socket bestaat nog steeds niet

Mysqladmin gaat ervan uit dat het socket-bestand zich in de directory */tmp* bevindt. In veel Linux distributies wordt het socket-bestand ergens anders geplaatst. Om erachter te komen waar het socket-bestand staat kunnen we de instellingen van MySQL opvragen. Dit doen we met het *mysqld* programma (de MySQL daemon):

```
mysqld --print-defaults
```

De output ziet er ongeveer als volgt uit:

```
mysqld would have been started with the following arguments:  
--user=mysql --pid-file=/var/run/mysqld/mysqld.pid --socket=/var/run/mysqld/mysqld.sock  
--port=3306 --basedir=/usr --datadir=/var/lib/mysql --tmpdir=/tmp --lc-messages-  
dir=/usr/share/mysql --skip-external-locking --bind-address=127.0.0.1 --key_buffer=16M --  
max_allowed_packet=16M --thread_stack=192K --thread_cache_size=8 --myisam-recover=BACKUP --  
query_cache_limit=1M --query_cache_size=16M --expire_logs_days=10 --max_binlog_size=100M
```

MySQL heeft drie verschillende bestandjes voor instellingen. Deze heten allemaal *my.cnf*, maar staan in verschillende locaties. MySQL leest instellingen uit de volgende bestanden:

Volgorde	bestand	toepassing
1.	<i>/etc/my.cnf</i>	alle client programma's
2.	<i>/etc/mysql/my.cnf</i>	het MySQL server programma
3.	<i>~/my.cnf</i>	specifieke gebruikers

Mysqladmin is een client-programma, dus we gaan de *socket* instelling in het */etc/my.cnf* bestandje wijzigen. Dit doen we met de *vi* tekst-editor:

```
vi /etc/my.cnf
```

² Een *socket* is een speciaal type bestand, waarmee met een daemon gecommuniceerd kan worden. Als een computerprogrammeur met een daemon wil communiceren, opent zij de betreffende socket als bestand en schrijft zij de gewenste commando's naar dat bestand. Een socket kun je herkennen aan de *s* aan het begin van de bestandsrechten.

Dit bestand ziet er in iedere Linux distributie anders uit, maar de instellingen zijn hetzelfde. Toets het **i** (insert) commando om naar de *INSERT* modus te gaan en zorg dat de nieuwe regel onder het kopje *[client]* staat:

```
[client]
socket = /var/run/mysqld/mysqld.sock
```

Toets **<Esc>** om de *INSERT* mode te verlaten en geef de commando's *w* (write) en *q* (quit):

```
:wq
```

Het bestand wordt opgeslagen en we zijn terug op de Linux command line.

Aanmaken systeem-database blijft hangen

Als je de systeem-database initialiseert (zie boven), blijft het commando hangen. Uiteindelijk werkt het, maar het duurt meer dan een minuut. Vertraging betekent dat de mysql daemon meerdere keren is gestart. Kijk eerst of dit het geval is:

```
ps -ef | grep mysql
```

De output is van een Suse systeem:

```
mysql 6099 1 1 18:03 ? 00:00:00 /usr/sbin/mysqld --defaults-file=/etc/my.cnf --user=mysql
```

De daemon draaide al toen je de initialisatie startte. Om ervoor te zorgen dat alle MySQL programma's gestopt worden kunnen we het *killall* commando gebruiken:

```
killall mysqld
```

Als het goed is, kun je nu de systeem-databases aanmaken, zonder vertraging.