

How to Setup a LAMP Server on a Ubuntu Linux system

The purpose of this brief guide is to take you through the process of setting up a LAMP (Linux, Apache, MySQL, PHP) server on a local Ubuntu Linux machine or virtual machine.

This will allow you to develop using PHP and MySQL (with phpMyAdmin). This is a common stack that is necessary for Wordpress development.

Install the necessary packages

You will need to install the following packages for the LAMP server. You can install them all at once by separating each package by a space, or one at a time like shown.

I prefer to download one at a time because it is easier to see if there were any errors.

Enter the terminal and type the following:

- `sudo apt-get install apache2`
- `sudo apt-get install php`
- `sudo apt-get install php-mysql`
- `sudo apt-get install mysql-server`

You should then be prompted to set a password for the MySQL root user. After setting the password continue to install:

- `sudo apt-get install libapache2-mod-php`
- `sudo apt-get install php-mcrypt`
- `sudo apt-get install phpmyadmin`

You should then be prompted which server to use. Select Apache by pressing enter. Select no for advanced server setup.

Change permissions to the /var/www/html

In order for PHP scripts and files to be run by the LAMP server they need to be saved in the /var/www/html directory. You can think of this location as your local server.

In order to make changes to this directory we need to change the permissions on it. In the terminal enter the command:

```
sudo chown {your ubuntu username} /var/www/html
```

Create a symbolic link to phpMyAdmin

By default, phpMyAdmin is installed in the /usr/share/ directory. We need to move it to our local server directory.

We navigate to the server directory that we want the link in by: `cd /var/www/html`

Then create the link by entering the command `ln -s /usr/share/phpmyadmin phpmyadmin`.

Restart Apache and test

Run the following command to restart Apache, setting the changes that were made:

```
sudo systemctl restart apache2
```

You should then be able to create an info.php file in the /var/www/html directory with this command: `touch /var/www/html/info.php`

In the file type the following php code:

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

Then, open a browser and type in localhost/info.php You should see a page from the php file you just wrote that gives you information about php.

Finally, to access phpMyAdmin go to localhost/phpmyadmin in your browser. The default root username is 'root' and the password is the password you chose earlier for the MySQL database.

Example Script

```
#!/bin/bash
# This will install Apache / MySQL / PHP on Linux (LAMP)
# It includes ImageMagick and enables .htaccess redirect files
# Many apps use those.

# Update repositories
sudo apt-get update -y

# Upgrade packages
sudo apt-get upgrade -y

# Install packages
sudo apt-get install -y apache2
sudo apt-get install -y php
sudo apt-get install -y php-mysql
sudo apt-get install -y php-cli
sudo apt-get install -y php-gd
sudo apt-get install -y php-curl
sudo apt-get install -y php-zip
sudo apt-get install -y mysql-server
sudo apt-get install -y libapache2-mod-php
sudo apt-get install -y phpmyadmin
sudo apt-get install -y imagemagick
sudo apt-get install -y php-imagick

# Prompt for MySQL root password
echo "Please enter the new password for the MySQL root user:"
read -s root_password

# Change authentication for root user
sudo mysql <<EOF
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY '${root_password}';
FLUSH PRIVILEGES;
EOF

# Enable mod_rewrite
sudo a2enmod rewrite

# Change AllowOverride directive
sudo sed -i 's/AllowOverride None/AllowOverride All/g' /etc/apache2/apache2.conf

# Change permissions to the /var/www/html
sudo chown $USER /var/www/html

# Remove any index.html or index.htm files in /var/www/html/
rm -f /var/www/html/index.htm
rm -f /var/www/html/index.html

# Create a symbolic link to phpMyAdmin
cd /var/www/html
ln -s /usr/share/phpmyadmin phpmyadmin

# Restart Apache
sudo systemctl restart apache2

# Create index.php file
echo "<?php phpinfo(); ?>" > /var/www/html/index.php

# Output completion message
echo "Setup completed. Visit localhost/index.php to check PHP info. Access phpMyAdmin at localhost/phpmyadmin with the root password you entered."
```

Installing SSL

```
#!/bin/bash
```

```
# Install Certbot and the Certbot Apache plugin
sudo apt-get install -y certbot python3-certbot-apache

# Prompt for the domain name
echo "Please enter the domain name for the SSL certificate:"
read domain_name

# Run Certbot for the domain
sudo certbot --apache -d $domain_name

# Test automatic renewal
sudo certbot renew --dry-run

# Check if ufw is installed
if command -v ufw &> /dev/null
then
    # Allow HTTPS through the firewall
    sudo ufw allow 'Apache Full'
else
    echo "ufw not found. If you have another firewall, please manually open port 443 (HTTPS)."
fi

# Output completion message
echo "SSL setup completed. Visit https://$domain_name to check the SSL status."
```

🕒Revision #2

★Created 13 July 2023 22:21:25 by Josh Levitsky

✎Updated 14 July 2023 00:30:31 by Josh Levitsky