

## Chapter 09: L.A.M.P. met PHPMyAdmin

<b>Doel</b>	Installeren van L.A.M.P. met PHPMyAdmin
<b>Benodigheden</b>	Debian 10.5 Server
<b>Tijdsduur</b>	Gemiddeld 4 lesuren



Theorie: Wat bedoelt men met L.A.M.P.

De term LAMP wordt in de informatietechnologie steeds vaker gebruikt. LAMP wordt gebruikt om een bepaalde omgeving in aan te duiden. Het verwijst naar de combinatie van de softwarepakketten Linux, Apache en MySQL. De laatste letter in de afkorting LAMP, de letter P, verwijst naar de programmeertalen PHP, Python en Perl.

De term LAMP-omgeving wordt vaak gebruikt om een set gratis softwarepakketten aan te duiden die in combinatie op een webserver worden gebruikt. De afkorting LAMP staat voor de volgende softwarepakketten:

Linux Apache MySQL PHP, Perl en/of Python

Al deze pakketten vallen onder de [GNU General Public License \(GPL\)](#). In deze licentie staat min of meer dat u mag doen wat u wilt met de software, op voorwaarde dat u ditzelfde recht doorgeeft aan anderen en de oorspronkelijke ontwikkelaars in uw software vermeldt. Wanneer u zich aan deze regels houdt mag u de software zelfs aanpassen en doorverkopen.

### Linux

Linux is een besturingssysteem of operating system dat zeer geschikt is voor servers op het internet. Linux is afgeleid van het besturingssysteem UNIX. Linux is gratis te downloaden en kan zo de kosten voor de aanschaf van een besturingssysteem drukken.

### Apache

Apache is een webserver of daemon voor Linux. Een webserver is het stuk software op een server dat het mogelijk maakt om webpagina's op de betreffende server te raadplegen. Elke webpagina die u op het internet kunt raadplegen, draait ergens op een webserver. Zo ook het artikel dat u nu leest.

### MySQL

MySQL is een relationeel databasemanagementsysteem (DBMS) voor Linux. In een dergelijk systeem kunt u databases of databanken aanmaken. Een database is niets anders dan een verzameling van gegevens die met elkaar verbonden staan. Via een gestructureerde taal kunt u de gegevens in de database opvragen. Deze taal heet SQL en staat voor Structured Query Language. Hier komt de naam MySQL vandaan.

### PHP, Perl of Python

PHP, Perl en Python zijn programmeertalen die u kunt gebruiken om uw website uit te breiden met programmacode. Zo kunt u bijvoorbeeld programmacode schrijven die de formulieren op uw website kan verwerken. Ook kunnen deze programmeertalen gebruikt worden om de gegevens uit de MySQL database op uw webpagina te tonen.

### Voordelen van de LAMP-omgeving

De LAMP-omgeving is voor veel bedrijven populair. De reden hiervan is dat Linux, Apache en MySQL in combinatie met PHP, Perl of Python gratis producten zijn. U mag al deze software gratis gebruiken voor commerciële doeleinden. De kosten voor software worden zo enigszins gedrukt.

Een andere reden om LAMP te gebruiken is het feit dat de verschillende softwarepakketten goed op elkaar zijn afgestemd. Apache laat zich gemakkelijk installeren op een Linux-omgeving. PHP heeft standaard ondersteuning voor MySQL.



### Stap 01 - Installeren van Apache2

1. We zorgen er eerst voor dat de Debian machine up-to-date is, login als root en typ daarna de volgende commando's in: `sudo apt-get update && sudo apt upgrade -y`
2. Daarna gaan we apache2 installeren: `sudo apt-get install apache2 -y`
3. Door de `-y`-optie installeert hij zonder vragen in 1 keer de apache2 webserver. Zoiest hebben we apache2 geïnstalleerd. Je kan controleren of het werkt door de volgende commando's in te typen: `sudo systemctl status apache2`

Je krijgt dan een soortgelijke uitkomst:

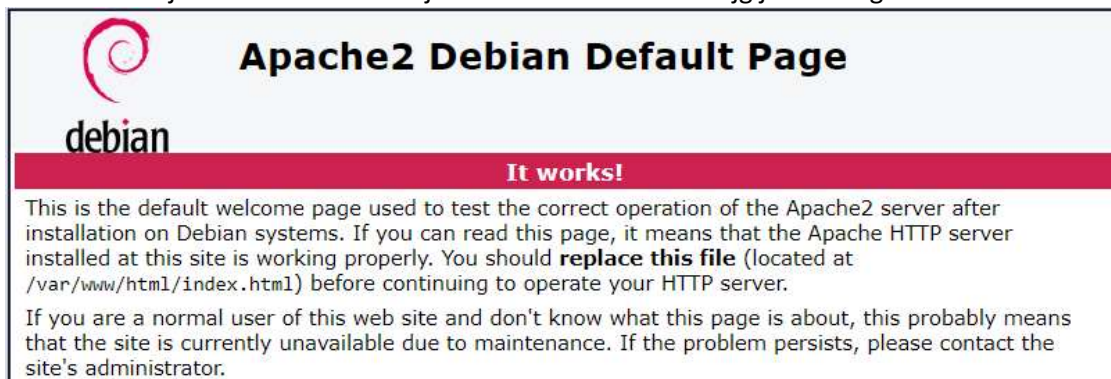
```
john@sl06035:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2020-11-21 22:19:32 CET; 36s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1879 (apache2)
    Tasks: 55 (limit: 4673)
   Memory: 22.5M
   CGroup: /system.slice/apache2.service
           └─1879 /usr/sbin/apache2 -k start
             └─1880 /usr/sbin/apache2 -k start
               └─1881 /usr/sbin/apache2 -k start

Nov 21 22:19:31 sl06035 systemd[1]: Starting The Apache HTTP Server...
Nov 21 22:19:32 sl06035 systemd[1]: Started The Apache HTTP Server.
```

⚠ **Let op:** Wanneer er geen **active (running)** staat kan je proberen om met de volgende commando's het nogmaals proberen:

```
sudo systemctl start apache2
sudo systemctl restart apache2
sudo systemctl enable apache2
sudo systemctl status apache2
```

Als je daarna nog geen **active (running)** ziet, volg je de bovenstaande stappen opnieuw uit. Wanneer je het IP-adres van je Debian machine in je webbrowser invoert krijg je het volgende te zien:

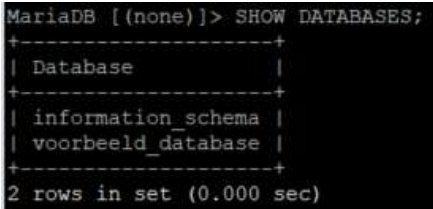


4. Met het commando `ip addr` krijg je het IP-adres van je Debian machine te zien.
5. Wanneer je de bovenstaande default page niet ziet, moeten we Apache2 toegang geven om naar buiten te treden door WWW Full (dit bezit alle poorten die we nodig zijn om een website te hosten (80,443/tcp)) `sudo ufw allow in "WWW Full"`



## Stap 02 - Installeren van MariaDB

Om een goede webserver te maken moeten we ook een database installeren. Binnen een database wordt alle data van de website geplaatst, hierdoor is een database server hoognodig. Wij gaan hiervoor MariaDB gebruiken. Binnen Linux kan je ook gebruik maken van MySQL, PostgreSQL, GNOME-DB en OracleDB. Wij gebruiken MariaDB omdat deze installatie stabiel is en na een aantal aanpassingen ook veilig is voor eventuele hackers.

1. Installeren van MariaDB: `sudo apt-get install mariadb-server -y`
2. Om de veiligheid te waarborgen gebruiken we het volgende commando:  
`sudo mysql_secure_installation`
3. Je krijgt na het invoeren van het commando een aantal vragen:  
Enter current password for root (enter for none): **<vul hier je root-wachtwoord in>**  
Switch to unix\_socket authentication [Y/n] **n**  
Change the root password? [Y/n] **n**  
Remove anonymous users? [Y/n] **Y**  
Disallow root login remotely? [Y/n] **Y**  
Remove test database and access to it? [Y/n] **Y**  
Reload privilege tables now? [Y/n] **Y**
4. Met het commando `sudo mariadb` kom je in de console van MariaDB terecht. Je ziet het CLI van `~#` wijzigen naar `MariaDB [(none)]>` Nu gaan we via de CLI een database aanmaken, hiervoor gebruiken we de volgende commando: `CREATE DATABASE <jouw_gebruikersnaam>`; Je ziet dat we het commando in hoofdletters hebben ingevoerd met daarna de naam. Het commando binnen MariaDB moet altijd in hoofdletters getypt worden en afgesloten worden met een punt-komma-teken (;).
5. Met de volgende 2 commando's maken we een gebruiker aan en geven we alle rechten over het zojuist aangemaakte voorbeeld-database aan een voorbeeld gebruiker met een specifiek wachtwoord.  
`CREATE USER '<jouw_gebruikersnaam>'@'localhost' IDENTIFIED BY 'wachtwoord';`  
`GRANT ALL PRIVILEGES ON <jouw_gebruikersnaam>.* TO '<jouw_gebruikersnaam>'@'localhost';`
- ⚠ **Let op:** Zorg ervoor dat je `<jouw_gebruikersnaam>` wijzigt naar je eigen `gebruikersnaam`.
6. Als laatste moet je met nog de database "doorspoelen" om de zojuist aangemaakte rechten toe te passen: `FLUSH PRIVILEGES;`
7. Nu heb je een database aangemaakt en deze met alle rechten aan een gebruiker gekoppeld. We gaan nu MariaDB als root verlaten en loggen in als de zojuist aangemaakte gebruiker: `exit`
8. Met de volgende commando loggen we in als de nieuwe gebruiker en met zijn rechten. Eerst gebruiken we het commando `mariadb` om de console van de database op te roepen, vervolgens gebruiken we de optie `-u` hierachter geef je de gebruiker waarmee je wilt inloggen op, doordat wij eerder hebben aangegeven dat we alleen met een wachtwoord in mogen loggen moeten we ook de optie `-p` gebruiken  
`sudo mariadb -u <jouw_gebruikersnaam> -p`
9. Vul nu je wachtwoord in, **let op: je ziet geen tekens verschijnen**. Je ziet na het inloggen dat het CLI weer gewijzigd is naar MariaDB `[(none)]>` dit betekend dat je succesvol ingelogd bent. We willen graag zien welke databases er allemaal aangemaakt zijn. Hiervoor gebruiken we het volgende commando: `SHOW DATABASES;` Je krijgt een lijst zoals hieronder weer wordt gegeven te zien:  

10. Verlaat MariaDB weer met de `exit` commando.



## Basic MySQL commando's:

Wijzig de rode woorden naar jouw eigen informatie (het commando in de rechter kolom is 1 commando)

Aanmelden mariadb	<code>mysql -u gebruikersnaam -p</code>
Aanmaken database	<code>CREATE DATABASE database;</code>
Aanmaken gebruiker	<code>CREATE USER 'gebruikersnaam'@'localhost' IDENTIFIED BY 'wachtwoord';</code>
Rechten toe kennen	<code>GRANT ALL PRIVILEGES ON database.* TO 'user'@'localhost';</code>
Rechten toe passen	<code>FLUSH PRIVILEGES;</code>
Databases bekijken	<code>SHOW DATABASES;</code>
Users bekijken	<code>SHOW USERS;</code>

⚠ **Let op:** Na elke wijziging moet je mariadb restarten. `sudo systemctl restart mysqld`

## Stap 03 - Installeren van PHP

1. Zonder een scripting of een programmeerserver hebben we niks aan de webserver. Als scripting of programmeerserver maken wij gebruik van PHP. Met PHP kan je, je code laten werken zodat je website nog soepeler verloopt en je meer kunt bereiken. Met de volgende commando installeren we de laatste bekende, stabiele versie van PHP:

```
sudo apt-get install php php-cli php-fpm php-json php-mysql libapache2-mod-php php-mbstring php-zip php-gd
```

⚠ **Let op:** Bovenstaande is 1 lange commando.

2. Normaal kijkt de apache server eerst naar index.html maar omdat wij PHP willen gebruiken i.p.v. HTML gaan we dit wijzigen naar index.php. Hiervoor gaan we een configuratiebestand van apache2 aanpassen: `sudo nano /etc/apache2/mods-enabled/dir.conf` Je krijgt het volgende te zien:

```
john@debian: ~  
GNU nano 7.2 /etc/apache2/mods-enabled/dir.conf  
#DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
```

Zorg ervoor dat index.php tussen DirectoryIndex en index.html komt te staan

```
john@debian: ~  
GNU nano 7.2 /etc/apache2/mods-enabled/dir.conf  
#DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
```

3. Sla het bestand op door de toets commando **<CTRL>+<X>** te gebruiken en deze te beantwoorden met **Y** daarna **<ENTER>** om het te bevestigen. Nu moeten we apache2 opnieuw starten: `sudo systemctl restart apache2`
4. Om te controleren of PHP goed werkt plaatsen we een bestand genaamd **info.php** in de root map van apache2: `sudo nano /var/www/html/info.php` Vul hier vervolgens het volgende stukje script in:  


```
<?php  
phpinfo();  
?>
```
5. Sla het bestand op door de toets commando **<CTRL>+<X>** te gebruiken en deze te beantwoorden met **Y** daarna **<ENTER>** om het te bevestigen. Open nu weer je webbrowser en vul na je IP-adres in vervolgt door **/info.php** bijvoorbeeld: **192.168.16.130/info.php** Als het goed is zie je welke versie is geïnstalleerd.




## Opdracht 02 – Installeren van PHPMyAdmin

PHPMyAdmin is een applicatie waarmee je je MySQL database online kunt beheren en raadplegen. Je kunt met PHPMyAdmin onder andere databases, tabellen en gegevensvelden aanmaken en verwijderen. Ook voor het maken van een back-up van je databases kun je in PHPMyAdmin terecht.

1. Voor we nu gaan beginnen met de installatie moeten we de server nog een keer updaten. Waarschijnlijk zijn er geen nieuwe pakketten beschikbaar maar voor de zekerheid doen we dit nog wel even:  
`sudo apt-get update && sudo apt-get upgrade`
2. We gaan ons eerst naar de map **temporary** (tijdelijk) verplaatsen: `cd /tmp`
3. Helaas zit PHPMyAdmin niet in een standaard repository hierdoor moeten we de ingepakte setup downloaden. Hiervoor gebruiken we het commando **wget** met deze commando kan je verschillende bestanden van diverse server via het internet downloaden. Het commando werkt als volgt: je begint met het commando vervolg door de weblink. Vul in je CLI de volgende commando in:  
`wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz`
4. Elk gedownload archief heeft een overeenkomstig .asc-bestand dat zijn unieke sleutelsignatuur bevat. Zodra beide bestanden zich in dezelfde map bevinden, kan de handtekening worden geverifieerd. Om de GPG-sleutel voor phpMyAdmin te verifiëren, download je de phpMyAdmin-sleutelhanger:  
`wget https://files.phpmyadmin.net/phpmyadmin.keyring`
5. Nadat de GPG-sleutel succesvol is gedownload moeten wij deze nog wel importeren:  
`sudo gpg --import phpmyadmin.keyring` Indien nodig installeer je eerst de optie gpg:  
`sudo apt-get install gpg`
6. Om PHPMyAdmin te kunnen benaderen maken we nu een map aan binnen de webserver:  
`sudo mkdir -p /var/www/html/phpmyadmin`
7. We hebben zojuist een map aangemaakt waar wij PHPMyAdmin gaan installeren. Hiervoor moeten we eerst het pakket uitpakken, wanneer we dit doen laten we hem direct in de juiste map plaatsen:  
`sudo tar xvf phpMyAdmin-latest-all-languages.tar.gz --strip-components=1`  
`-C /var/www/html/phpmyadmin`

 **Let op:** Bovenstaande commando is 1 lange commando, controleer dit goed!

8. Nu we alles op de juiste plek hebben staan, moet PHPMyAdmin nog wel geconfigureerd worden. Hiervoor pakken we de default configuration file van PHPMyAdmin zelf en passen deze een beetje aan. Eerst gaan we de configfile kopiëren én hernoemen. Dat doen we als volgt:  
`sudo cp /var/www/html/phpmyadmin/config.sample.inc.php`  
`/var/www/html/phpmyadmin/config.inc.php`

 **Let op:** Bovenstaande commando is 1 lange commando, controleer dit goed!

9. Open het bestand met een ander tekst editor dan wij in de eerste modules hebben gebruikt. **Nano** is een wat gebruikersvriendelijker CLI teksteditor dan VI. Als je wel VI wilt gebruiken dan vervang je het commando nano met vi. `sudo nano /var/www/html/phpmyadmin/config.inc.php`
10. We moeten nu opzoek naar de woord **blowfish** om een wachtwoord toe te voegen. Gebruik binnen Nano de volgende toetscombinatie: **<CTRL>+<W>** en type vervolgens **blowfish** in als zoekopdracht en druk daarna op **<ENTER>**. Kijk naar de afbeelding hieronder, als het goed is gegaan dan zie je hetzelfde.

```
$cfg['blowfish_secret'] = ''; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */
```





11. Vul tussen de 2 ' ' jouw wachtwoord in (zie afbeelding hierboven). Kan je geen wachtwoordzin bedenken? Dan mag je, aangezien het een schoolopdracht is, ook P@ssw0rd gebruiken als wachtwoord. Als het goed is komt het straks als volgt uit te zien:

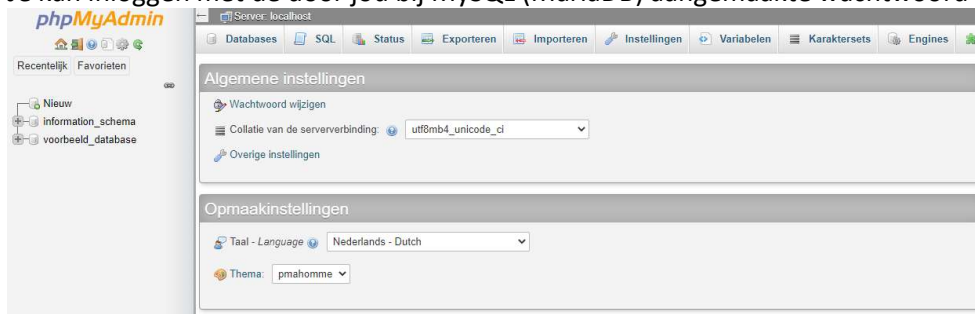
```
$cfg['blowfish_secret'] = 'Welkom01!'; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */
```

12. Gebruik na het aanpassen de volgende toetscombinatie <CTRL>+<X> om het bestand op te slaan, druk op Y wanneer hier om gevraagd wordt.
13. We gaan nu de juiste rechten toekennen aan het bestand, de eigenaar en de groep mogen alleen Lezen en Schrijven wie geen eigenaar of in de groep zit mag niks met het bestand doen:  
`sudo chmod 775 /var/www/html/phpmyadmin/config.inc.php`
14. Wanneer we nu kijken met het commando `ls -l` zien we dat jij de eigenaar en de groep bent van het bestand. Dit willen we niet, hiervoor maken we een algemene groep, eigenaar van het bestand:  
`sudo chown www-data:www-data /var/www/html/phpmyadmin`
15. Nu is de installatie van PHPMyAdmin als goed is voltooid. We moeten alleen nog wel Apache2 opnieuw opstarten. `sudo systemctl restart apache2`

16. Nu kunnen we PHPMyAdmin via een webbrowser van jouw werklaptop benaderen. Dit doe je doormiddel van het IP-adres van de Debian server in je adresbalk van een webbrowser in te vullen vervolgd met /phpmyadmin. Bijvoorbeeld: [192.168.2.100/phpmyadmin](http://192.168.2.100/phpmyadmin) of wanneer je de GUI hebt geïnstalleerd kan je de webbrowser openen van de Virtual Machine en localhost/phpmyadmin invullen in je adresbalk. Het resultaat zou in beide gevallen zichtbaar moeten zijn:



17. Je kan inloggen met de door jou bij MySQL (MariaDB) aangemaakte wachtwoord en gebruikersnaam



**Schrijf in je logboek** welke databases je al ziet staan in jouw PHPMyAdmin omgeving

