

AWS EC2 LAMP Stack Deployment with Apache on Custom Port and MySQL Database Restoration via phpMyAdmin


– ASWIN VTK –

✓ TASK OVERVIEW (what we will do)

1. Create an EC2 instance with Ubuntu Operating system.
 2. Install Apache2, mysql, PHP and phpmyadmin.
 3. Web Server apache should run on port 6397.
 4. Web server running ports should be publicly accessible.
 5. Restore the given mysql backup using phpmyadmin, the database name should be "testmysql".
-

● STEP 1: Create EC2 Instance (Ubuntu)

In AWS Console

- **AMI:** Ubuntu Server 22.04 / 24.04
 - **Instance type:** t2.micro
 - **Key pair:** Create or select
 - **Security Group:**
 - SSH → Port **22** → My IP
 -  HTTP Custom → Port **6397** → **0.0.0.0/0**
-

● STEP 2: Connect to EC2 - Update system

```
ssh -i key.pem ubuntu@<EC2_PUBLIC_IP>
sudo apt update && sudo apt upgrade -y
```

● STEP 3: Install Apache, MySQL, PHP

```
sudo apt install -y apache2 mysql-server php libapache2-mod-php  
php-mysql php-cli php-curl php-zip php-gd php-mbstring php-xml  
php-bcmath
```

Enable Apache

```
sudo systemctl enable apache2  
sudo systemctl start apache2
```

● STEP 4: Change Apache Port to 6397

Edit Apache ports config

```
sudo nano /etc/apache2/ports.conf
```

Change: Listen 80

To: Listen 6397

Edit default site config

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

Change: <VirtualHost *:80>

To: <VirtualHost *:6397>

Restart Apache

```
sudo systemctl restart apache2
```

● STEP 5: Allow Port 6397 in Firewall

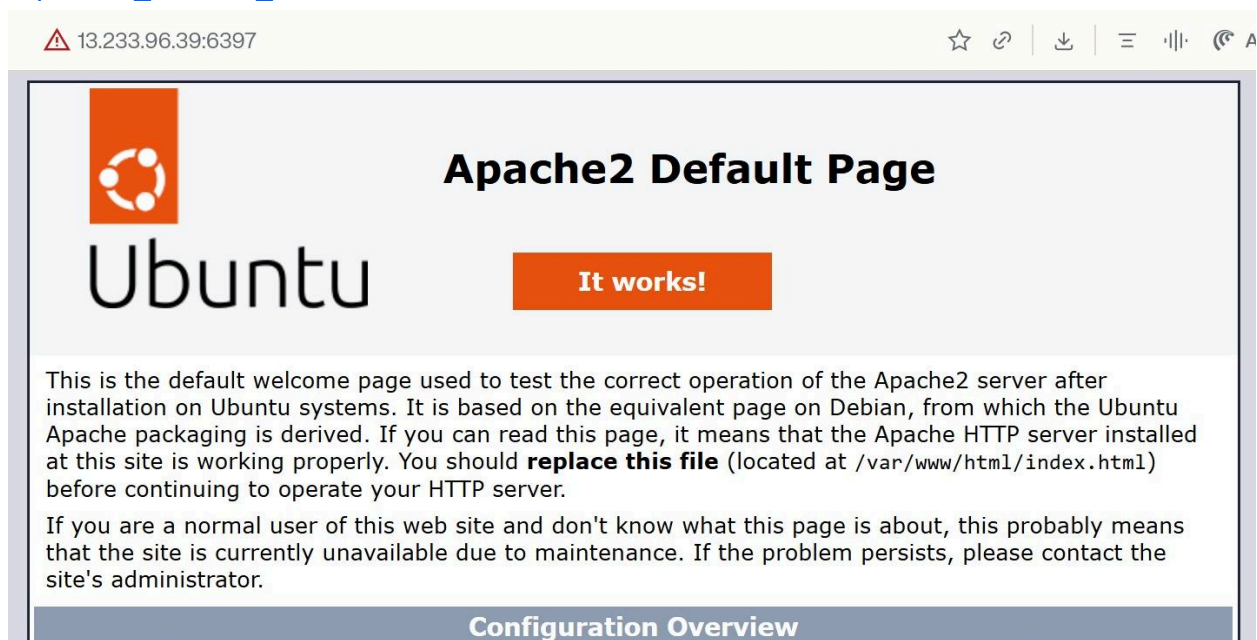
```
sudo ufw allow 6397
```

```
sudo ufw reload
```

● STEP 6: Test Apache

Open browser:

http://EC2_PUBLIC_IP:6397



● STEP 7: Install phpMyAdmin

```
sudo apt install -y phpmyadmin
```

When prompted:

- Web server → **apache2**
- Database config → **Yes**
- Set phpMyAdmin password

Enable phpMyAdmin in Apache

```
sudo ln -s /usr/share/phpmyadmin /var/www/html/phpmyadmin
sudo systemctl restart apache2
```

13.233.96.39:6397/info.php

PHP Version 8.3.6

System	Linux ip-172-31-9-148 6.14.0-1018-aws #18-24.04.1-Ubuntu SMP Mon Nov 24 19:46:27 UTC 2025 x86_64
Build Date	Jan 7 2026 08:40:32
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.3/apache2
Loaded Configuration File	/etc/php/8.3/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.3/apache2/conf.d
Additional .ini files parsed	/etc/php/8.3/apache2/conf.d/10-mysqld.ini, /etc/php/8.3/apache2/conf.d/10-opcache.ini, /etc/php/8.3/apache2/conf.d/10-pdo.ini, /etc/php/8.3/apache2/conf.d/15-xml.ini, /etc/php/8.3/apache2/conf.d/20-bz2.ini, /etc/php/8.3/apache2/conf.d/20-calendar.ini, /etc/php/8.3/apache2/conf.d/20-ctype.ini, /etc/php/8.3/apache2/conf.d/20-curl.ini, /etc/php/8.3/apache2/conf.d/20-dom.ini, /etc/php/8.3/apache2/conf.d/20-exif.ini, /etc/php/8.3/apache2/conf.d/20-ffi.ini, /etc/php/8.3/apache2/conf.d/20-fileinfo.ini, /etc/php/8.3/apache2/conf.d/20-ftp.ini, /etc/php/8.3/apache2/conf.d/20-gd.ini, /etc/php/8.3/apache2/conf.d/20-gettext.ini, /etc/php/8.3/apache2/conf.d/20-iconv.ini, /etc/php/8.3/apache2/conf.d/20-imagick.ini, /etc/php/8.3/apache2/conf.d/20-imap.ini, /etc/php/8.3/apache2/conf.d/20-intl.ini, /etc/php/8.3/apache2/conf.d/20-mbstring.ini, /etc/php/8.3/apache2/conf.d/20-mcrypt.ini, /etc/php/8.3/apache2/conf.d/20-mysqli.ini, /etc/php/8.3/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.3/apache2/conf.d/20-phar.ini, /etc/php/8.3/apache2/conf.d/20-posix.ini, /etc/php/8.3/apache2/conf.d/20-readline.ini, /etc/php/8.3/apache2/conf.d/20-shmop.ini, /etc/php/8.3/apache2/conf.d/20-simplexml.ini, /etc/php/8.3/apache2/conf.d/20-soap.ini, /etc/php/8.3/apache2/conf.d/20-sockets.ini, /etc/php/8.3/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.3/apache2/conf.d/20-sysvsem.ini, /etc/php/8.3/apache2/conf.d/20-sysvshm.ini, /etc/php/8.3/apache2/conf.d/20-tokenizer.ini

STEP 8: Create Database **testmysql**

```
sudo mysql
CREATE DATABASE testmysql;
EXIT;
```

STEP 9: Access phpMyAdmin

Open browser:

http://EC2_PUBLIC_IP:6397/phpmyadmin



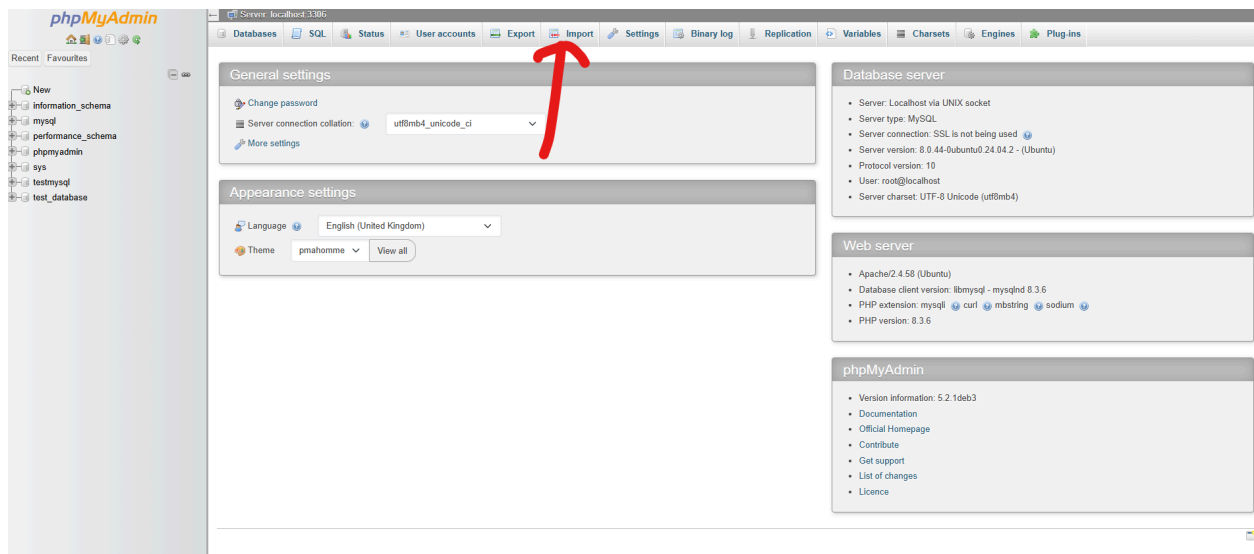
The image shows the phpMyAdmin login interface. At the top, there is a logo with the text "phpMyAdmin" and "Welcome to phpMyAdmin". Below the logo, there is a "Language" dropdown menu set to "English". Underneath, there is a "Log in" button with a plus icon. Below that, there are input fields for "Username:" and "Password:". At the bottom right, there is a "Log in" button.

Login using:

- Username: **root**
- Password: (phpMyAdmin password or MySQL user)

● STEP 10: Restore MySQL Backup

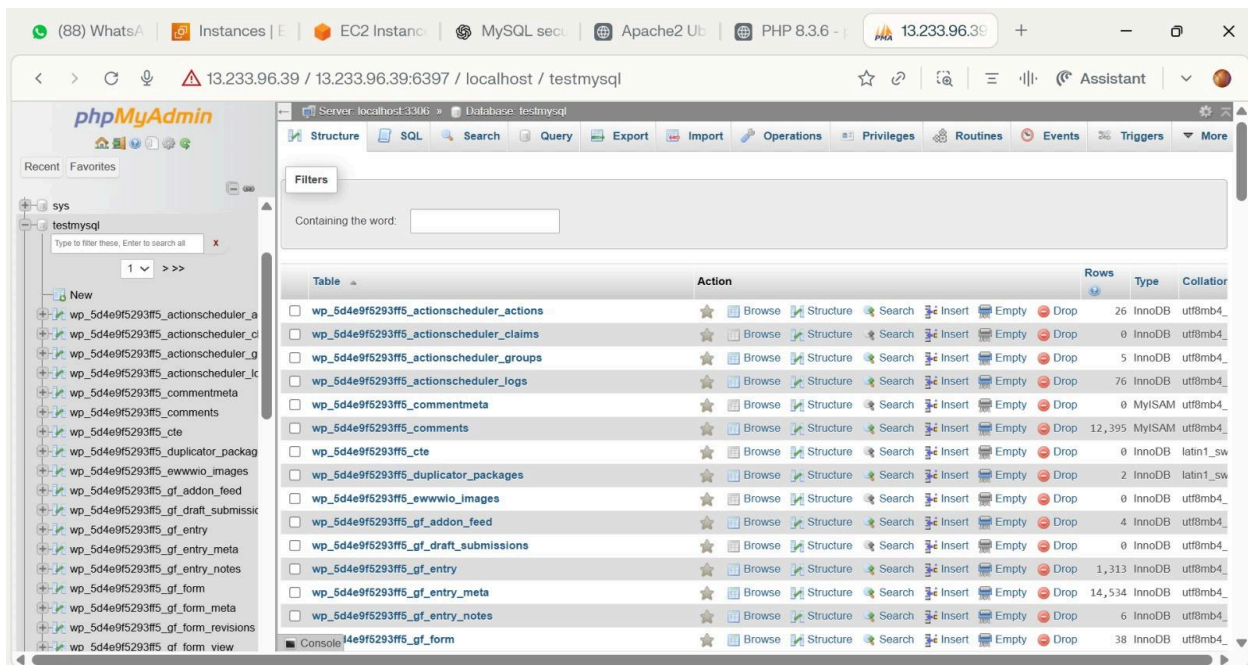
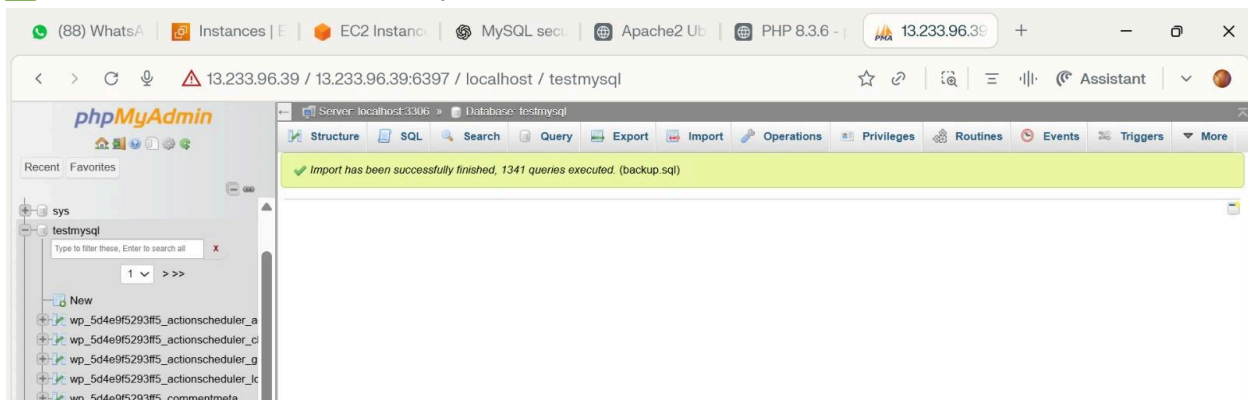
In phpMyAdmin:



The image shows the phpMyAdmin main interface. On the left, there is a sidebar with a tree view showing databases: information_schema, mysql, performance_schema, phpmyadmin, sys, testmysql, and test_database. The main area has a top navigation bar with tabs: Databases, SQL, Status, User accounts, Export, Import, Settings, Binary log, Replication, Variables, Charsets, Engines, and Plug-ins. A red arrow points to the "Import" tab. Below the navigation bar, there are three main sections: "General settings" (with a "Change password" link and a "Server connection collation" dropdown set to "utf8mb4_unicode_ci"), "Appearance settings" (with a "Language" dropdown set to "English (United Kingdom)" and a "Theme" dropdown set to "pmahomme"), and "Database server" (with server information: Server: localhost via UNIX socket, Server type: MySQL, Server connection: SSL is not being used, Server version: 8.0.44-0ubuntu0.24.04.2 - (Ubuntu), Protocol version: 10, User: root@localhost, Server charset: UTF-8 Unicode (utf8mb4)). Below these, there is a "Web server" section (with information: Apache/2.4.58 (Ubuntu), Database client version: libmysql - mysqlnd 8.3.6, PHP extension: mysql, curl, mbstring, sodium) and a "phpMyAdmin" section (with version information: 5.2.1deb3, Documentation, Official Homepage, Contribute, Get support, List of changes, Licence).

1. Select database **testmysql**
2. Click **Import**
3. Choose your **.sql** backup file
4. Click **Go**

✓ Database restored successfully



Common Error

Uploading sql a common error shown

“You probably tried to upload a file that is too large. Please refer to documentation for a workaround for this limit.”

It means PHP upload limits are too small for your MySQL backup file.

fix it properly so phpMyAdmin import works.

You probably tried to upload a file that is too large

- This is NOT a MySQL issue
- It is a PHP configuration limit

SOLUTION 1 : Increase PHP upload limits

- ♦ Step 1: Find your PHP version

```
php -v
```

You'll see something like:

PHP 8.1.x

So config path will be: </etc/php/8.1/apache2/php.ini>

- ♦ Step 2: Edit php.ini

```
sudo nano /etc/php/*/apache2/php.ini
```

Find and change these values:

```
upload_max_filesize = 256M
post_max_size = 256M
max_execution_time = 300
max_input_time = 300
memory_limit = 512M
```

SOLUTION 2 : Alternative

- ✓ No nano
- ✓ No need to find line
- ✓ Safe and fast

Open php.ini directly at needed lines:

```
sudo sed -i 's/upload_max_filesize = .*/upload_max_filesize = 256M/' /etc/php/*/apache2/php.ini
sudo sed -i 's/post_max_size = .*/post_max_size = 256M/' /etc/php/*/apache2/php.ini
sudo sed -i 's/max_execution_time = .*/max_execution_time = 300/' /etc/php/*/apache2/php.ini
sudo sed -i 's/max_input_time = .*/max_input_time = 300/' /etc/php/*/apache2/php.ini
sudo sed -i 's/memory_limit = .*/memory_limit = 512M/' /etc/php/*/apache2/php.ini
```

```
sudo systemctl restart apache2
```