Install LAMP Server Packages

Before we install the LAMP packages, we need to update Raspbian:

```
sudo apt-get update
sudo apt-get upgrade -y
```

Install Apache2

Once Raspbian has finished installing updates, we can now install the LAMP packages, we start with Apache2 and we also want to enable mod_rewrite:

```
sudo apt-get install apache2 -y
sudo a2enmod rewrite
sudo service apache2 restart
```

We're using the `-y` flag which will automatically install required and suggested Apache2 packages.

We need to change one setting on Apache2 to allow .htaccess overrides in the directory.
directory, so open `sudo nano /etc/apache2/apache2.conf` and change the `AllowOverride None` to `AllowOverride All` as shown below:

```xml
<Directory /var/www/>
   Options Indexes FollowSymLinks
   AllowOverride All
   Require all granted
</Directory>
```

...then restart Apache2:

```
sudo service apache2 restart
```

Now we need to find the **IP address** of your Pi, so type the following command:

```
ifconfig eth0
```

Look for `inet addr: 192.xxx.x.xxx` which will show your Pi’s IP address (it could also start with 10.x.x.x or 172.x.x.x). Go to any desktop computer that’s connected to your local network and enter the IP address into your browser’s address bar ie. [http://192.xxx.x.xxx](http://192.xxx.x.xxx) and hit Enter. You should be presented with the following **Apache2 Debian Default Page**:

---

**Install PHP**

Now let’s install **PHP** by entering the following command:
When finished installing PHP, we need to test it's working, so let's create a new PHP file:

```bash
sudo apt-get install php libapache2-mod-php -y
```

When finished installing PHP, we need to test it's working, so let's create a new PHP file:

```bash
cd /var/www/html
sudo nano index.php
```

...then enter the following code into your new PHP file:

```php
<?php
    echo "Hello World";
?>
```

Now on your keyboard, hold `Ctrl + O` to save, then `Ctrl + X` to exit nano. Now we need to delete the default Apache2 Debian page and restart Apache:

```bash
sudo rm index.html
sudo service apache2 restart
```

Go to your web browser again and enter the IP address of your Pi just like we did before, or if you're still on that page, hit `Ctrl + F5` to refresh. You should be greeted with a blank webpage that just says "Hello World".

## Install MySQL

Now we install the MySQL database engine by typing the following command:

```bash
sudo apt-get install mysql-server php-mysql -y
sudo service apache2 restart
```

## Install phpMyAdmin

In order to manage our MySQL databases, it's far easier to use phpMyAdmin so let's go ahead and install it by entering the following command, but note* you'll be asked the following questions during the installation process:

- Automatic Configuration? – Choose `[*]apache2` with your Spacebar, hit Tab then Enter
- Configure database for phpmyadmin with dbconfig-common? – Choose Yes
- phpMyAdmin application password – [enter new password] [confirm new password] this is for logging into the phpMyAdmin web interface

Once you are ready, type the following command:
Now let’s check if phpMyAdmin is working by entering the IP address of your Pi into your web browsers address bar appended with `/phpmyadmin/`, like this: 

http://192.xxx.x.xxx/phpmyadmin/ hit Enter and you should be presented with the standard **phpMyAdmin login page** as shown below:

...now login with username: `phpmyadmin` and password that you entered during phpMyAdmin setup.

**Setup FTP**

Now we need to install and setup VSFTPD and lock-down user pi to the `/var/www` folder, so let’s install the FTP service with the following command:

```bash
sudo apt-get install vsftpd -y
```

We need to open the `sudo nano /etc/vsftpd.conf` file and comment out the following two options:

```
local_enable=YES
ssl_enable=NO

#local_enable=YES
#ssl_enable=NO
```
...then add this code to the very bottom of the file:

```plaintext
# CUSTOM
ssl_enable=YES
local_enable=YES
chroot_local_user=YES
local_root=/var/www
user_sub_token=pi
write_enable=YES
local_umask=002
allow_writeable_chroot=YES
ftpd_banner=Welcome to my Raspberry Pi FTP service.
```

We also need to add user `pi` to the `www-data` group, give ownership of the `/var/www` folder to `www-data` user and group, change user `pi`'s home folder to same, and loosen some permissions on the `/var/www` folder:

```plaintext
sudo usermod -a -G www-data pi
sudo usermod -m -d /var/www pi
sudo chown -R www-data:www-data /var/www
sudo chmod -R 775 /var/www
```

...then restart the VSFTPD service:

```plaintext
sudo service vsftpd restart
```

**Login to FTP (VSFTPD) Service**

You can now open your FTP program and setup an FTP connection with the following settings:

- Host – **192.xxx.x.xxx** (IP address of your Pi with no prefix)
- Port – **21**
- Protocol – **FTP** (File Transfer Protocol)
- Encryption – **Use explicit FTP over TLS if available**
- Logon Type – **Normal** (username & password)
- Username – **pi**
- Password – [enter password]
About Richie

I started working with computers in 1996 with the advent of the Internet and started my own online shop selling musical instrument accessories in 2000.

In 2006 I studied Multimedia Development & Design which opened up the world of Photoshop, HTML, Flash and JavaScript.

Since then I have designed dozens of websites as well as maintaining my own group of websites including Ricmedia PC Help and Ricmedia Guitar among others.

I am currently the webmaster and content creator for all Ricmedia.com websites.
Kenneth H says
February 4, 2018 at 6:28 pm

awesome tutorial!

everything worked perfect, untill i installed wordpress.

after i Enter database details and then click Submit, i get this error

Error establishing a database connection
This either means that the username and password information in your wp-config.php file is incorrect or we can't contact the database server at localhost. This could mean your host's database server is down.

Are you sure you have the correct username and password?
Are you sure that you have typed the correct hostname?
Are you sure that the database server is running?
If you're unsure what these terms mean you should probably contact your host. If you still need help you can always visit the WordPress Support Forums.

i am 100% sure i enter the right username and password.

what could the problem be, and how can i fix it?

Reply

Richie says
February 4, 2018 at 6:50 pm

Hi Kenneth, thanks mate. This error means that either one or more of the following is incorrect: db username, db password, db host (which should be localhost). When you entered the code to create the database user: `create user 'wordpress'@'localhost' identified by '[enter password]';` did you...
enter password without the brackets? like this: `create user 'wordpress2'@'localhost' identified by 'abc123';`? If you’re unsure, create a new database and user as follows:

```
create user 'wordpress2'@'localhost' identified by 'abc123';
create database wordpress2;
grant all privileges on wordpress2.* to 'wordpress2'@'localhost';
flush privileges;
exit;
```

Then enter username: `wordpress2`, password: `abc123`, database name: `wordpress2`, host: `localhost`, db prefix: `wp_` and see if that works mate.

Let me know how you go.

cheers

Richie

---

Kenneth H says
February 4, 2018 at 10:44 pm

that did the trick 😊

thank you ! and sorry for my noob mistake

---

Richie says
February 4, 2018 at 10:46 pm

No worries at all mate, cheers, Richie

---

Richie says
January 30, 2018 at 8:43 pm
Please Note* This guide has been fully updated (2018-01-30) to reflect the changes to MySQL in Raspbian “Stretch” version 9.x, which has caused a lot of the code to break. There are a few different aspects to how we create databases in that we run MySQL with the sudo prefix instead of logging in as root. Also, all code is now written to conform to the Linux sudo system which greatly increases security.

Let me know if you have any issues with the new code and instructions, but I think you’ll find it all works far better now.

Cheers!
Richie

Angelo says
January 30, 2018 at 10:30 am

Thanks alot for the guide. Please note that default phpmyadmin login details are: phpmyadmin with the password that you provided in the setup prompt.

orthodoxcumbriawebmaster says
January 27, 2018 at 4:08 am

I think something has changed in the way phpmyadmin is set up, I have tried several tutorials without success. I can log in as phpmyadmin, but then can’t create a new database (no privileges), and root access is denied:

#1698 – Access denied for user ‘root’@’localhost’
mysqli_real_connect(): (HY000/1698): Access denied for user ‘root’@’localhost’

Googling for a solution offers many, but there is no trace of AllowRootLogin in any of the configuration files......
I've done this many times on other machines, but this has been stumped.

Reply

Richie says
January 27, 2018 at 10:59 am

Hi Richard, yes it’s actually MySQL that has changed and now does not allow root login via phpmyadmin interface. But you can fix this issue by following this guide: https://askubuntu.com/questions/763336/cannot-enter-phpmyadmin-as-root-mysql-5-7

Let me know how you go mate.

cheers!
Richie

Reply

Joshua says
January 7, 2018 at 5:40 am

Hi Richie,

Thanks for putting this tutorial together! I was going through the steps using Raspbian 9.3 and encountered a problem with http:///phpmyadmin/ similar to Michu after completing the phpmyadmin install instructions above. I corrected the 404 not found by including an extra step:

```
nano /etc/apache2/apache2.conf
```

At the bottom of the file append:

```
Include /etc/phpmyadmin/apache.conf
```

Save and quit, then restart the service:

```
service apache2 restart
```
These steps made the phpmyadmin web interface login available.

Reply

Richie says
January 7, 2018 at 11:46 am

Hi Joshua, thanks for the info. The issue is due to Debian changing the default MySQL engine from MySQL to MariaDB in version 9.x which is why I link directly to version 8.x Jessie in the guide. I will have to update the guide to reflect these change as soon as I can.

cheers
Richie

Reply

Michu says
December 30, 2017 at 10:42 pm

Hi Richie
Thanks for the great tutorial!
I can't access the phpmyadmin site... how can I delete it on the raspberry to install it again?

Reply

Richie says
December 30, 2017 at 10:53 pm

Hi Michu, are you sure you put in the right web address? Remember you need to find the Raspberry Pi IP address with `ifconfig` or `ip address` to find your IP address, then enter `http://[ip-address]/phpmyadmin/` so be sure you try that first.

If you still want to uninstall PHPMyAdmin, login as root and enter the following code on line at a time:
dpkg -P phpmyadmin
rm -f /etc/apache2/conf.d/phpmyadmin.conf
service apache2 restart

Cheers!
Richie

Angel says
December 28, 2017 at 8:01 pm

Thank you very much for this tutorial.

During the Install MySQL you mention “note” that you will be asked to enter and confirm a root password during the installation process”, this is not my case the process it doesn’t request MySQL password.

Late in the Setup WordPress Database request the above password, but I don’t know. “Login to MySQL as root using the same MySQL root password we entered when installing it earlier”

Could you please tell me how to get the root MySQL password in your installation process?

Reply

Richie says
December 28, 2017 at 8:15 pm

Hi Angel, apparently Debian/Raspbian now has MariaDB as the default database instead of MySQL engine. This has made the setup slightly different in that no root password is given. Basically, instead of using the old `u root -p` command you simply use `sudo mysql` or if you’re already logged in as root just type `mysql` to bring up the interface. AFAIK MariaDB has no root password making it impossible to actually login as root as a security measure.
you need to login with a user that has lower permissions (ie, user of a particular database) or follow this guide on how to give root as password.

Let me know how you go.
cheers!
Richie

---

**Tevis C says**
November 30, 2017 at 6:12 am

I think a great addition to this would be to include a web IDE like Codiad into the tutorial. It allows a new user to get into the web development without getting too deep into setting up an IDE in Linux. This also allows the user to code their web page from other devices. I built a WAMP + Codiad + phpMyAdmin and will try to build the same on my R-Pi.

---

**Richie says**
November 30, 2017 at 8:22 am

I would love to do many more RPi tutorials, time is the factor 😊 cheers, Richie

---

**Scott J says**
June 9, 2017 at 12:03 pm

My bad.... I missed it.... you have a link to FileZilla in the section just above where I jumped to.... thank you!
Scott J says
June 8, 2017 at 1:25 pm

I think this tutorial is great, and I’m using it this evening to learn about LAMP servers and about my RPi 3. I’m basically new to Linux and while not “new car smell” new, I’m wondering how exactly you envision connecting SFTP? Just since i’m playing around, I figure the SSH-FTP route would be the way to go, but not sure how I’m supposed to use connection data. Are you suggesting using WinSCP or similar which takes the IP address, port, protocol, login, and password? I’m just not clear how that login process is supposed to happen. I’ll dig around on your site some more, but how you’re intending us to apply that information is unclear (I have a guess on some options, but your intention might be easier than what else I might try). Still, I greatly appreciate the post!

Reply

Richie says
June 8, 2017 at 1:48 pm

Hi Scott, thanks for your question and feedback. In this tutorial I give two ways to connect to your RPi; (1) via SSH-FTP (SFTP) which is a bit dangerous as you’ll login as root and have access to entire filesystem or (2) install and setup VSFTPD then login using FTPS which is the better option. The client for both connection types is PuTTY, but any FTP client should do. FTPS is FTP over a secure SSL or TLS encrypted connection, whereas SFTP or SSH-FTP is data transfer over the SSH connection. The reason I recommend FTPS is you can lock-down the folder access to just the /var/www folder, whereas the SSH-FTP gives access to the entire filesystem, I realize I could have created a separate Linux user, but it’s difficult to lock down to the /var/www folder, and permissions are a nightmare. Cheers! Richie

Reply

Scott J says
June 10, 2017 at 11:33 am

Thank you Richie. The precaution of using VSFTPD makes sense both in your reply above, and in the main article for the security reasons you cited. For my purposes,
which is just trying it out at home, the SSH-FTP route is just fine, but I understand your point and its validity.

One more point, which I think I have right and I’ll make here for anyone passing this way that momentarily is confused on that last section: Although you’re saying to use root for login after changing the sshd_config file, root is not enabled by default on Debian (which Raspbian is based) distributions. You would need to enable the root account, which is not covered in the procedure above.

Reply

Richie says
June 10, 2017 at 12:30 pm

Hi Scott, yes VSFTPD is the better/safer option. I actually do go through changing the AllowRootLogin without-password to AllowRootLogin yes under the “Login to FTP as Root User” section, but you may not have read it as you went down the VSFTPD path 😊 Cheers, Richie

Reply

Leave a comment

Enter your comment here...