

# Ubuntu LAMP Server Install

To begin this lab you must already have 2 VMs (1 Ubuntu server, 1 Ubuntu client) with the OS installed & networking configured & the 2 machines able to ping each other.

Once, the 2 machines can ping each other you will need to be aware of when you need to have the VM n/w adapter set to NAT (to connect to the web to download packages) or to Host-Only (for comm. between the client & the server.)

We will now look at 2 different ways to install the Ubuntu LAMP Server.

The LAMP service is basically a bundle that consists of an Apache Server, PHP & MySQL.

Please note that before you start this lab you should take a snapshot of your server VM as we will be installing the LAMP server in 2 different ways. So you will use 1 copy of the server VM for the first method the other copy for the second method.

## **Method 1 Installing Ubuntu LAMP Server using tasksel**

**This way of Installing LAMP uses the tasksel command (this uses a scripted install)**

### Installation instructions

Make sure your package repositories and installed programs are up to date by issuing the following commands.

To refresh the list of available packages:

```
#apt-get update
```

To upgrade installed packages to latest versions:

```
#apt-get upgrade --show-upgraded
```

To install LAMP:

```
#tasksel
```

Now Select LAMP Server.

During the install you will be asked to provide a root password for mysql.

Now check to see if php is working **#nano /var/www/info.php**

Add the following lines:

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

Exit and save as **/var/www/info.php**

Restart apache2

```
#/etc/init.d/apache2 restart
```

Now open a browser and type

**http://LAMP-Server-IP-Address/info.php**

To fully manage your lamp Server database, install phpmyadmin.

```
#apt-get install phpmyadmin
```

Now open a browser and type:

**http://LAMP-Server-IP-Addresss/phpmyadmin**

Install WordPress

```
#apt-get install wordpress
#find / -name wordpress
#ls /usr/share
#mv /usr/share/wordpress /var/www
```

Setup MySQL

```
mysql -u root -p
```

Enter your MySQL root user password from above, and you'll find yourself at the MySQL> local client prompt. First, create a database for WordPress to use:

```
CREATE DATABASE wordpress;
```

Once the database is created, you'll need to create a database user for WordPress to use.

```
CREATE USER admin;
```

Then create a password for your new user (note that your password will actually go within the quotation marks, and is case-sensitive):

```
SET PASSWORD FOR admin = PASSWORD("password");
```

Now that you've got your WordPress database and your WordPress database user, you'll need to grant the user all permissions on the database (again, the password goes within the quotation marks, and is case sensitive):

```
GRANT ALL PRIVILEGES ON wordpress.* TO admin@localhost IDENTIFIED BY
'password' ;
```

(Make sure to append "@localhost" to wordpressuser; otherwise you'll get a database error when you try to start WordPress for the first time.)

Configure WordPress to talk to the MySQL database. To do so, you must create a wp-config.php file in the WordPress directory. Fortunately, WordPress includes a handy wp-config-sample.php you can use as a template.

```
nano /var/www/wordpress/wp-config-sample.php
```

Once there, change the following values (This assumes you used the default database names and passwords listed above):

Change **database\_name\_here** to **wordpress**.

Change **username\_here** to **admin**.

Change **password\_here** here to the password you selected in the GRANT ALL ON wordpress.\* TO wordpressuser IDENTIFIED BY '**password**'; command.

Save your changed file to **/var/www/wordpress/wp-config.php**.

Everything should now be ready. Open up a web browser, and navigate to **http://ipaddress/wordpress**. If you configured everything correctly, you should then be greeted by the WordPress configuration page.

## **Method 2 Installing Ubuntu LAMP Server using a traditional install (manually through the Command Line)**

Here is an [article](#) that describes how to install Apache2, PHP5, MySQL on Ubuntu 12.04. This is an alternative and more traditional install as it does not use the scripted install that we used in the lab with **taskel**. It doesn't matter which method you use, the objective is the same, a fully working LAMP server. Use whichever method you wish, however it's worth looking at this article as the install is more detailed.