Setting Up LAMP on a Debian Machine

A Step-by-Step Guide on Seting Up LAMP on a GCP Compute Engine with a Debian Instance

Fernando B. Enad (*Author*)
MIS Student
UP Open University
Los Baños, Laguna
fbenad@up.edu.ph

***Abstract*—This electronic document serves as a guide on how to get a LAMP stack running on a Google Compute Engine virtual machine instance running Debian.** (*Abstract*)

***Keywords*—linux debian; web server; LAMP; Google Cloud Platform** *(key words)*

I. Introduction

Much like Windows and Mac OS X, Debian is a computer operating system- a particular flavor of Linux that includes additional software and utilities. Debian has a long history of release cycles and being a community distro, new versions of Debian are released only when the community regards them as ready for public. The benefit of this is that each stable release is, indeed, stable.

Debian bills itself as "The Universal Operating System", so there is only one image for you to download and install. The default will work as the server edition but if you install a desktop manager, it becomes the desktop edition.

Moreover, Debian employs the Advanced Packaging Tool (APT) so with just a single command, one can add and remove software from your system, and you can also update your packages from one central repository.

Although no precise statistics are available (since Debian does not require users to register), evidence is quite strong that Debian is used by a wide range of organizations, large and small, as well as many thousands of individuals. In fact, Debian is known as the king of Linux distributions and the most popular Linux Server Distro at present. As it provides various package manager and various API tools, not to mention of it being a secure and more stable server, Debian has become a popular choice for servers.

To this effect, this document is focused at providing the step-by-step process about LAMP installation on Debian running machines with multi-tier architecture on GCP (Google Cloud Plartform) where the Compute Engine service of GCP for Linux server, Apache Web server and php will be used. On the other hand, for the MySQL database, the SQL service of GCP as a separate micro service, will be used and both services will be connected to design a multi-tier architecture for LAMP installation. Furthermore, this document will only focus on creating a virtual machine instance, connecting to your instance via Secure Shell (SSH), deploying the LAMP (Linux, Apache, MySQL, PHP) to on your instance, and finally, configuring cron jobs.

II. Prerequisites

Google Cloud Platform (GCP) is a suite of cloud computing services offered by Google and runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search, Youtube, etc. GCP was publicly introduced by Google in 2011 and since then, it provides a number of cloud services which includes computing, storage, data analytics, machine learning, etc. In this document, the steps on how you can create a Compute Engine VM (Virtual Machine) instance on Google Cloud Platform will be discussed.

Before you start creating a VM in GCP though, you need to first register on the GCP console at https://cloud.google.com with your existing GMail account. Registration requires a credit card or bank account details and upon registration, you will also get a $300 free credit usage for 12 months. If you have already registered for GCP, you are good to go.

III. Creating a Virtual Machine Instance

In creating a VM instance, first login to your GCP console and on the navigation menu on the top left, under the compute section, select Compute Engine > VM Instances to bring out the Compute Engines interface. In this interface, click on the Create Instance button to proceed with creating a VM instance.

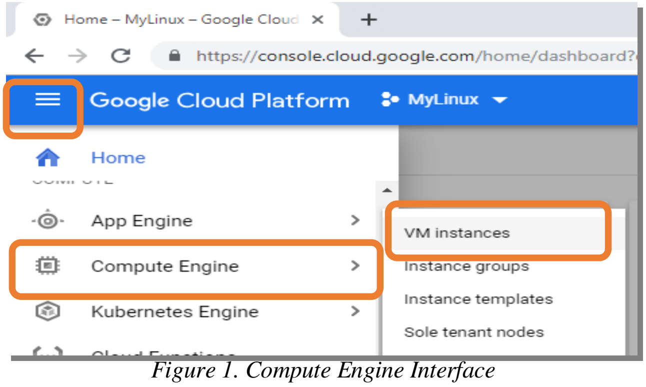
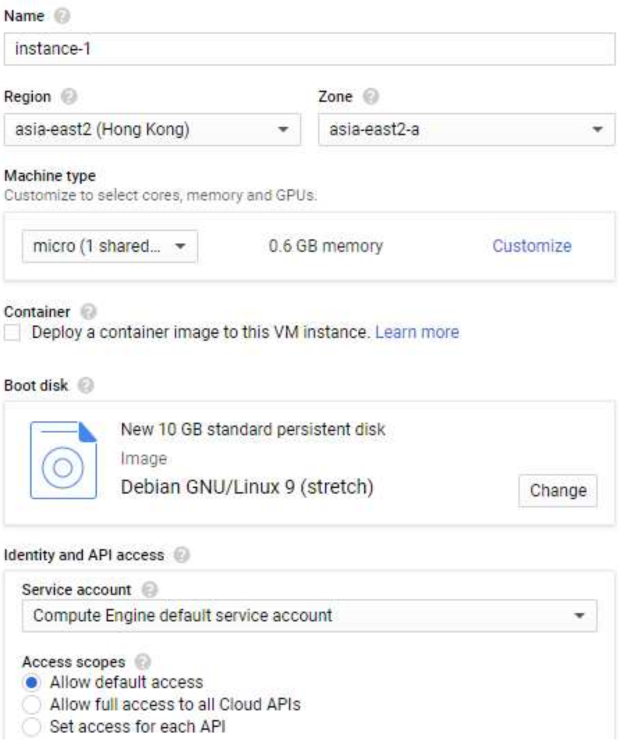


Figure 1. Compute Engine Interface

Next, you will see a “Create an instance” screen where you will need to provide the required configuration details. For the sake of this tutorial, you may replicate the following procedures:

1. Set the Name to lamp-tutorial.
2. Set the region and zone to the ones nearest to your geographical location. On this tutorial, you may use
3. Set the Machine type to f1-micro.
4. Leave the container section unchecked.
5. In the Boot disk section, click Change to begin configuring your boot disk.
6. In the OS images tab, choose Debian GNU/Linux 9 (stretch) and you may leave the Boot disk type and Size with their default values. Click to select to continue.
7. In the identify and API access section, leave the configuration as is.
8. In the Firewall section, select Allow HTTP traffic and Allow HTTPS traffic.
9. Next, you will find the “Management, security, disks, networking, sole tenancy” section which you can access from the drop-down arrow. These settings allow you to adjust the other settings as per your requirements. On this tutorial, you may use the default settings.
10. Click the Create button to create the instance.
11. Give the instance a few seconds to start up.



Name
instance-1

Region
asia-east2 (Hong Kong)

Zone
asia-east2-a

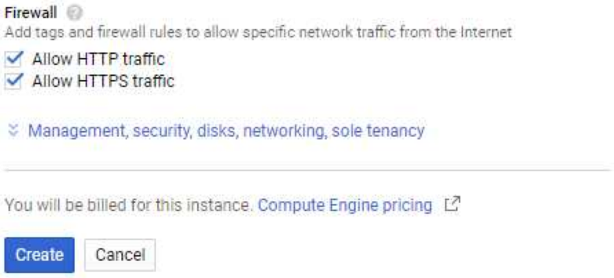
Machine type
Customize to select cores, memory and GPUs.
micro (1 shared... | 0.6 GB memory | Customize

Container
□ Deploy a container image to this VM instance. Learn more

Boot disk
New 10 GB standard persistent disk
Image
Debian GNU/Linux 9 (stretch)
Change

Identity and API access
Service account
Compute Engine default service account

Access scopes
● Allow default access
○ Allow full access to all Cloud APIs
○ Set access for each API



Firewall
Add tags and firewall rules to allow specific network traffic from the Internet
☑ Allow HTTP traffic
☑ Allow HTTPS traffic

Management, security, disks, networking, sole tenancy

You will be billed for this instance. Compute Engine pricing

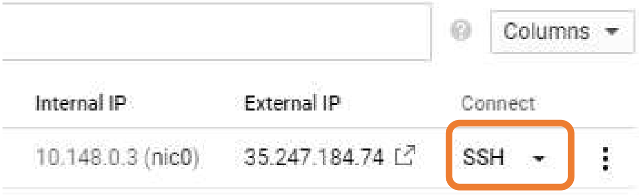
Create | Cancel

Figure 2. Create VM Instance Interface

In the list of VM instances, the instance that you have previously created should already appear here. In the event of creation issues, a notification will be reflected correspondingly which you need to read and comply. Make a note of the IP address of your VM instance reflected in the External IP column.

IV. Connecting to Your Instance Via Secure Shell (SSH)

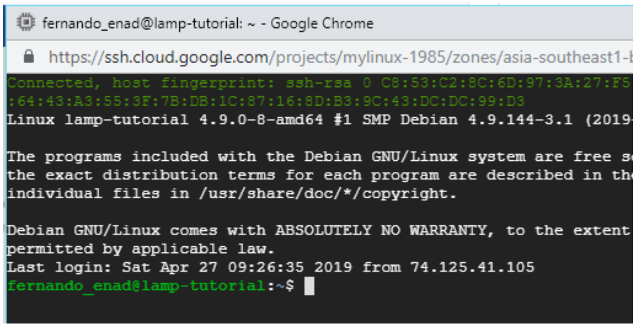
You can login to SSH either via GCP provided web console or third-party SSH tools like terminal or putty. On this tutorial, the GCP web console for SSH login will be used. To SSH login to the VM instance via “SSH web interface”, click on the SSH button of the instance from the console.



Internal IP	External IP	Connect
10.148.0.3 (nic0)	35.247.184.74	SSH

Figure 3. VM Instances Interface

On clicking on the SSH button of the VM instance, an SSH web interface will be prompted where you will be logged in to that particular instance. Here you can perform any operations on the server via SSH.



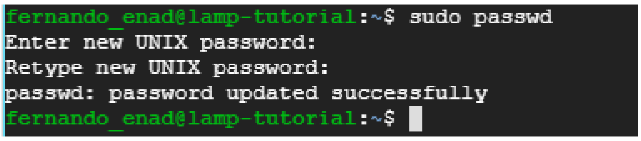
fernando_enad@lamp-tutorial: ~ - Google Chrome

https://ssh.cloud.google.com/projects/mylinux-1985/zones/asia-southeast1-

```
Connected, host fingerprint: ssh-rsa 0 C8:53:C2:8C:6D:97:3A:27:F5
:64:43:A3:55:3F:7B:DB:1C:87:16:8D:B3:9C:43:DC:DC:99:D3
Linux lamp-tutorial 4.9.0-8-amd64 #1 SMP Debian 4.9.144-3.1 (2019

The programs included with the Debian GNU/Linux system are free s
the exact distribution terms for each program are described in th
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Apr 27 09:26:35 2019 from 74.125.41.105
fernando_enad@lamp-tutorial:~$
```

Figure 4. VM Instance Secure Shell (SSH)

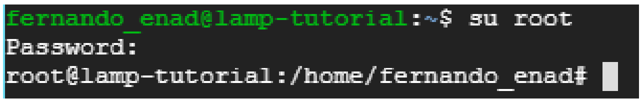
The VM's in cloud engine don't come with a root password setup by default so you'll first need to change the password using the **sudo passwd** command as shown in Figure 5.



```
fernando_enad@lamp-tutorial:~$ sudo passwd
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
fernando_enad@lamp-tutorial:~$
```

Figure 5. Root Password Creation Interface

You will then be asked to input your desired password and will be asked to retype it for confirmation. Once successful, you can now proceed to logging in as root user by using the **su root** command as shown below.



```
fernando_enad@lamp-tutorial:~$ su root
Password:
root@lamp-tutorial:/home/fernando_enad#
```

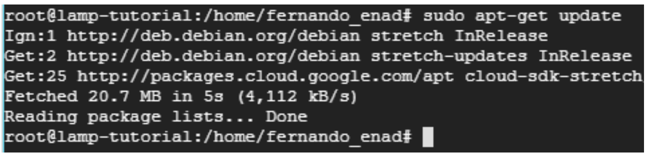
Figure 5.1. Root Login Interface

Be very careful though when logged on as root user as you can do everything -- the system won't ask.

V. Deploying The LAMP Stack On Compute Engine

Now that your virtual machine instance is running, you may now configure the LAMP stack. By creating an instance though, you already have the "Linux" part of LAMP, that said you may proceed to installing Apache and PHP.

Before going any further, your Debian environment has to be updated first and you can do so using the **sudo apt-get update** command as shown below.

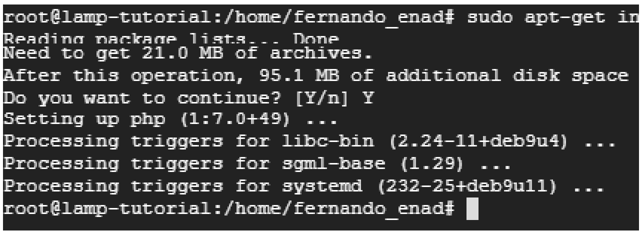


```
root@lamp-tutorial:/home/fernando_enad# sudo apt-get update
Ign:1 http://deb.debian.org/debian stretch InRelease
Get:2 http://deb.debian.org/debian stretch-updates InRelease
Get:25 http://packages.cloud.google.com/apt cloud-sdk-stretch
Fetched 20.7 MB in 5s (4,112 kB/s)
Reading package lists... Done
root@lamp-tutorial:/home/fernando_enad#
```

Figure 6. Environment Update Interface

A. *Install Apache and PHP on Your Instance*

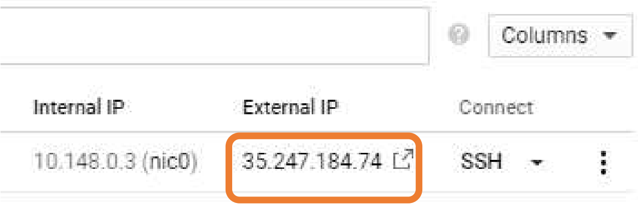
To install apache web server, use the **sudo apt-get install apache2 php libapache2-mod-php** commands as shown below.



```
root@lamp-tutorial:/home/fernando_enad# sudo apt-get in
Reading package lists... Done
Need to get 21.0 MB of archives.
After this operation, 95.1 MB of additional disk space
Do you want to continue? [Y/n] Y
Setting up php (1:7.0+49) ...
Processing triggers for libc-bin (2.24-11+deb9u4) ...
Processing triggers for sgml-base (1.29) ...
Processing triggers for systemd (232-25+deb9u11) ...
root@lamp-tutorial:/home/fernando_enad#
```

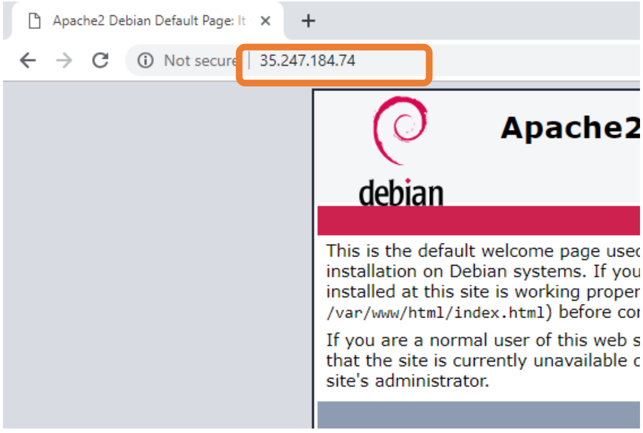
Figure 7. Apache Web Server and PHP Installation Interface

You will be asked to confirm whether to proceed or not after the additional disk space requirement notification and you would need to type "Y" to complete the installation process of the Apache Web Server and the PHP programming language. Once completed, you will already have a fully functioning Web Server and using the external IP address of your VM instance, you may now access your site over a browser as shown in Figure 8.



Internal IP External IP Connect
10.148.0.3 (nic0) 35.247.184.74 SSH

Figure 8. VM Instance External IP



This is the default welcome page use
installation on Debian systems. If you
installed at this site is working proper
/var/www/html/index.html) before co

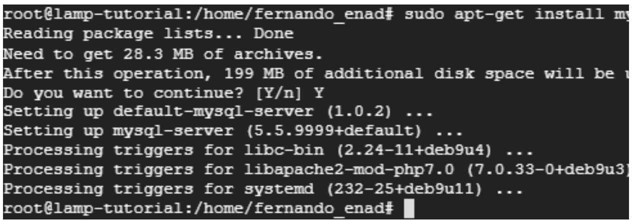
If you are a normal user of this web s
that the site is currently unavailable c
site's administrator.

Figure 9. Web Server Viewed via a Browser

Congratulations, you have successfully configured your Web Server!

B. *Install MySQL and related PHP Components*

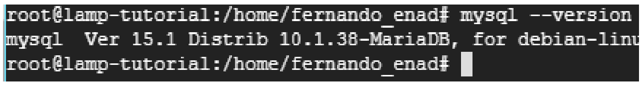
Now that the Apache Web Server and PHP components have been installed on your server, you may also need to install the MySQL service by using the **sudo apt-get install mysql-server php-mysql php-pear** commands as shown below.



```
root@lamp-tutorial:/home/fernando_enad# sudo apt-get install my
Reading package lists... Done
Need to get 28.3 MB of archives.
After this operation, 199 MB of additional disk space will be u
Do you want to continue? [Y/n] Y
Setting up default-mysql-server (1.0.2) ...
Setting up mysql-server (5.5.9999+default) ...
Processing triggers for libc-bin (2.24-11+deb9u4) ...
Processing triggers for libapache2-mod-php7.0 (7.0.33-0+deb9u3)
Processing triggers for systemd (232-25+deb9u11) ...
root@lamp-tutorial:/home/fernando_enad#
```

Figure 10. MySQL Installation Interface

You will be asked to confirm whether to proceed or not after the additional disk space requirement notification and you would need to type "Y" to complete the installation process of the MySQL service. You may check the MySQL installation and its version, by using **mysql --version** command as shown below.

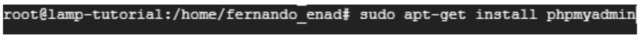


```
root@lamp-tutorial:/home/fernando_enad# mysql --version
mysql  Ver 15.1 Distrib 10.1.38-MariaDB, for debian-linu
root@lamp-tutorial:/home/fernando_enad#
```

Figure 11. MySQL Version Interface

C. *Install phpMyAdmin*

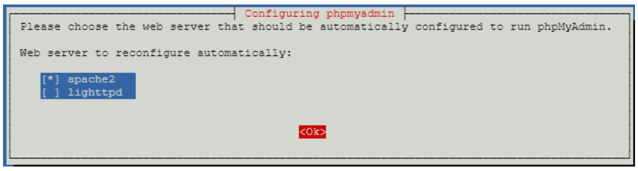
To easily administer your database through a UI, you can install the phpMyAdmin component by using the **sudo apt-get install phpmyadmin** command as shown below.



```
root@lamp-tutorial:/home/fernando_enad# sudo apt-get install phpmyadmin
```

Figure 12. phpMyAdmin Installation Interface

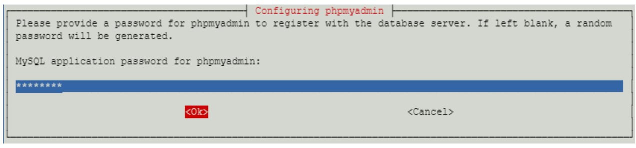
This will prompt you the configuration window from which you have to select "apache2" from the option list and clicking on "OK" as shown below.



```
Configuring phpmyadmin
Please choose the web server that should be automatically configured to run phpMyAdmin.

Web server to reconfigure automatically:

    [*] apache2
    [ ] lighttpd

                                  <Ok>
```

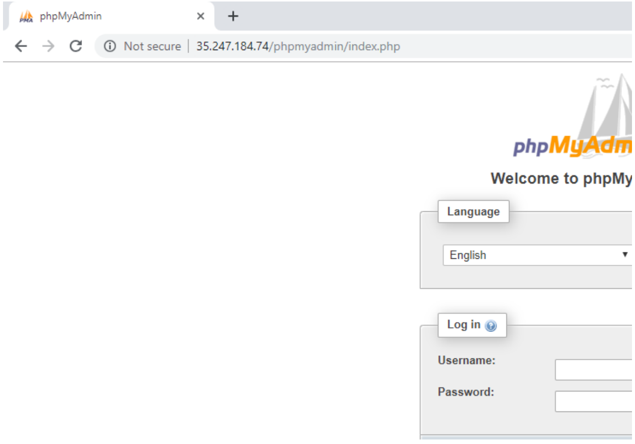
Figure 13. phpMyAdmin Package Installation Initial

The automated installation process will then follow until the MySQL root user password is prompted for you to supply and confirm as shown below.



```
Configuring phpmyadmin
Please provide a password for phpmyadmin to register with the database server. If left blank, a random
password will be generated.

MySQL application password for phpmyadmin:

********

           <Ok>                                  <Cancel>
```

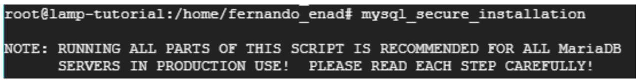
Figure 13.1. phpMyAdmin Root Password Creation



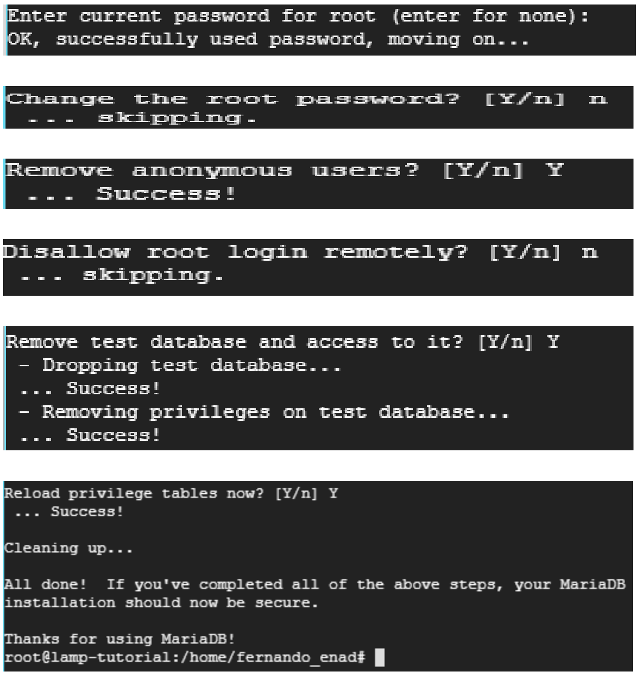
Not secure | 35.247.184.74/phpmyadmin/index.php

Figure 14. phpMyAdmin Portal

Now that you have MySQL installed, you should run the **mysql_secure_installation** command to improve the security of your installation. This performs steps such as setting the root user password if it is not yet set, removing the anonymous user, restricting root user access to the local machine, and removing the test database as shown below.



```
root@lamp-tutorial:/home/fernando_enad# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!
```



```
Enter current password for root (enter for none):
OK, successfully used password, moving on...
```

```
Change the root password? [Y/n] n
 ... skipping.
```

```
Remove anonymous users? [Y/n] Y
 ... Success!
```

```
Disallow root login remotely? [Y/n] n
 ... skipping.
```

```
Remove test database and access to it? [Y/n] Y
 - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!
```

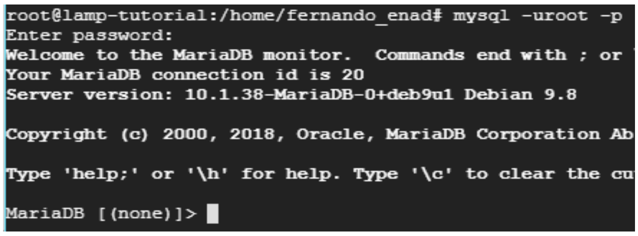
```
Reload privilege tables now? [Y/n] Y
 ... Success!

Cleaning up...

All done!  If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
root@lamp-tutorial:/home/fernando_enad# █
```

Figure 15. phpMyAdmin Configuration Interfaces

D. *Configuring Access to phpMyAdmin*

Now that the phpMyAdmin component has been successfully installed, you can now get inside the phpMyAdmin portal, however, the root user is not configured to access it yet, thus you need to create another user whom you need to grant access to.

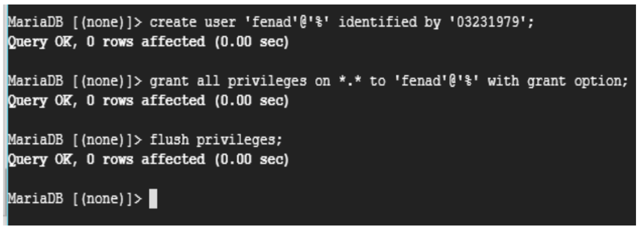
To do so, you would need to login to the MySQL service by using the **mysql -uroot -p** command where you will be asked to supply the root password as shown below.



```
root@lamp-tutorial:/home/fernando_enad# mysql -uroot -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or
Your MariaDB connection id is 20
Server version: 10.1.38-MariaDB-0+deb9u1 Debian 9.8

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab

Type 'help;' or '\h' for help. Type '\c' to clear the cu

MariaDB [(none)]> █
```

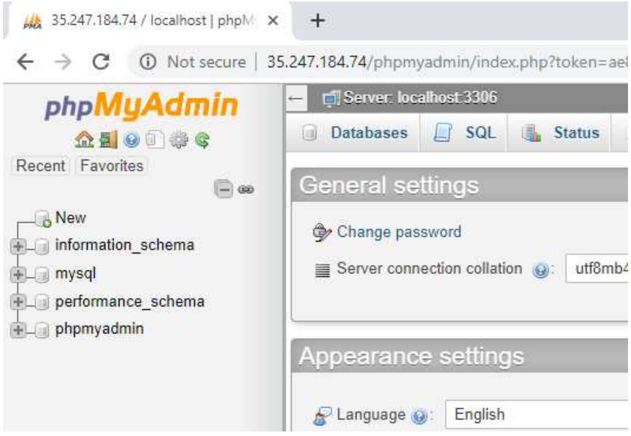
Figure 16. MySQL Service Interface

From there, you can proceed with creating a user account and granting it with all privileges as using the commands as shown below.



```
MariaDB [(none)]> create user 'fenad'@'%' identified by '03231979';
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> grant all privileges on *.* to 'fenad'@'%' with grant option;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> █
```

Figure 17. MySQL User Account Creation Interface



Server connection collation: utf8mb4

Language: English

Figure 18. phpMyAdmin User Interface

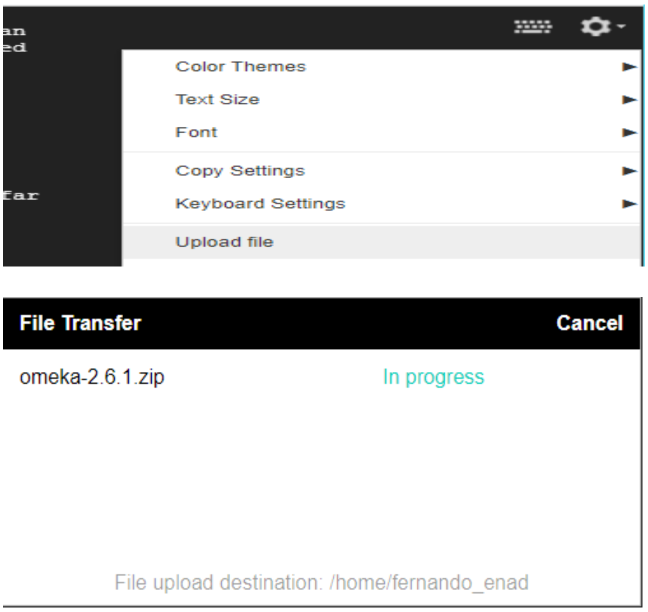
Congratulations, you have successfully installed the MySQL service and the phpMyAdmin component! Furthermore, the user account which you need to use on the next phase of this documentation has now been created.

VI. Installing Omeka Web Application

Now that you already have access to the phpMyAdmin portal, you are now ready to proceed with installing the Omeka Web Application. Omeka is a free, open-source content management system for online digital collections. To install this application, you would have to download it (Omeka Classic) first from the https://omeka.org/ site.

A. Uploading Installer File to VM Instance

The installer file is a compressed file (zipped) which you need to upload using the SSH interface which can be done by clicking on the gear button of the SSH window and selecting Upload File from the options.

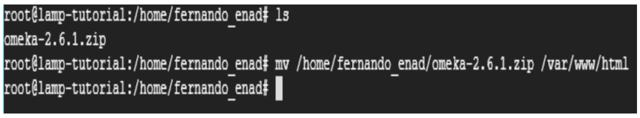


omeka-2.6.1.zip In progress

File upload destination: /home/fernando_enad

Figure 19. File Upload Interfaces of the VM Instance via SSH

Since the uploaded file has been moved to the /home/[user]/ path, you need to move it to the public folder at /var/www/html by using the **mv /home/[user]/omeka-2.6.1.zip /var/www/html** commands as shown.

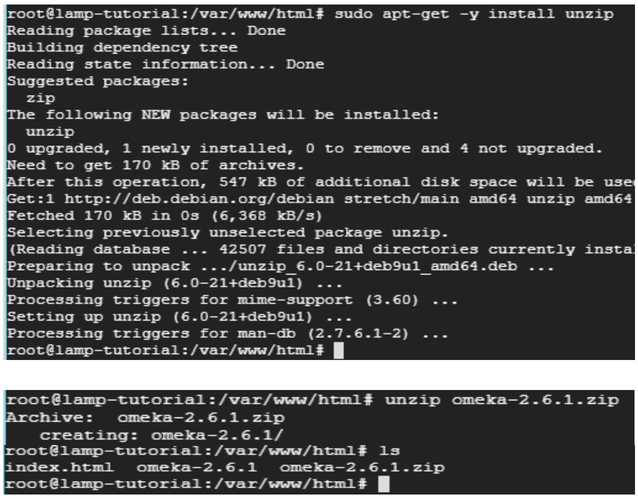


```
root@lamp-tutorial:/home/fernando_enad# ls
omeka-2.6.1.zip
root@lamp-tutorial:/home/fernando_enad# mv /home/fernando_enad/omeka-2.6.1.zip /var/www/html
root@lamp-tutorial:/home/fernando_enad# 
```

Figure 20. File Upload Interfaces of the VM Instance via SSH

B. Extracting Installer File to the Public Folder

After the successful transfer to the public folder, you then can have it extracted using the **unzip filename** command after installing using the **sudo apt-get -y install unzip** command as shown below.

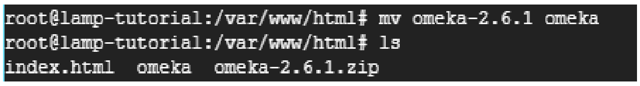


```
root@lamp-tutorial:/var/www/html# sudo apt-get -y install unzip
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 4 not upgraded.
Need to get 170 kB of archives.
After this operation, 547 kB of additional disk space will be use
Get:1 http://deb.debian.org/debian stretch/main amd64 unzip amd64
Fetched 170 kB in 0s (6,368 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 42507 files and directories currently insta
Preparing to unpack .../unzip_6.0-21+deb9u1_amd64.deb ...
Unpacking unzip (6.0-21+deb9u1) ...
Processing triggers for mime-support (3.60) ...
Setting up unzip (6.0-21+deb9u1) ...
Processing triggers for man-db (2.7.6.1-2) ...
root@lamp-tutorial:/var/www/html# 
```

```
root@lamp-tutorial:/var/www/html# unzip omeka-2.6.1.zip
Archive:  omeka-2.6.1.zip
   creating: omeka-2.6.1/
root@lamp-tutorial:/var/www/html# ls
index.html  omeka-2.6.1  omeka-2.6.1.zip
root@lamp-tutorial:/var/www/html# 
```

Figure 21. Zip File Extraction

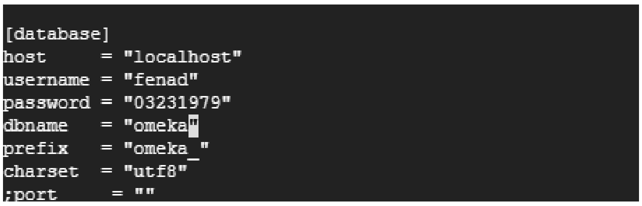
After a successful extraction, rename the generated omeka folder to omeka using the **mv** command as shown below.



```
root@lamp-tutorial:/var/www/html# mv omeka-2.6.1 omeka
root@lamp-tutorial:/var/www/html# ls
index.html  omeka  omeka-2.6.1.zip
```

Figure 22. File Upload Interfaces of the VM Instance via SSH

Get inside the omeka folder and make some modification to the db.ini using the **sudo nano** command as shown below.



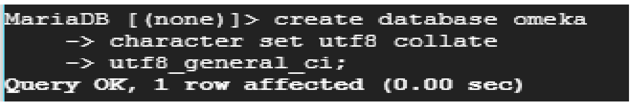
```
[database]
host     = "localhost"
username = "fenad"
password = "03231979"
dbname   = "omeka"
prefix   = "omeka_"
charset  = "utf8"
;port     = ""
```

Figure 23. Resource-based File Contents of db.ini

Once done with the modification, press Ctrl + O to save the changes and press on the Return [Enter] Key once and another Ctrl + X to exit from the editor and go back to the main SSH terminal.

C. *Omeka Database Creation*

Since the files have been extracted, you can proceed with creating a database named omeka (which you have specified explicitly in the db.ini file) via the MySQL console (see Figure 16 for the steps) using the **create database** command as shown below.



```
MariaDB [(none)]> create database omeka
    -> character set utf8 collate
    -> utf8_general_ci;
Query OK, 1 row affected (0.00 sec)
```

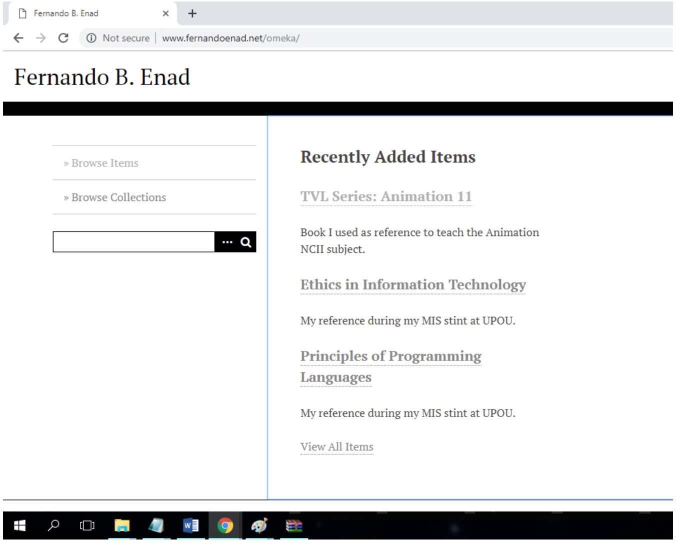
Figure 24. Omeka Database Creation

D. *Setting Up the Omeka Web Application*

Since the prerequisites have been complied, you are now ready to finally setup the Omeka Web Application and you can do so by accessing it via a browser using the VM instance's external IP (http://external_ip/omeka).

Accessing this url, should lead you to the setup and configuration of the Web Application. In the event of an installation error, like for example "mod_rewrite is not enabled", you may follow the steps provided below.

1. Execute the **sudo a2enmod rewrite** command in the SSH terminal.
2. Launch the site again and follow carefully the steps in setting up and configuring the Omeka Web Application.
3. Once completed and you are done supplying some contents, it should look like as shown below.



Fernando B. Enad

Recently Added Items

TVL Series: Animation 11

Book I used as reference to teach the Animation NCII subject.

Ethics in Information Technology

My reference during my MIS stint at UPOU.

Principles of Programming Languages

My reference during my MIS stint at UPOU.

View All Items

Figure 25. Successfully Configure Omeka Web Application

VII. Setting Up Cron Jobs

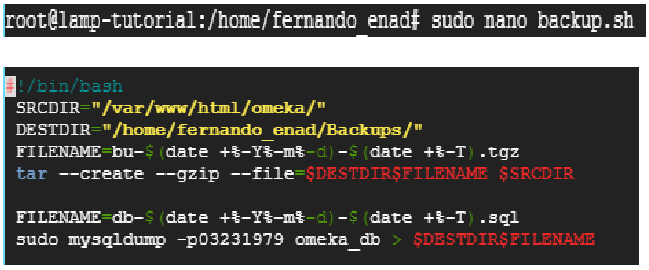
Cron jobs are the scheduled task themselves and they can be very useful to automate repetitive tasks.

A. *Scripts for Backup*

Cron jobs are based on the shell scripts that you want to execute repetitively.

1) *Backing Up of Files and Corresponding Databases*

For you to be able to create a shell script to backup files and corresponding databases, you must use the **sudo nano filename** command as shown below.



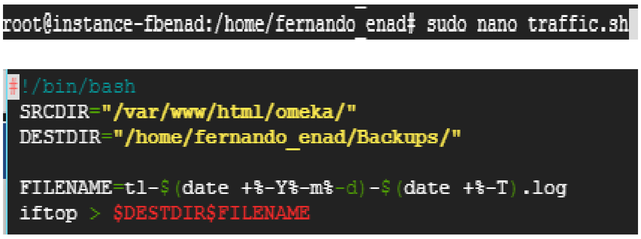
```
root@lamp-tutorial:/home/fernando_enad# sudo nano backup.sh
```

```
#!/bin/bash
SRCDIR="/var/www/html/omeka/"
DESTDIR="/home/fernando_enad/Backups/"
FILENAME=bu-$(date +%-Y%-m%-d)-$(date +%-T).tgz
tar --create --gzip --file=$DESTDIR$FILENAME $SRCDIR

FILENAME=db-$(date +%-Y%-m%-d)-$(date +%-T).sql
sudo mysqldump -p03231979 omeka_db > $DESTDIR$FILENAME
```

Figure 26. File and Database Backup Shell Script

2) *Backup of the Server's Network Traffic*

For you to be able to create a shell script to store the server's network traffic, you must use the **sudo nano filename** command as shown below. Before so doing though execute the **sudo apt-get install iftop** command to install the iftop component.



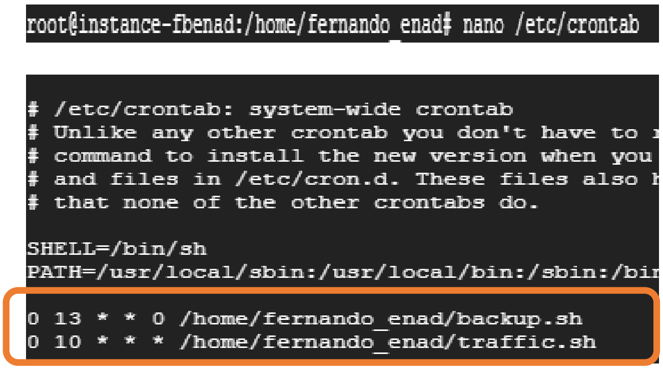
```
root@instance-fbenad:/home/fernando_enad# sudo nano traffic.sh
```

```
#!/bin/bash
SRCDIR="/var/www/html/omeka/"
DESTDIR="/home/fernando_enad/Backups/"

FILENAME=tl-$(date +%-Y%-m%-d)-$(date +%-T).log
iftop > $DESTDIR$FILENAME
```

Figure 27. Server Network Traffic to Text File Shell Script

B. *Cron Jobs*

To setup the cron jobs, you have to edit your **/etc/crontab** file as shown below.



```
root@instance-fbenad:/home/fernando_enad# nano /etc/crontab
```

```
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to
# command to install the new version when you
# and files in /etc/cron.d. These files also
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bi

0 13 * * 0 /home/fernando_enad/backup.sh
0 10 * * * /home/fernando_enad/traffic.sh
```

Figure 28. File Upload Interfaces of the VM Instance via SSH

On this setup, the backup.sh script will be executed at 1:00 PM every Sunday while the traffic.sh will be executed at 10:00AM daily.

REFERENCES

[1] https://www.debian.org/intro/about

[2] https://www.maketecheasier.com/choosing-the-best-linux-distro-for-a-web-server/

[3] https://www.ubuntupit.com/best-linux-server-distro-top-10-compared-recommendation/

[4] https://support.combell.com/en/how-do-i-add-cronjobs-using-ssh/479

[5] https://cloud.google.com/community/tutorials/setting-up-lamp