

Einführung in den LAMP-Stack

Sebastian Meisel

5. Mai 2025

Inhaltsverzeichnis

1	Was ist der LAMP-Stack?	1
2	Installation unter Ubuntu	1
2.1	Apache Webserver installieren	1
2.2	PHP und MariaDB installieren	1
2.3	MariaDB absichern	1
2.4	Test: PHP funktioniert?	1
3	Beispielanwendung: Einfaches Gästebuch	2
3.1	Datenbank anlegen	2
3.2	PHP-Datei erstellen: /var/www/html/gaestebuch.php	2
3.3	Datei bereitstellen	2
4	Umsetzung mit Podman und podman-compose	3
4.1	Voraussetzungen	3
4.2	Projektstruktur	3
4.3	Datei: podman-compose.yml	3
4.4	Datei: init.sql (Initialisiert die Datenbank)	3
4.5	Datei: html/gaestebuch.php	3
4.6	Start der Container	4
4.7	Hinweise	4
5	Anmerkungen	4

1 Was ist der LAMP-Stack?

LAMP steht für:

- **Linux** – das Betriebssystem
- **Apache** – der Webserver
- **MySQL** – die Datenbank (oder MariaDB)
- **PHP** – die serverseitige Scriptsprache

Der LAMP-Stack bildet die Grundlage für viele dynamische Webseiten und Webanwendungen.

2 Installation unter Ubuntu

2.1 Apache Webserver installieren

```
1 sudo apt update
2 sudo apt install apache2
3 sudo systemctl enable --now apache2
```

2.2 PHP und MariaDB installieren

```
1 sudo apt install php libapache2-mod-php mariadb-server php-mysql
2 sudo systemctl enable --now mariadb
```

2.3 MariaDB absichern

Das folgende Skript führt einen mit Fragen, durch die Absicherung der Datenbank.

```
1 sudo mysql_secure_installation
```

2.4 Test: PHP funktioniert?

```
1 echo "<?php␣phpinfo();␣?>" | sudo tee /var/www/html/info.php
```

Dann im Browser aufrufen: <http://localhost/info.php>

3 Beispielanwendung: Einfaches Gästebuch

3.1 Datenbank anlegen

```
1 CREATE DATABASE gaestebuch;
2 USE gaestebuch;
3 CREATE TABLE eintraege (
4     id INT AUTO_INCREMENT PRIMARY KEY,
5     name VARCHAR(100),
6     nachricht TEXT,
7     zeitstempel TIMESTAMP DEFAULT CURRENT_TIMESTAMP
8 );
```

3.2 PHP-Datei erstellen: /var/www/html/gaestebuch.php

```
1 <?php
2 $pdo = new PDO("mysql:host=localhost;dbname=gaestebuch", "root", "");
3
4 if ($_SERVER["REQUEST_METHOD"] == "POST") {
5     $name = $_POST["name"];
6     $nachricht = $_POST["nachricht"];
7     $stmt = $pdo->prepare("INSERT␣INTO␣eintraege␣(name,␣nachricht)␣VALUES␣(?,␣?)")
8     ;
9     $stmt->execute([$name, $nachricht]);
10 }
```

```

11 $eintraege = $pdo->query("SELECT_*_FROM_eintraege_ORDER_BY_zeitstempel_DESC")->
    fetchAll();
12 ?>
13 <!DOCTYPE html>
14 <html lang="de">
15 <head><meta charset="UTF-8"><title>Gästebuch</title></head>
16 <body>
17     <h1>Gästebuch</h1>
18     <form method="post">
19         Name: <input type="text" name="name"><br>
20         Nachricht:<br>
21         <textarea name="nachricht"></textarea><br>
22         <input type="submit" value="Eintragen">
23     </form>
24     <h2>Einträge:</h2>
25     <ul>
26         <?php foreach ($eintraege as $e): ?>
27             <li><strong><? = htmlspecialchars($e["name"]) ?></strong> (<? = $e["
                zeitstempel"] ?>):<br>
28                 <? = nl2br(htmlspecialchars($e["nachricht"])) ?></li>
29         <?php endforeach; ?>
30     </ul>
31 </body>
32 </html>

```

3.3 Datei bereitstellen

```

1 sudo cp gaestebuch.php /var/www/html/

```

Dann im Browser öffnen: <http://localhost/gaestebuch.php>

4 Umsetzung mit Podman und podman-compose

4.1 Voraussetzungen

- Installiertes Podman und podman-compose

```

1 sudo apt install podman podman-compose

```

4.2 Projektstruktur

```

lamp-podman/
├── podman-compose.yml
├── html/
│   └── gaestebuch.php
└── init.sql

```

4.3 Datei: podman-compose.yml

```

1 version: "3.8"
2 services:
3     web:

```

```

4     image: php:8.2-apache
5     ports:
6       - "8080:80"
7     volumes:
8       - ./html:/var/www/html
9     depends_on:
10      - db
11  db:
12    image: mariadb:10.6
13    environment:
14      MYSQL_ROOT_PASSWORD: example
15      MYSQL_DATABASE: gaestebuch
16    volumes:
17      - mariadb_data:/var/lib/mysql
18      - ./init.sql:/docker-entrypoint-initdb.d/init.sql:ro
19
20 volumes:
21   mariadb_data:

```

4.4 Datei: init.sql (Initialisiert die Datenbank)

```

1 CREATE TABLE eintraege (
2   id INT AUTO_INCREMENT PRIMARY KEY,
3   name VARCHAR(100),
4   nachricht TEXT,
5   zeitstempel TIMESTAMP DEFAULT CURRENT_TIMESTAMP
6 );

```

4.5 Datei: html/gaestebuch.php

(Siehe oben im Abschnitt "Beispielanwendung")

4.6 Start der Container

```

1 cd lamp-podman
2 podman-compose up -d

```

Dann im Browser aufrufen: <http://localhost:8080/gaestebuch.php>

4.7 Hinweise

- Die Datenbank wird beim ersten Start durch 'init.sql' automatisch eingerichtet.
- Änderungen am PHP-Code in 'html/' wirken sich sofort aus, da das Verzeichnis in den Container gemountet ist.
- Das System ist portabel und kann auch mit rootless Podman verwendet werden.

5 Anmerkungen

- Das Skript ist bewusst einfach gehalten und nicht gegen SQL-Injection oder XSS abgesichert – für den Produktiveinsatz müssten Schutzmaßnahmen ergänzt werden.
- Der Zugriff auf die Datenbank erfolgt als root ohne Passwort – dies sollte ebenfalls angepasst werden.

- Die Trennung von HTML und PHP wäre bei komplexeren Anwendungen empfehlenswert.
- Für produktive Umgebungen sollten separate Benutzer und sichere Passwörter verwendet werden.
- Für Mehrsprachigkeit oder Templates empfiehlt sich die Erweiterung mit Werkzeugen wie Twig oder einer kleinen PHP-Frameworklösung.