

Docker Compose

Setting up LAMP (Linux Apache MySQL PHP) Stack

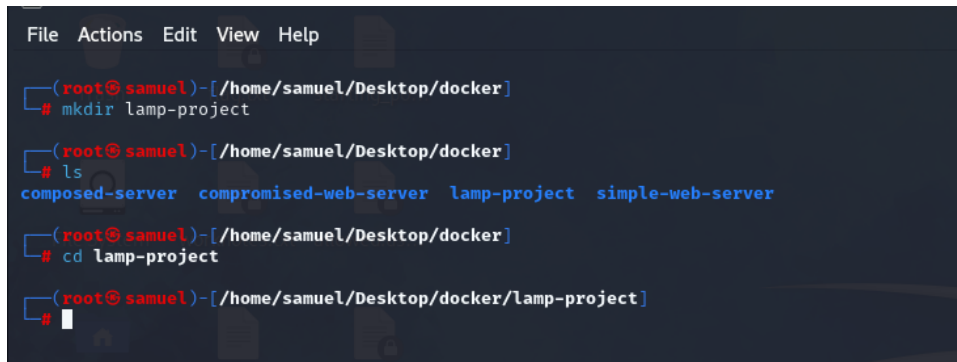
Introduction:

This documentation provides a comprehensive guide to setting up a LAMP (Linux, Apache, MySQL, PHP) stack environment using Docker Compose. This environment facilitates the development and testing of web applications locally, leveraging Docker to create isolated, reproducible environments. The guide covers the setup of the project directory structure, the creation of a Docker Compose file to define and run the LAMP stack services, and the development of a simple PHP script to interact with the MySQL database.

Task 1: Environment Setup and Docker Compose File

1. Project Directory Structure:

- Create a new folder named lamp-project.
- Inside lamp-project, create two subfolders: www for your PHP files and mysql for your database persistence.



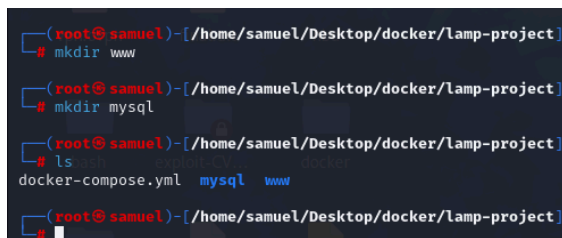
```
File Actions Edit View Help
(root@samuel)-[/home/samuel/Desktop/docker]
# mkdir lamp-project

(root@samuel)-[/home/samuel/Desktop/docker]
# ls
composed-server  compromised-web-server  lamp-project  simple-web-server

(root@samuel)-[/home/samuel/Desktop/docker]
# cd lamp-project

(root@samuel)-[/home/samuel/Desktop/docker/lamp-project]
#
```

○



```
(root@samuel)-[/home/samuel/Desktop/docker/lamp-project]
# mkdir www

(root@samuel)-[/home/samuel/Desktop/docker/lamp-project]
# mkdir mysql

(root@samuel)-[/home/samuel/Desktop/docker/lamp-project]
# ls
docker-compose.yml  mysql  www

(root@samuel)-[/home/samuel/Desktop/docker/lamp-project]
#
```

○

2. Docker Compose File Creation:

- Inside lamp-project, create a file named docker-compose.yml.
- Open docker-compose.yml in your preferred text editor.

○

```
(root@samuel) ~/home/samuel/Desktop/docker/lamp-project
touch docker-compose.yml
(gedit:4443) tepl-WARNING **: 13:16:13.668: Style scheme 'Kali-Dark' cannot be found, falling back to 'Kali-Dark' default style scheme.
(gedit:4443) tepl-WARNING **: 13:16:13.669: Default style scheme 'Kali-Dark' cannot be found, check your installation.
```

3. Service Definitions:

- Define a service named web using the php:7.4-apache image.
- Define a service named db using the mysql:5.7 image. Specify MYSQL_ROOT_PASSWORD, MYSQL_DATABASE, MYSQL_USER, and MYSQL_PASSWORD under environment.
- Define a service named admin using the phpmyadmin/phpmyadmin image, linking it to your db service for easy database management.

4. Networking and Volumes:

- Configure a custom network named lamp-network for your services to ensure they can communicate with each other.
- Set up volumes for both your www directory (to be linked with Apache's document root) and the mysql directory (to persist database data).

```
Open  *docker-compose.yml  Save
/home/samuel/Desktop/docker/lamp-project

1 version: '3.8'
2 services:
3   web:
4     image: php:7.4-apache
5     container_name: php_web
6     volumes:
7       - ./www:/var/www/html
8     ports:
9       - "8080:80"
10    networks:
11      - lamp-network
12  db:
13    image: mysql:5.7
14    container_name: mysql_db
15    environment:
16      MYSQL_ROOT_PASSWORD: rootpassword
17      MYSQL_DATABASE: lamp_db
18      MYSQL_USER: lamp_user
19      MYSQL_PASSWORD: userpassword
20    volumes:
21      - ./mysql:/var/lib/mysql
22    networks:
23      - lamp-network
24  admin:
25    image: phpmyadmin/phpmyadmin
26    container_name: php_admin
27    environment:
28      PMA_HOST: db
29    ports:
30      - "8081:80"
31    depends_on:
32      - db
33    networks:
34      - lamp-network
35  networks:
36    lamp-network:
37      driver: bridge
38
```

○

Task 2: Creating a PHP Script

1. PHP Script:

- Inside the www folder, create a file named index.php.
- Write a PHP script that attempts to connect to your MySQL database and displays a message on success or failure.

```
(root@samuel):~/home/samuel/Desktop/docker/lamp-project
# cd www

(root@samuel):~/home/.../Desktop/docker/lamp-project/www
# touch index.php

(root@samuel):~/home/.../Desktop/docker/lamp-project/www
# gedit index.php

(gedit:6510): tepl-WARNING **: 13:20:17.564: Style scheme 'Kali-Dark' cannot be found, falling back to 'Kali-Dark' default style scheme.

(gedit:6510): tepl-WARNING **: 13:20:17.564: Default style scheme 'Kali-Dark' cannot be found, check your installation.
```

```
index.php
/home/samuel/Desktop/docker/lamp-project/www
Save

1 $servername = "db"; // Same as your DB service name in docker-compose
2 $username = "lamp_user";
3 $password = "userpassword";
4 $database = "lamp_db";
5
6 // Create connection
7 $conn = new mysqli($servername, $username, $password, $database);
8
9 // Check connection
10 if ($conn->connect_error) {
11     die("Connection failed: " . $conn->connect_error);
12 }
13 echo "Connected successfully to the database.";
14 ?>
15 |
```

Task 3: Running and Testing Your LAMP Stack

1. Starting the LAMP Stack:

- Navigate to your project directory (lamp-project) and start your LAMP stack using docker-compose up -d.

```

➥ docker-compose up -d
Creating network 'lamp-project_lamp-network' with driver 'bridge'
Pulling web (ghcr.io:44-saache)...
42419999: Pulling from library/php
a8d3f5a3b41: Pull complete
15b7480b7f8: Pull complete
79a1c1a8f25: Pull complete
2f850409f5b: Pull complete
9a013ea38ac7: Pull complete
fa23a75ac4f: Pull complete
d140a2ed1a7: Pull complete
8d608f738ac: Pull complete
42c34c5ef8c: Pull complete
a550a40aa47: Pull complete
8892ca208b7: Pull complete
85a45aa9a9e: Pull complete
Digest: sha256:c50e0e6f78326737978d6aac81e011ec75d19b5f63ae3f2ee3ad
Status: Downloaded newer image for ghcr.io:44-saache
Pulling db (mysql:5.7)...
28ad4a0a499: Pull complete
1c5c3a5c94: Pull complete
a9f83a124cc: Pull complete
dc839a1215: Pull complete
6095a94b786: Pull complete
80a160a0a6e: Pull complete
ae7311cb779: Pull complete
fc9c9ef6f8: Pull complete
43a895336198: Pull complete
8a02c278f8a: Pull complete
d9fa4d8550b: Pull complete
Digest: sha256:530c9b950e0a153076cc5853eb4b515678baee01c815cbe472aad773b
Status: Downloaded newer image for mysql:5.7
Pulling admin (ghpmirror/ghpmirror)...
Latest: Pulling from ghpmirror/ghpmirror
a8a1d6c92a8: Pull complete
6978c9722d0: Extracting [====> ] 10.58MB/104.3KB
021a76b8ba2: Download complete
9a16a6cd987: Download complete
3f6c6135c3a: Download complete
7a27e9088b7: Download complete
d71699a02c3: Download complete
786c4cd2718: Download complete
78a1513383c: Download complete
15e1b4f4ec7: Download complete
53e16da4993: Download complete
77f69180c84: Download complete
68f05d43291: Download complete
99ca2716a8f7: Download complete
7a9c37c5c45: Download complete

```

```

Pulling admin (phpmyadmin/phpmyadmin):...
latest: Pulling from phpmyadmin/phpmyadmin
faef57eae888: Pull complete
989a1d6c052e: Pull complete
0705c9c2f22d: Pull complete
621478e043ce: Pull complete
98246dcca987: Pull complete
bfe98c155cb6: Pull complete
7a7c2e908867: Pull complete
d176994b625c: Pull complete
2d8ace6a2716: Pull complete
c70df516383c: Pull complete
15e1b44fe4c7: Pull complete
65e50d44e95a: Pull complete
77f68910bc0a: Pull complete
605dd3a6e332: Pull complete
99ce7188f07: Pull complete
74d64e32c5d5: Pull complete
ef5fc9928b9f: Pull complete
163f3256e112: Pull complete
Digest: sha256:67ba2550fd004399ab0b95b64021a88ea544011e566a9a1995180a3decb6410d
Status: Downloaded newer image for phpmyadmin/phpmyadmin:latest
Creating mysql_db ... done
Creating php_web ... done
Creating php_admin ... done

```

-
- Verify all containers are running properly with docker-compose ps.

```

(root@samuel)-[/home/samuel/Desktop/docker/lamp-project]
# docker-compose ps

```

Name	Command	State	Ports
mysql_db	docker-entrypoint.sh mysqld	Up	3306/tcp, 33060/tcp
php_admin	/docker-entrypoint.sh apac ...	Up	0.0.0.0:8081→80/tcp, :::8081→80/tcp
php_web	docker-php-entrypoint apac ...	Up	0.0.0.0:8080→80/tcp, :::8080→80/tcp

```

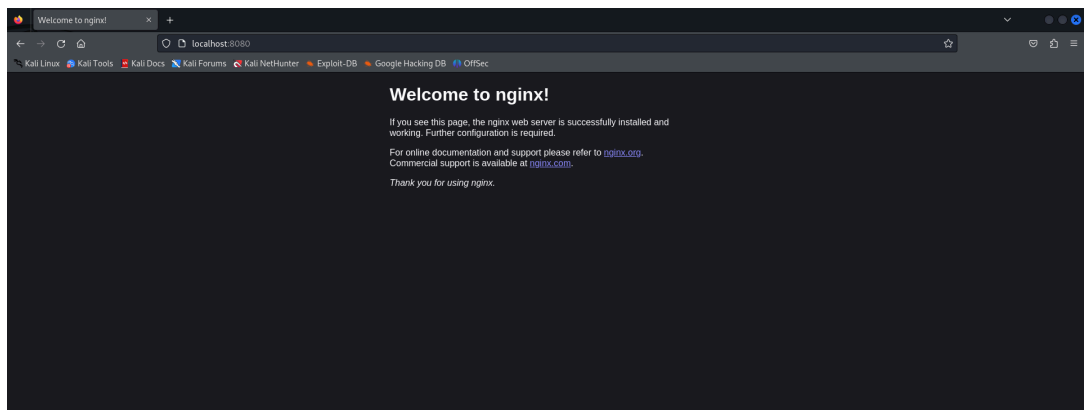
(root@samuel)-[/home/samuel/Desktop/docker/lamp-project]
#

```

○

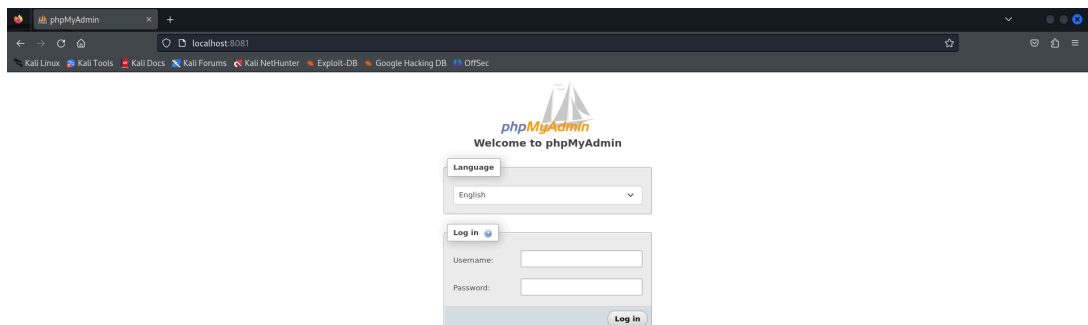
2. Testing:

- Access your PHP application by navigating to <http://localhost:8080> in your web browser.



○

- Access phpMyAdmin by navigating to <http://localhost:8081> in your web browser. Log in using your MySQL credentials.



○