INSTALL LAMP STACK ON AWS - UBUNTU 18

Step 1 — Installing Apache and Updating the Firewall

sudo apt update

sudo apt upgrade

sudo apt install apache2

sudo ufw app list

sudo ufw app info "Apache Full"

sudo ufw allow in "Apache Full"

APACHE INSTALLED SUCCESFULLY TILL HERE, YOU CAN CHECK BY ENTERING YOUR PUBLIC IP OR PUBLICK DNS ADDRESS - http://your\_server\_ip

==================================================

Step 2 — Installing MySQL

sudo apt install mysql-server

sudo mysql\_secure\_installations

sudo mysql

SELECT user,authentication\_string,plugin,host FROM mysql.user;

ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY '12345678'; // please note here replace the "password" with yours.

FLUSH PRIVILEGES;

SELECT user,authentication\_string,plugin,host FROM mysql.user;

exit

At this point, your database system is now set up and you can move on to installing PHP, the final component of the LAMP stack.

=========================================================

Step 3 — Installing PHP

sudo apt install php libapache2-mod-php php-mysql

In most cases, you will want to modify the way that Apache serves files when a directory is requested. Currently, if a user requests a directory from the server, Apache will first look for a file called index.html. We want to tell the web server to prefer PHP files over others, so make Apache look for an index.php file first.

sudo nano /etc/apache2/mods-enabled/dir.conf

Move the PHP index file (highlighted above) to the first position after the DirectoryIndex specification, like this:

<IfModule mod\_dir.c>

DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm

</IfModule>

-----------------------------------

sudo systemctl restart apache2

sudo systemctl status apache2

Press Q to exit this status output.

Check PHP Version by entering

php -v

Install the commonly required php modules by using the below commands - do remeber replace the php version number with your by checking the php -v command.

sudo apt install php8.1-common php8.1-mysql php8.1-xml php8.1-xmlrpc php8.1-curl php8.1-gd php8.1-imagick php8.1-cli php8.1-dev php8.1-imap php8.1-mbstring php8.1-opcache php8.1-soap php8.1-zip php8.1-intl -y

sudo systemctl restart apache2

==================================================

Step 4 — Testing PHP Processing on your Web Server

sudo nano /var/www/html/info.php

This will open a blank file. Add the following text, which is valid PHP code, inside the file:

<?php

phpinfo();

?>

The address you will want to visit is:

http://your\_ip/info.php

You will get the php info page

===========================

PHPMYADMIN INSTALL STEPS BELOW

Step 1 — Installing phpMyAdmin

sudo apt update

sudo apt install phpmyadmin php-mbstring php-gettext

Warning: When the prompt appears, “apache2” is highlighted, but not selected. If you do not hit SPACE to select Apache, the installer will not move the necessary files during installation. Hit SPACE, TAB, and then ENTER to select Apache.

sudo phpenmod mbstring

sudo systemctl restart apache2

===================================================

http://your\_domain\_or\_IP/phpmyadmin

====================================PERMISSIONS ADJUSTMENT================================

Step 2: Locate the PHP configuration file

Determining the right PHP configuration file can be very confusing especially because the ‘php.ini’ file can be located on a different folder depending on the PHP version.

The correct php.ini file should be in the Apache directory (e.g. ‘/etc/php/7.1/apache2/php.ini’). This will depend on the version of PHP. For instance, in Php7.2, the configuration file is located on ‘/etc/php/7.2/apache2/php.ini’

==============================================================

Step 3: Edit the Php Configuration file

sudo nano /etc/php/7.1/apache2/php.ini

Standard ‘php.ini’ settings file - Change the INI settings according to the below values:

memory\_limit = 128M

upload\_max\_filesize = 50M

post\_max\_size = 50M

max\_execution\_time = 120

sudo service apache2 restart

Step 4: Verify the php.ini settings

Refreshing the info.php page should now show your updated settings. Remember to remove the info.php when you are done changing your PHP configuration.

=========================================================================

Important Notes - Common Issues during/after PHP Install

MOST IMPORTANT

PERMISSION - YOU SHOULD OWN THE FILE BEFORE YOU CAN EDIT - YOU SHOULD KNOW THE USERNAME OF THE OPERATING SYSTEM.

=====================

Issues: Website pages not visible, not able to edit the files/folder , permsission denied issue, .htaccess not able to rewrite links.

Execute the Comands below to set proper file permissions on Directories and files.

sudo chown -R ubuntu:root /var/www/html

sudo find html -type d -exec chmod 775 {} \;

sudo find html -type f -exec chmod 664 {} \;

======================================================

Enabling mod\_rewrite on apache2

By default, Apache does not allow the use of ‘.htaccess’ file so you will need to edit the configuration of each website’s virtual host file by adding the following code:

OWN THE APACHE2 FOLDER FIRST IF YOU WANT TO EDIT VIA FILEZILLA OR FTP - sudo chown -R ubuntu:root /etc/apache2/ - Revert back to root:root chown when done.

OR VIA SSH TERMINAL

sudo nano /etc/apache2/apache2.conf

Change the setting as below : AllowOverride All

<Directory /var/www/html>

Options Indexes FollowSymLinks MultiViews

AllowOverride All

Require all granted

</Directory>

------------

sudo a2enmod rewrite

sudo service apache2 restart

=================

TO VERIFY ANY CONFIG ERRORS OR MISTAKE OR SYSNTAX ERRORS IN APACHE CONFIG FILE, RUN THE BELOW COMMAND - Its should show Syntax OK , if it not then there is some error in config due to which apache will not start.

sudo service apache2 status (Press Q to exit)

# apachectl configtest

Syntax OK

# apachectl -t

Syntax OK