A LAMP (Linux, Apache, MySQL, PHP) stack is a common web stack used for hosting web content. This guide shows you how to install a LAMP stack on a CentOS 7 server.

1. Ensure that you have followed the Getting Started and Securing Your Server guides, and the Linode’s hostname is set.

To check your hostname run:

hostname

hostname -f

The first command should show your short hostname, and the second should show your fully qualified domain name (FQDN).

1. Update your system:
2. sudo yum update

Apache

Install and Configure

1. Install Apache 2.4:
2. sudo yum install httpd
3. Edit httpd.conf and add the code below to turn off KeepAlive and adjust the resource use settings. The settings shown below are a good starting point for a **Linode 2GB**:

**Note**

Before changing any configuration files, it is advised that you make a backup of the file. To make a backup:

cp /etc/httpd/conf/httpd.conf ~/httpd.conf.backup

**/etc/httpd/conf/httpd.conf**

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 910 | KeepAlive **Off**  **<IfModule** prefork.c**>**  StartServers 4  MinSpareServers 20  MaxSpareServers 40  MaxClients 200  MaxRequestsPerChild 4500  **</IfModule>** |

These settings can also be added to a separate file if so desired. The file must be located in the conf.module.d or conf directories, and must end in .conf.

Configure Name-based Virtual Hosts

There are different ways to set up virtual hosts; however, the method below is recommended.

1. Within the conf.d directory create vhost.conf to store your virtual host configurations. The example below is a template for website example.com; change the necessary values for your domain:

**/etc/httpd/conf.d/vhost.conf**

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 910 | NameVirtualHost \*:80  **<VirtualHost** \*:80**>**  ServerAdmin webmaster@example.com  ServerName example.com  ServerAlias www.example.com  DocumentRoot /var/www/html/example.com/public\_html/  ErrorLog /var/www/html/example.com/logs/error.log  CustomLog /var/www/html/example.com/logs/access.log combined  **</VirtualHost>** |

Additional domains can be added to the vhost.conf file as needed.

**Note**

ErrorLog and CustomLog entries are suggested for more fine-grained logging, but are not required. If they are defined (as shown above), the logs directories must be created before you restart Apache.

1. Create the directories referenced above:
2. sudo mkdir -p /var/www/html/example.com/{public\_html,logs}
3. Enable Apache to start at boot, and restart the service for the above changes to take place:
4. sudo systemctl enable httpd.service
5. sudo systemctl restart httpd.service

You can now visit your domain to test the Apache server; a default Apache page will be visible.

MySQL / MariaDB

Install and Configure

MySQL is replaced with MariaDB in CentOS 7. MariaDB is a popular drop-in replacement for MySQL.

**Note**

If you prefer to use the MySQL branded database in CentOS 7, you will need to add the required repositories by issuing the following command:

sudo yum install http://dev.mysql.com/get/mysql57-community-release-el7-7.noarch.rpm

1. Install the MariaDB-server package:
2. sudo yum install mariadb-server
3. Set MariaDB to start at boot and start the daemon for the first time:
4. sudo systemctl enable mariadb.service
5. sudo systemctl start mariadb.service
6. Run mysql\_secure\_installation to secure MariaDB. You will be given the option to change the MariaDB root password, remove anonymous user accounts, disable root logins outside of localhost, and remove test databases and reload privileges. It is recommended that you answer yes to these options:
7. mysql\_secure\_installation

Create a MySQL/MariaDB Database

1. Log in to MariaDB:
2. mysql -u root -p

Enter MariaDB’s root password. You will get the MariaDB prompt.

1. Create a new database and user with permissions to use it:
2. create database webdata;
3. grant all on webdata.\* to 'webuser' identified by 'password';

In the above example webdata is the name of the database, webuser the user, and password a strong password.

1. Exit MariaDB
2. quit

With Apache and MariaDB installed, you are now ready to move on to installing PHP to provide scripting support for your web pages.

PHP

Install and Configure

1. Install PHP:
2. sudo yum install php php-pear

If you wish to install MySQL support for PHP also install the php-mysql package:

sudo yum install php-mysql

1. Edit /etc/php.ini for better error messages and logs, and upgraded performance. These modifications provide a good starting point for a **Linode 2GB**:

**/etc/php.ini**

|  |  |
| --- | --- |
| 123 | error\_reporting = E\_COMPILE\_ERROR|E\_RECOVERABLE\_ERROR|E\_ERROR|E\_CORE\_ERROR  error\_log = /var/log/php/error.log  max\_input\_time = 30 |

**Note**

Ensure that all lines noted above are uncommented. A commented line begins with a semicolon (**;**).

1. Create the log directory for PHP and give the Apache user ownership:
2. sudo mkdir /var/log/php
3. sudo chown apache /var/log/php
4. Reload Apache:

sudo systemctl reload httpd