

How To Install Linux, Apache, MySQL, PHP (LAMP) stack on Arch Linux

Authored by: **ASPHostServer Administrator** [asphostserver@gmail.com]

Saved From: <http://faq.asphosthelpdesk.com/article.php?id=213>

About LAMP

The LAMP stack is a group of open source software used to get web servers up and running. The acronym stands for Linux, Apache, MySQL, and PHP. Arch Linux uses the powerful Pacman installer which can download all of the most recent, required packages for each program with one command.

Set Up

The steps in this tutorial require the user to have root privileges. You can see how to set that up here in steps 3 and 4.

Step One—Install Apache

Apache is a free open source software that runs over 50% of the world's web servers.

Before installing any of the LAMP programs, we should update the package manager:

```
sudo pacman -Syu
```

Once the update is complete, we can install Apache:

```
sudo pacman -S apache
```

After Apache is installed, we need to make a couple of changes in the configuration.

Open up the apache configuration file:

```
sudo nano /etc/httpd/conf/httpd.conf
```

Comment out the `unique_id_module` (you can use `ctrl w` to find it quickly):

```
#LoadModule unique_id_module modules/mod_unique_id.so
```

Restart Apache:

```
sudo systemctl restart httpd
```

During the apache restart, you may get a notice that says the following:

```
httpd: apr_sockaddr_info_get() failed for panel1
httpd: Could not reliably determine the server's fully qualified domain name,
using 127.0.0.1 for ServerName
```

[DONE]

Although this alert does not interfere with the apache launch, you can easily eliminate it if you like by adding your hostname to your configuration.

Open up the hosts file:

```
sudo nano /etc/hosts
```

Add your hostname to the end of the line beginning with 127.0.0.1:

```
127.0.0.1          localhost.localdomain  localhost panell
```

Subsequent restarts of Apache will no longer display that message.

With that, Apache is now installed on your server. Directing your browser to your server's IP address (http://12.34.56.789) will display an auto-index directory.

You can quickly try out a sample page by adding an index.html file to Arch's document root directory, located within "srv/http":

```
sudo nano /srv/http/index.html
```

```
<html>
<title>Welcome</title>
<body>
<h2>Hello, Welcome to Arch</h2>
</body>
</html>
```

You can visit the placeholder page by going to your server's IP address in your browser.

How to Find your Server's IP address

You can run the following command to reveal your server's IP address.

```
curl -s icanhazip.com
```

Step Two—Install MySQL

MySQL is a powerful database management system used for organizing and retrieving data.

Since March 2013, MariaDB has become the default implementation of MySQL in Arch repositories. Installing the package "mysql" will automatically drop in MariaDB as a replacement. See the original announcement for more details.

To install MySQL, open terminal and type in these commands:

```
sudo pacman -S mysql
```

Hit enter if it asks you anything, unless you know you don't want the defaults.

Once you have installed MySQL, start it running

```
sudo systemctl start mysqld
```

Finish up by running the MySQL set up script:

```
sudo mysql_secure_installation
```

The process will ask you for your current MySQL root password (not to be confused with your server password) and, since it has not yet been set, simply press enter.

When prompted, "Set root password?" type Y, and afterwards enter in your new MySQL root password.

After that it's easiest just to say Yes to all the options. At the end, MySQL will reload and implement the new changes.

By default, a MySQL installation has an anonymous user, allowing anyone to log into MySQL without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

```
Remove anonymous users? [Y/n] y
```

```
... Success!
```

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

```
Disallow root login remotely? [Y/n] y
```

```
... Success!
```

By default, MySQL comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

```
Remove test database and access to it? [Y/n] y
```

```
- Dropping test database...
```

```
... Success!
```

```
- Removing privileges on test database...
```

```
... Success!
```

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

```
Reload privilege tables now? [Y/n] y
```

```
... Success!
```

```
Cleaning up...
```

Once you're done installing MySQL, finish the LAMP stack by installing PHP.

Step Three—Install PHP

PHP is an open source web scripting language that is widely used to build dynamic webpages.

To install PHP, open terminal and type in this command.

```
sudo pacman -S php php-apache
```

PHP should also be added to the apache configuration file:

```
sudo nano /etc/httpd/conf/httpd.conf
```

Paste in this text block within the config.

```
# Use for PHP 5.x:
LoadModule php5_module          modules/libphp5.so
AddHandler php5-script php
Include conf/extra/php5_module.conf
```

Step Four—See your PHP Config and Add Programs to Startup

After LAMP is completely installed, we can take a look and see the PHP details by creating a quick php info page

To set this up, first create a new blank file:

```
sudo nano /srv/http/info.php
```

Add in the following line:

```
<?php
phpinfo();
?>
```

Then Save and Exit.

Restart apache so that all of the changes take effect:

```
sudo systemctl restart httpd
```

Visit your php info page (make sure you replace the example ip address with your correct one):
<http://12.34.56.789/info.php>

To close out the LAMP installation, open up the configuration file of Arch's innitscripts and add the apache server and MySQL daemons to the list of programs that run upon server startup:

```
sudo systemctl enable mysqld httpd
```

Congratulations, you have now installed the LAMP server on Arch Linux!