# **Setting up a Debain 10 LAMP Server for PHP Web Development**

**Introduction:** In today's digital age, the demand for robust and scalable web applications continues to grow exponentially. Linux, renowned for its stability, security, and flexibility, stands as a preferred choice for hosting web servers. Developing a web server on Linux involves harnessing the power of open-source tools, leveraging the kernel's capabilities, and optimizing performance to meet diverse application needs.

**Motivation:** This Linux web server project is familiar with the Basic knowledge of MySQL; I have knowledge in it. From a learner perspective, choosing a Linux web server development project offers a blend of practical learning, career readiness, exposure to open-source principles, scalability exploration, community support, cost-effectiveness, and personal interest fulfillment. These factors collectively make it a compelling choice among various project options, particularly for those aiming to build foundational skills that are directly applicable to real-world IT environments and job opportunities.

**Role and responsibilities:**

* I am taking care of installing and configuring of MySQL and MariaDB

**What area of work?**

* In PHP web development with databases, the key areas of work include:

**1.** **Database Design**: Designing the structure of the database tables, establishing relationships between entities (e.g., one-to-one, one-to-many, many-to-many), and optimizing schema for efficient data storage and retrieval.

**2. Querying**: Writing SQL queries to interact with the database, including CRUD operations (Create, Read, Update, Delete) to manipulate data, retrieve information based on specific criteria (SELECT statements), and manage transactions to ensure data integrity.

**3. Data Validation and Sanitization**: It's crucial to validate and sanitize user input to prevent SQL injection attacks and ensure data integrity. PHP developers implement validation rules and sanitize data before inserting or updating it in the database.

**4. ORM (Object-Relational Mapping)**: Some PHP frameworks (like Laravel with Eloquent ORM) provide ORM capabilities, which abstract database interactions into object-oriented methods. Developers work with ORM to simplify database operations and improve code readability.

**5. Performance Optimization**: PHP developers optimize database performance by writing efficient SQL queries, using indexes, caching query results, and optimizing database configuration settings.

**6. Database Security**: Ensuring database security involves implementing authentication and authorization mechanisms, encrypting sensitive data, and following best practices to protect against data breaches.

**7. Database Migration and Maintenance**: Developers perform database migrations to update schema changes, manage database backups, and handle data migration between environments (e.g., development to production).

**Purpose of your work?**

* In PHP web development, working with databases serves several fundamental purposes that are essential for building dynamic and data-driven web applications:
* Data Storage:
* Data Retrieval:
* Data Manipulation
* Data Validation and Security:
* Scalability and Performance:
* Reporting and Analytics

**Feasibility study:** A feasibility study in a project context assesses the practicality and viability of a proposed project or initiative. It aims to determine whether the project is technically, financially, and operationally feasible before committing resources to its development.

* **Software required?**
* **MariaDB** is an open-source relational database management system (RDBMS) that originated as a fork of MySQL, designed to remain free and open-source under the GNU General Public License.

**Inputs?**

* In PHP web development, when working with databases, inputs generally refer to the data that comes from various sources and is processed by the application before being stored in the database or used for other purposes
* user inputs
* external inputs
* system inputs
* database inputs

**Outputs?**

In PHP web development, outputs refer to the data that is retrieved from the database and presented to users or used within the application.

1. **Displaying Data**: Outputs often involve displaying data to users through web pages or APIs. Examples include:

* Displaying user profiles
* Showing product listings
* Presenting search results

1. **Sending Data to Other Systems**: Outputs may involve sending data to other systems or services. This can include:

* Sending data to external APIs (e.g., for payment processing, notifications)
* Integrating with third-party services

1. **Processing Outputs**: Outputs can also involve processing data within the application itself. This includes:

* Aggregating and summarizing data for reports or analytics
* Formatting data for export (e.g., generating CSV files)

**LAMP Stack Web Server**

**Introduction:**

The LAMP Stack Web Server is also known as Linux Web Server. It is an open-source web development platform used for building dynamic websites and web applications. LAMP stands for Linux, Apache, MySQL, and PHP/Perl/Python. These are the four main components required to setup a Web Server.

**Components of the LAMP Stack:**

**Linux (Operating System)**

• Linux servers as the foundation of the LAMP stack. It Provides the Operating System on which the other components are installed and run.

• Linux distributions for LAMP Stack deployment include Ubuntu and Debian.

Apache (Web Server Software)

• Apache is a widely used open-source web server software that serves web content to clients over the internet.

• Apache handles HTTP requests from web browsers and delivers web pages.

• It extends various modules for extending its functionality.

**MySQL (Database):**

• It is a software which is used to maintain and manage the Database.

• It provides security and authorization are the important features of the DBMS.

• MySQL is commonly used in web applications to store user information and other data.

PHP/Per/Python (Server Side Shifting Language)

PHP, Perl, Python are the server-side scripting languages used for dynamic web development

• These languages enable developers to create interactive and dynamic web pages that respond to user input.

• PHP is mostly commonly used language in LAMP Stack.

Setting up a LAMP Stack Web Server

**Step-1: Install Linux**

• Before we are going to further you must need a Linux or One of Linux Distributions

Ubuntu by using this link https://ubuntu.com/download.

**Step-2: Installing MySQL/MariaDB**

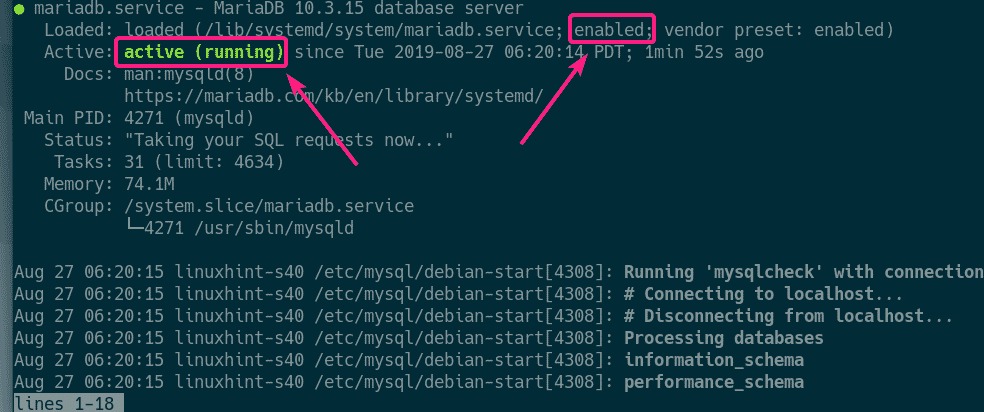
• Now, install MariaDB server and client packages from the official package repository of Debian 10 with the following command:

C:\Users\Nanda Kumar\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\7F6FFAA6BB0B408017B62254211691B5\WhatsApp Image 2024-05-09 at 21.18.03_b91726d9.jpg

conform your Installation, Press Y to confirm or n to stop the process.

• If you want to check the status of MariaDB server is running or not Enter the following command

C:\Users\Nanda Kumar\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\5FD0B37CD7DBBB00F97BA6CE92BF5ADD\WhatsApp Image 2024-05-09 at 21.20.12_135dbdd0.jpg

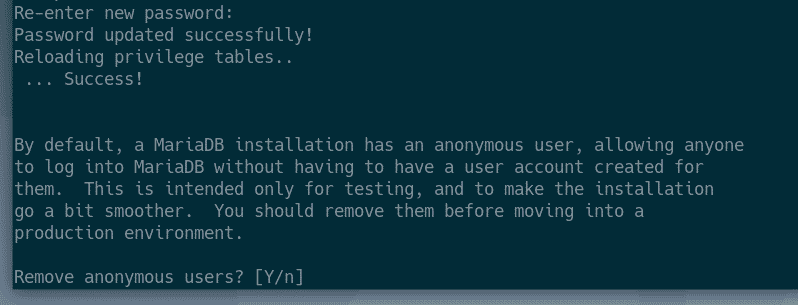
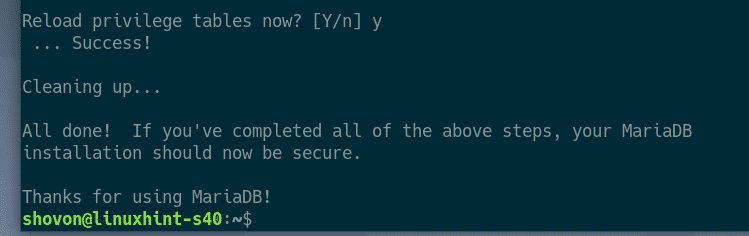


* If in any case, **MariaDB** service is not running, then start the service with the following command:

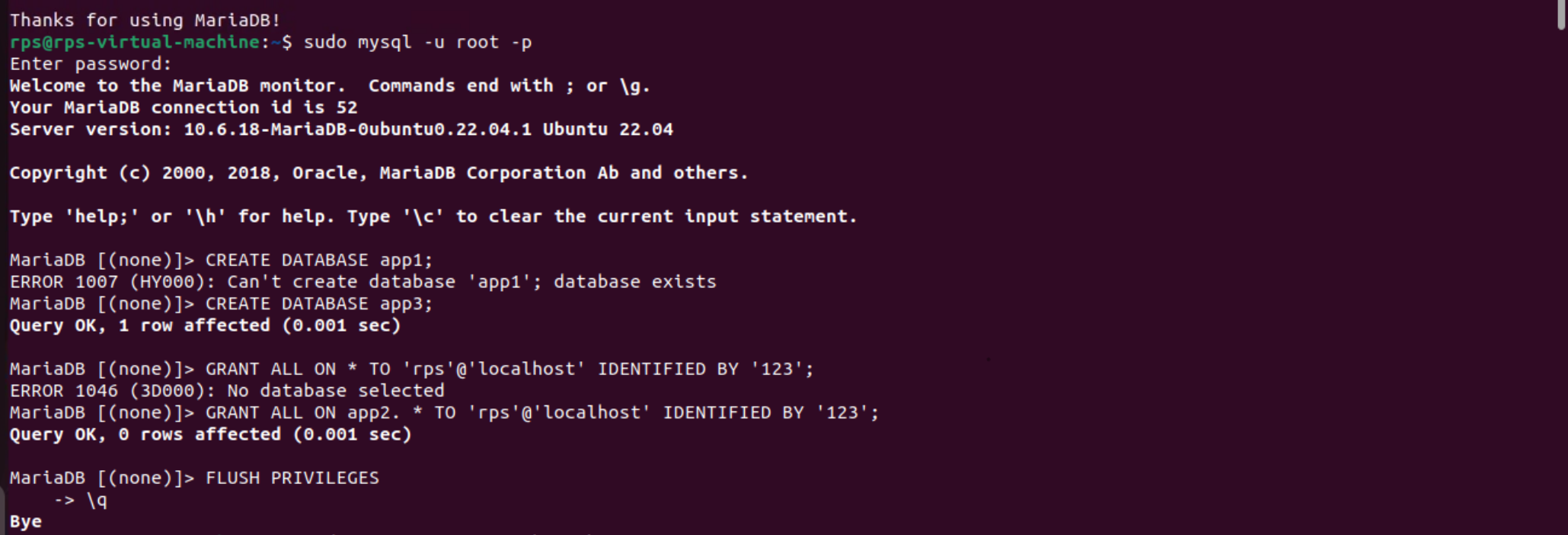
$ sudo systemctl start mariadb

* Now, run the following command to set a **root** password for MariaDB:

$ sudo mysql\_secure\_installation

* We can create a new password and Press **Y** and then press **<Enter>** to remove anonymous users.
* 
* MariaDB should be configured.
* 

### **Creating New MySQL/MariaDB Users and Databases:**

* Now, you have to create a new user and a database for your web application.
* Login to MariaDB shell with the following command:
* $ sudo mysql -u root -p
* 

**Step-3: Install Apache2 and PHP**

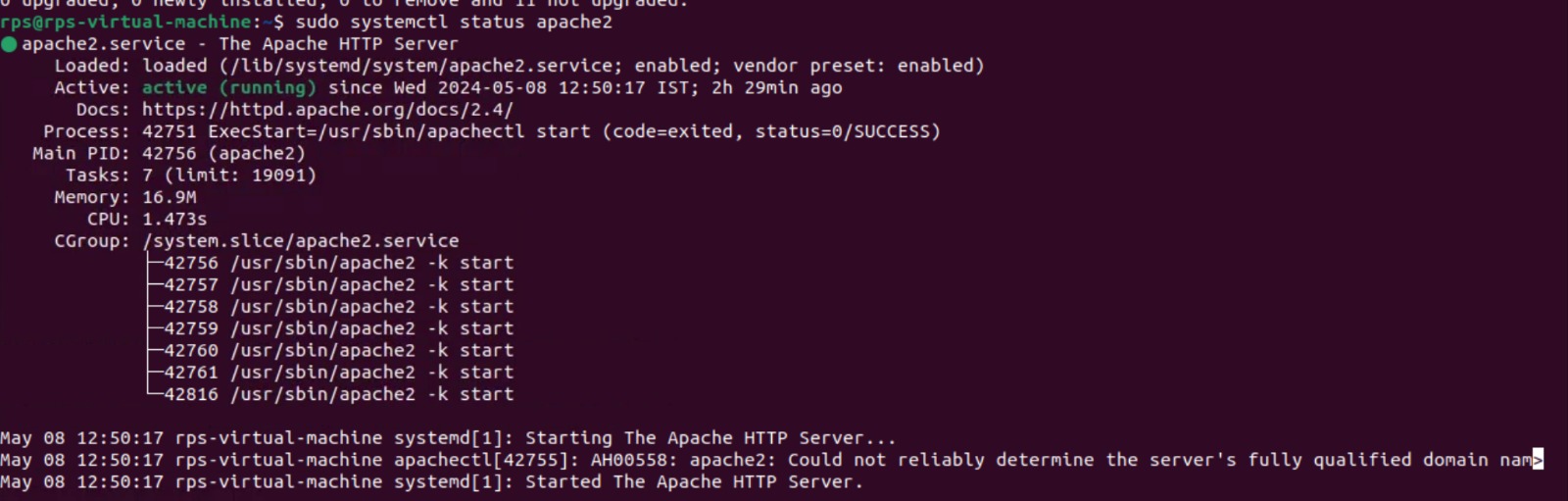
* install To Apache2 and PHP in your Operating System use the following command and setup according to it.

$ Sudo apt install apache2 php

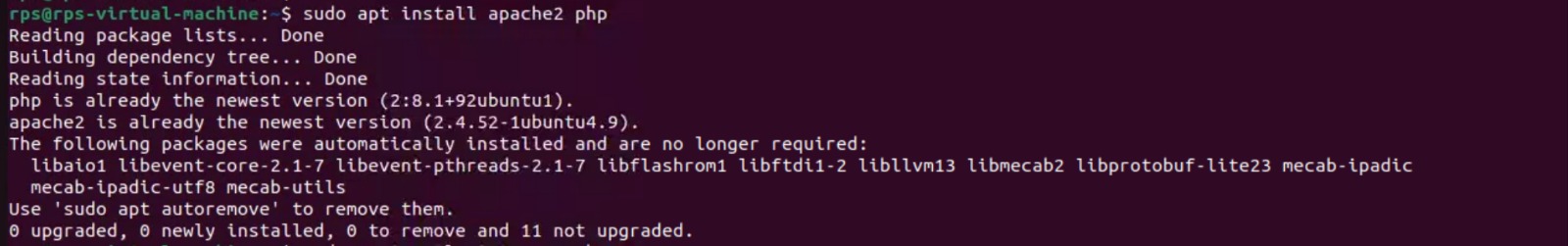
* Apache 2 web server and PHP should be installed.
* Now, check whether **apache2** service is running with the following command:

$ Sudo systemctl status apache2

As you can see, **apache2** service is running. It’s also **enabled** to start automatically on system boot



* After the installation done check the status of that server is running or not by using the following command



* To install the most common PHP extensions/libraries, run the following command:



* Then, Again Restart the Apache2 Server by using following command

C:\Users\Nanda Kumar\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\069059B7EF840F0C74A814EC9237B6EC\WhatsApp Image 2024-05-09 at 21.36.58_e3d6a465.jpg

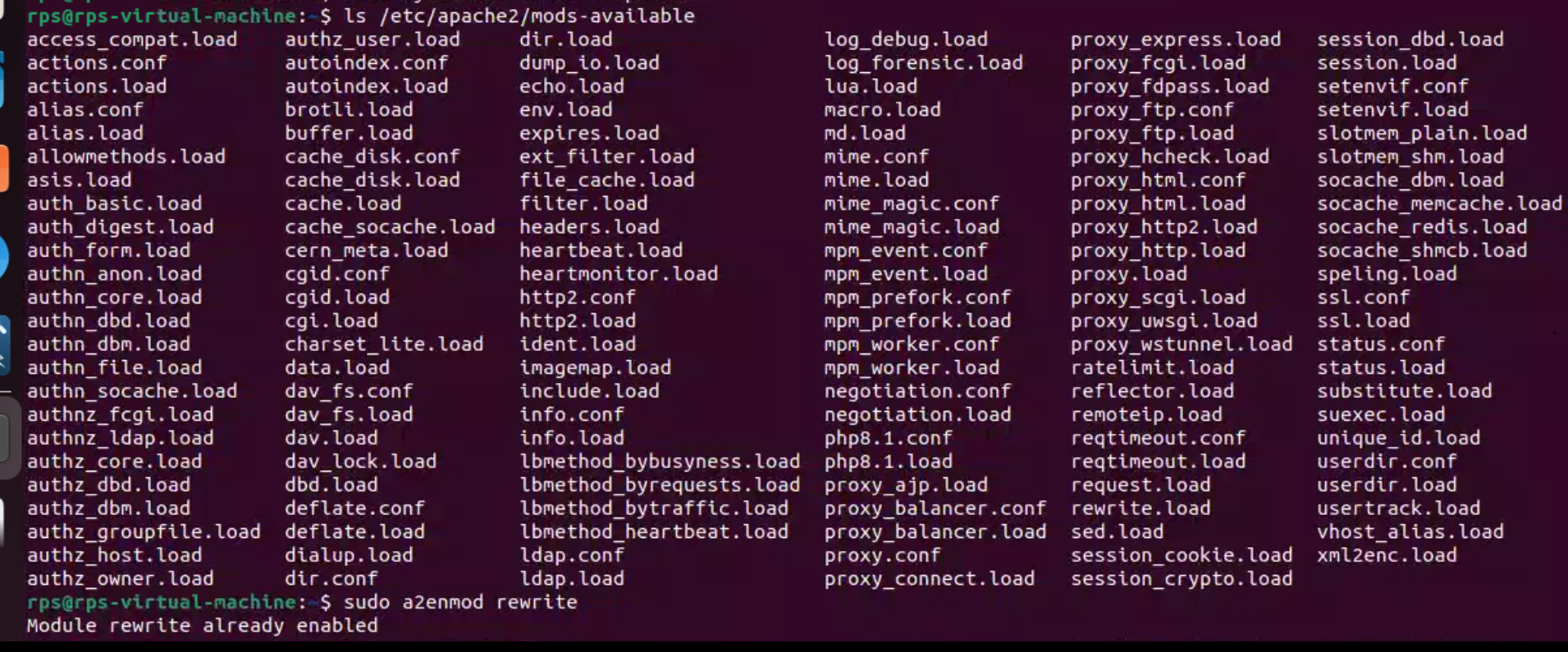
* To change the Apache run user, edit /etc/apache2/euvvars configuration file with the following command:

**Enabling/Disabling Apache Modules:**

* Apache web server comes with a lot of modules. You can enable or disable them as you need.
* To list all the available Apache 2 modules, run the following command

*$* ls /etc/apache2/mods-available

* As you can see, all the available Apache 2 modules are listed.

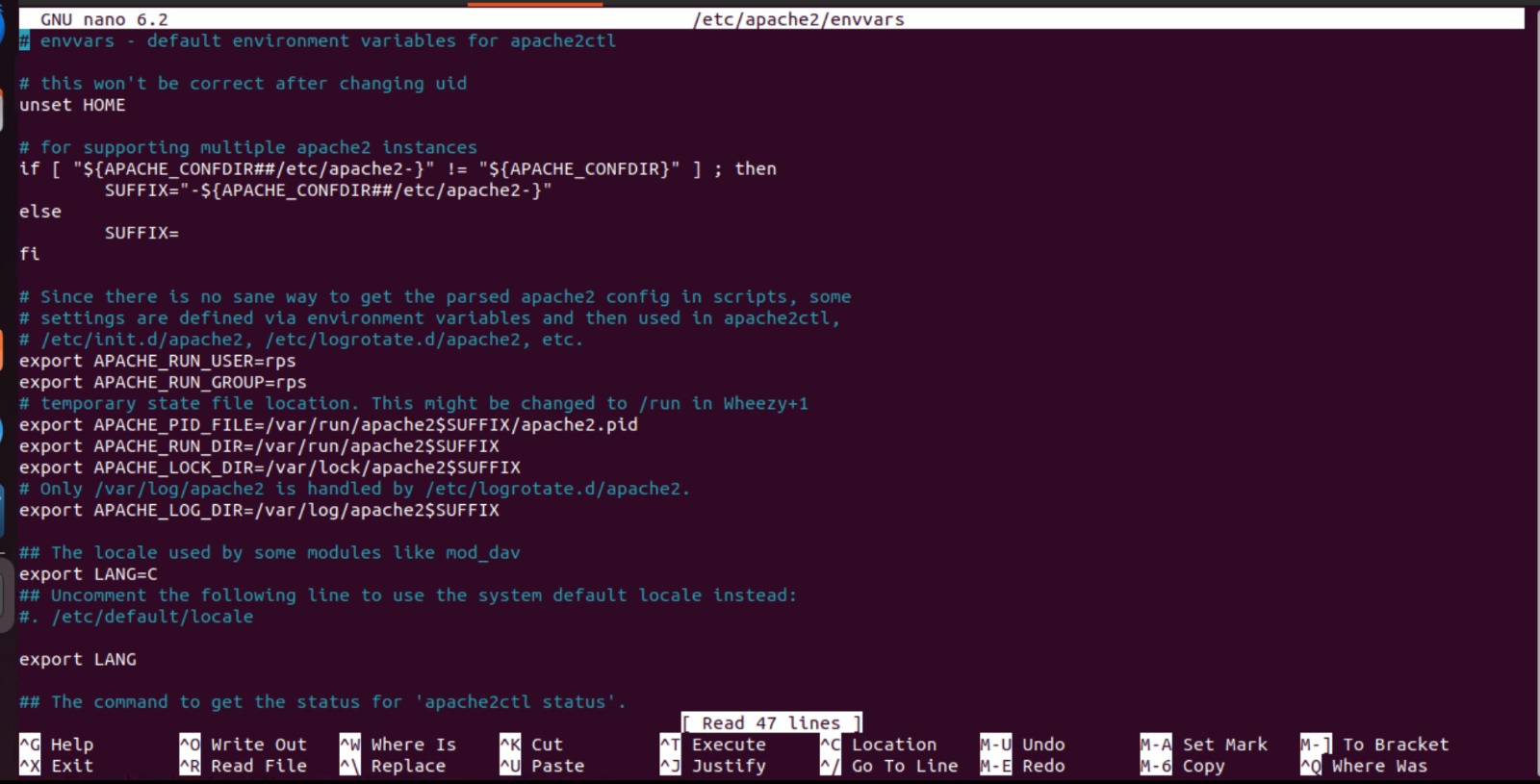


### **Changing Apache Run User:**

The default Apache run user on Debian 10 is **www-data** and the default web root directory is **/var/www/html**. So, as an ordinary user, you won’t be able to create files/directories, or modify existing files/directories in the web root directory. As you’re setting up a development LAMP server, this is not what you want. To solve this problem, you should change the Apache run user to your login user and change the owner and group of the webroot **/var/www/html** to your login user.

C:\Users\Nanda Kumar\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\76DC611D6EBAAFC66CC0879C71B5DB5C\WhatsApp Image 2024-05-09 at 21.38.45_946f233c.jpg

* Modify-the APACHE\_ RUN\_USER and APACHE\_ RUN\_GROUP environment variables.



* Now, instead of PS you have to give your root user. In this case my user is rps.
* Now, change the owner and group of the /var/www/html directory to the username of Your login user with the following command:

C:\Users\Nanda Kumar\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\42A0E188F5033BC65BF8D78622277C4E\WhatsApp Image 2024-05-09 at 21.46.47_94dda971.jpg

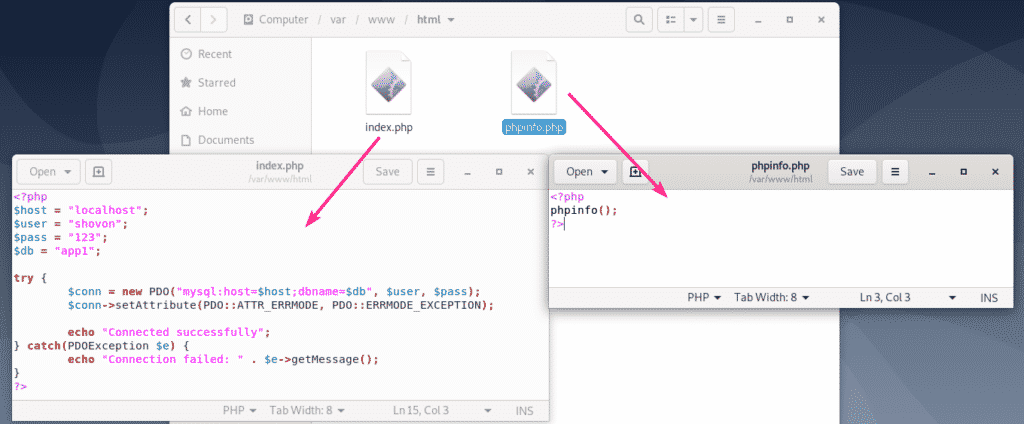
* I have created 2 PHP scripts index.php and phpinfo.php in the web root /var/www/html.

### **Testing LAMP Server:**

I have created 2 PHP scripts **index.php** and **phpinfo.php** in the webroot **/var/www/html**.

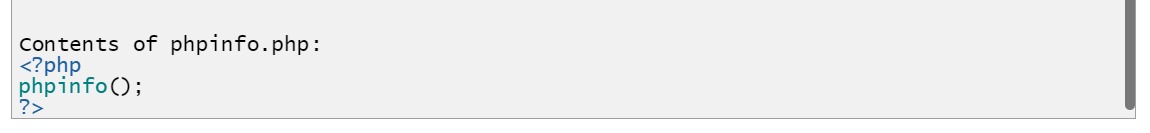
**Contents of index.php:**



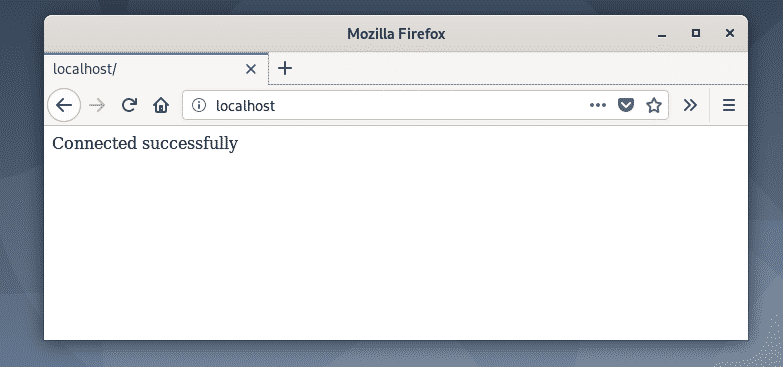


Now, you should be able to access the PHP scripts from your browser as you can see in the screenshot below.

**Contents of phpinfo.php:**

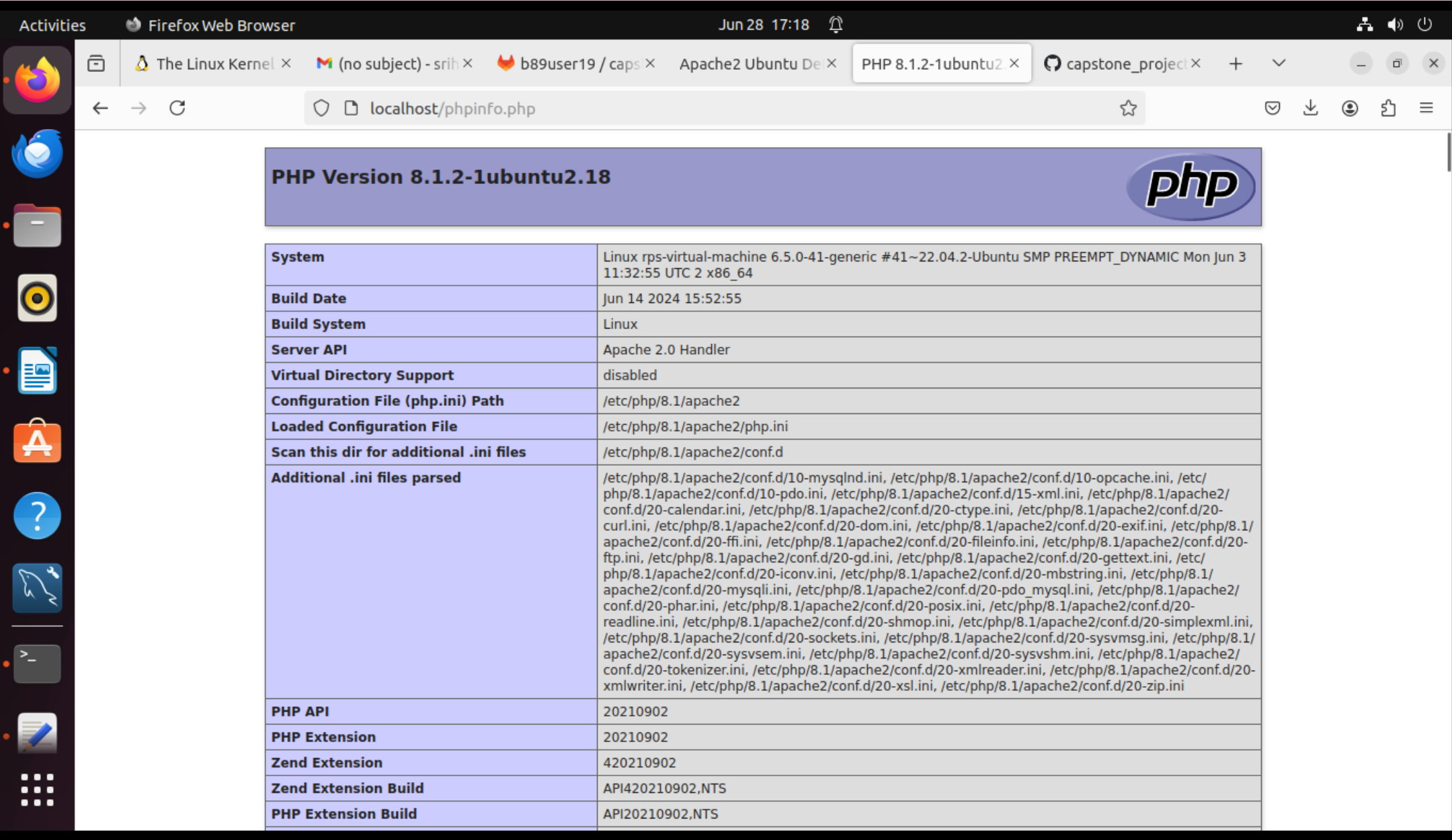


[**http://localhost/**](http://localhost/)**:**





[**http://localhost/phpinfo.php**](http://localhost/phpinfo.php)



So, that’s how you setup a Debian 10 LAMP server for PHP web development.

**conclusion:**

Linux web server development offers unparalleled advantages in terms of performance, security, and flexibility, making it the preferred choice for hosting mission-critical web applications. By leveraging Linux's robust ecosystem and adhering to best practices in development and deployment, developers can build scalable and reliable web servers that meet the evolving demands of modern digital landscapes.