**Step 1: Update Your System**

First, ensure your package lists and installed packages are updated.

bash

Copy code

sudo apt update && sudo apt upgrade -y

**Step 2: Install Apache Web Server**

Install the Apache2 HTTP Server.

bash

Copy code

sudo apt install apache2 -y

After installation, enable and start the Apache service.

bash

Copy code

sudo systemctl enable apache2 sudo systemctl start apache2

To check if Apache is running, you can navigate to **http://your\_server\_ip/** in your web browser. You should see the default Ubuntu Apache web page.

**Step 3: Install MySQL**

Install the MySQL database server.

bash

Copy code

sudo apt install mysql-server -y

After installation, run the security script that comes with MySQL which will remove some unsafe defaults.

bash

Copy code

sudo mysql\_secure\_installation

Follow the on-screen prompts to configure your MySQL installation, including setting a root password.

**Step 4: Install PHP 8.2**

Ubuntu’s default repositories might not have PHP 8.2, so you should get it from a third-party repository, such as the ondrej/php repository.

bash

Copy code

sudo apt install software-properties-common -y sudo add-apt-repository ppa:ondrej/php -y sudo apt update

Install PHP 8.2 along with common extensions.

bash

Copy code

sudo apt install php8.2 libapache2-mod-php8.2 php8.2-mysql php8.2-cli php8.2-curl php8.2-zip php8.2-gd php8.2-mbstring php8.2-xml -y

To verify PHP installation:

bash

Copy code

php -v

**Step 5: Configure Apache to Use PHP**

Confirm that Apache picks up PHP by default:

bash

Copy code

sudo a2enmod php8.2 sudo systemctl restart apache2

**Step 6: Install phpMyAdmin**

Install phpMyAdmin to manage MySQL through a web interface.

bash

Copy code

sudo apt install phpmyadmin -y

During installation, you will be prompted to choose the web server that should be automatically configured to run phpMyAdmin. Select "apache2".

Set up the database for phpMyAdmin with dbconfig-common; select 'Yes' and provide your MySQL root password when asked.

**Step 7: Enable PHP extensions**

Enable any necessary PHP extensions.

bash

Copy code

sudo phpenmod mysqli mbstring sudo systemctl restart apache2

**Step 8: Access phpMyAdmin**

You can now access phpMyAdmin by going to **http://your\_server\_ip/phpmyadmin** in your web browser. Use the MySQL credentials to log in.

**Step 9: Secure Your phpMyAdmin**

Create an Apache configuration file for phpMyAdmin to enhance security (like URL obfuscation).

bash

Copy code

sudo nano /etc/apache2/conf-available/phpmyadmin.conf

Add an alias line under the existing alias:

apacheconf

Copy code

Alias /your-secret-url /usr/share/phpmyadmin

Restart Apache to apply changes:

bash

Copy code

sudo systemctl restart apache2

**Pin PHP Version**: You can pin the PHP version to 8.2. This action tells **apt** to prioritize version 8.2 over newer versions when installing or updating packages.

* Create a preference file:

bash

Copy code

sudo nano /etc/apt/preferences.d/php

* Add the following lines to pin the version:

plaintext

Copy code

Package: php\* Pin: version 8.2\* Pin-Priority: 1000

1. **Reinstall PHP 8.2**: If PHP 8.3 was installed, you might need to uninstall it and reinstall PHP 8.2 packages, along with the specific extensions needed by phpMyAdmin.

bash

Copy code

sudo apt remove php8.3\*

sudo apt install php8.2 php8.2-xml php8.2-mbstring php8.2-mysql libapache2-mod-php8.2 php8.2-cli php8.2-curl php8.2-zip php8.2-gd -y

1. **Check phpMyAdmin Dependencies**: Before reinstalling phpMyAdmin, check if it explicitly requires PHP 8.3. If not, you can proceed to install or reinstall it.

bash

Copy code

sudo apt install phpmyadmin -y

1. **Set Up Virtual Host**: Create or modify a Virtual Host file for your Laravel application.

bash

Copy code

sudo nano /etc/apache2/sites-available/localhost.conf

Here’s an example configuration:

apache

Copy code

<VirtualHost \*:8080>

ServerAdmin webmaster@localhost

DocumentRoot /var/www/html

ServerName 192.168.0.2

ServerAlias www.192.168.0.2

<Directory /var/www/html/public>

    AllowOverride All

     Require all granted

</Directory>

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

1. **Enable the Site and Rewrite Module**:

bash

Copy code

sudo a2ensite your-laravel-app.conf sudo a2enmod rewrite sudo systemctl restart apache2

sudo nano apache2.conf

Then write this there after port.conf

ServerName localhost