TME5- Scenario and Server Documentation

Web Server Management

Comp - 470- Scenario and Business Requirements

Winter, 2019

Submitted to: Mike Procter

Submitted by:

Agyapal Singh

Scenario:

In this scenario, there has been implemented PHP application for a school. With this application the administration unit could manage all student data those have done internships/coops. Also it can be said as an data storage for student in terms of records. This school is providing two different courses. For ?Business Studies? and for the ?IT Studies? and can be enhanced as well. This project is made up using MySQL and PHP using because Linux(Ubunto) is more popular and secure than windows and it has good networking functionality. This is indented for good, fast and reliable and secure date storage or information of students. This was a simple idea so that it can be easily maintained and used later on.

Students they are using the same database with a well formatted table. With this application they can store and pull up any students data by using the current student ID. This application is active on the web so that they could manage this from anywhere with an active internet connection. This is just like another idea of storing large number of movies and sorting them accordingly then by alphabetically and it can be a huge movie database. So this was a simple idea for so that student who want to work with professional companies and explore their skills and knowledge.

This was a basic idea that I decided to implement while using the PHP as in most schools student think to interact with Alumni so that they can know who went to internship and who did not. It can be further expanded by adding contact information and email id. This is for those students who study in IT or Business Studies or computer studies and want to go for an Internship. So they can get some help by connecting to students, maybe someone can refer them or can help them in anyway. So as per my previous experience and skills I used the following tools and technologies as discussed under:-

Business Justifications for Technologies used:

- PHP- The ability for employee and admin to create and manage data/information of students doing or done Coop.
- MYSQL- The ability to update data anytime from anywhere being on active server.
- PHP Application- Simple and friendly interface interaction for Admin and Students.
- AJAX - Ajax is used for fast and dynamic pages.

- Apache 2.0- Apache which is open server is used here.
- Ubuntu- Operating system used as Linux is more powerful and secure.
- Can be further expanded by giving access to each student and making this huge DB to connect within the school.
- It can be easily maintained and can be handled by career department.

Server Documentation

Configuration

1. VPS: A Fully equipped Virtual Private Server with an ?Ubuntu Linux 18.04? Operating System. This server has been bought from AWS.AMAZON.COM as I wanted to use a live server so in the research we did in our previous TME, I came to know we can have our online server with minimum requirements to produce this app.

Server Configurations:

- Virtual CPU 1 Core.
- 1 Giga Bytes of memory (RAM).
- 10 Giga Bytes of file storage.
- HVM Virtualization technology.
- Private IP: 172.31.92.116
- Public IP: 3.87.12.239

Since the application is very lightweight so this is the best fit for running this application. On the below image I?m showing the output of some basic information from inside the server.

2. Domain Name: This server is also having a FDQN (Fully qualified domain name) app.webmasterdomain.info and some alias for this same domain to run couple more things inside the same server. The internet domain name is webmasterdomain.info that has been bought from NAMECHEAP.COM. It was easy to get a name for the course I am doing and it will easy for me to use this information in the future to refer to my work.

Explanation of the Configuration Used:

1. DNS: This is an abbreviation of ?Domain Name System?. It has been invented till 1983. The main purpose of inventing this was giving a name to all the web servers IP addresses that are being used to run websites. With the help of Domain Name System it?s very easy to remember a website URL rather than remembering an ?Internet Protocol/IP Address?.

To avoid complexity I am using the default DNS server provided by the domain provider NAMECHEAP.COM. Here I have added some DNS records on the DNS zone.

I?m describing the use of the DNS records below:

- The ?A? records are usually used to point a domain or subdomain to an IP Address.
- The ?CNAME? records are usually used to create a subdomain for a domain to use contents from the same

server to load different website/webpage.

On the below image I?m showing how the host records are actually set on the DNS zone.

2. Web Server: In this Ubuntu Machine I am using Apache2 web server up and running to host the application online. Apache2 is the latest version of Apache. Apache is a very popular and it is very easy to manage. It has been installed from Ubuntu Bionic repository by using ?apt-get install apache2? command. On Ubuntu apache2 is shipped with lot of pre-installed useful modules to run the web server with a lot of features. On the below image I?m showing the current service status of the Apache2 web server.

3. Virtual Directory: In this server every web page has been installed on different Virtual Directory or Document Root instead of using the same Document Root for all the Website/Alias. The default Document Root for apache web server is ?/var/www/html?. It?s absolutely good to use the default Document Root to host a single application or website in one single server but if the goal is to run multiple Website/Alias from inside the same server then Virtual Directory comes handy. Virtual Directory is a part of Virtual Host/Virtual Server that will be demonstrated later.

In this scenario we are using three virtual directories to give a separate Document Root to all the server aliases. That has been described below:

a) ?/var/www/172.31.92.116/public_html? This virtual directory is containing nothing but the private IP address of the server is showing on the top to make it look separate from other PHP information output web page on this server. This is the default directory of this server. To reach this page the administrator should visit the public IP address of this server from any browser ?3.87.12.239? just like the image below.

b) ?/var/www/webmasterdomain.info/public_html? This is a virtual directory that contains the PHP Information output Since this page contains very sensitive data about the server so it?s protected with the HTTP Authentication. This is containing the domain alias on the top of that page to make it look separate from other PHP information output web page on this server. The administrator should visit ?webmasterdomain.info? or ?www.webmasterdomain.info? from any web browser to reach this page. The below image demonstrate this.

c) ?/var/www/app.webmasterdomain.info/public_html? This directory path contains the PHP Application. The directory index should load the application directly. To reach this page the administrator should visit ?app.webmasterdomain.info? from the browser. The below image demonstrate this.

4. Virtual Server: Virtual Server is well known as Virtual Host. It is used to make a single web server capable of hosting more than one website and make all of them seem to be hosting from different server. By creating Virtual Host on a web server it does make the web server capable of running different websites or web apps even in different IP addresses or in different ports from inside the same server.

In this scenario I?ve configured the Apache2 web server to three different Virtual Server or Virtual Host.

I?m demonstrating the configuration and use off all the Virtual Host below:

First of all I had to put the correct location of all the VirtualHost configuration file from apache2 main configuration file which is located in ?/etc/apache2/apache2.conf? I had to add ?IncludeOptional sites-enabled/*.conf? this line at the bottom of that page to locate the VirtualHost configuration files from the apache2 web server. Without doing this the VirtualHost configuration won?t work. The below image is the demonstration.

Then two directory has been created under ?/etc/apache2/? named ?sites-available? and ?sites-enabled?

All the Virtual Host configuration files 172-31-92-116.conf, webmasterdomain.conf, app-webmasterdomain.conf are made inside the ?sites-available? directory and they had been soft linked to ?sites-enabled? directory with the below commands.

?ln -s /etc/apache2/sites-available/172-31-92-116.conf /etc/apache2/sites-enabled/172-31-92-116.conf?

?ln -s /etc/apache2/sites-available/ webmasterdomain.conf /etc/apache2/sites-enabled/ webmasterdomain.conf?

?ln -s /etc/apache2/sites-available/ app-webmasterdomain.conf /etc/apache2/sites-enabled/ app-webmasterdomain.conf?

Usually the soft links had created a mirror of all the configuration files inside the ?sites-enabled? directory from the ?sites-available? directory. From further any modification of that files has to be done on ?sites-available? directory and it will be updated on ?sites-enabled? directory automatically. This was the main purposes of creating soft links.

All the Virtual Host configuration files are given below with images:

172-31-92-116.conf

webmasterdomain.conf

app-webmasterdomain.conf

On the images above there are some syntax. They are described below.

<VirtualHost *:80> defines that this Virtual Server is listening to port 80 for the incoming requests from any IP address of the server.

ServerName/ServerAlias defines the domain name or domain alias for the current virtualhost.

DocumentRoot defines the Virtual Directory for the current Virtual Server.

ErrorLog defines the apache error log directory.

CustomLog defines the apache log directory.

5. PHP: PHP is the abbreviation of ?PHP Hypertext Preprocessor?. It?s an object oriented programming language that is used to create web applications. Since PHP is a server side scripting language so the PHP code are being hidden from the regular users. The regular users can only see the HTML source code of a webpage. In this example this server is using PHP Version 7.2.

Running PHP with CGI Engine creates a new process for PHP inside the server and the web server runs in a different process. But in this example PHP is running with apache2 handler. That means the apache2 web server itself is responsible for running PHP. We can check this on the PHP Information file by locating the ?Server API?

6. MySQL: SQL is the abbreviation of Structured Query Language. In this scenario MySQL database management system has been used to manage the database for the PHP Application is installed on the server. This server is using MySQL Version 5.7.

MySQL is a very secure database management system and it?s possible to create different users for different database and grant different type of permission for all users. So it was the first choice for this server.

On the below image MySQL server version, some basic information about the MySQL Server and MySQL server status has been provided.

The PHP Application is using a database named ?students?. All privileges has been granted for the user ?webmaster? with a strong password and the table name is ?data?.

7. HTTP Authentication: HTTP Authentication is a server based authentication system that uses by the web servers. It?s a security feature provided by web servers to protect sensitive data from being viewed by regular users. When HTTP Authentication is enabled in a webpages will require valid Authentication before it gives anyone access to that current webpage. In this demonstration this server is using Basic HTTP Auth. That?s actually a Username and Password based authentication. Once anyone provides the valid username and password then they will be authorized as a valid user otherwise they will be redirected to an ?Access Denied? page.

By using HTTP Authentication all authentication data could be stored on a database but in this scenario the authentication is stored on a file inside the server that?s restricted for the normal visitors.

In this demonstration HTTP Authentication has been configured for domain ?webmasterdomain.info? and ?www.webmasterdomain.info?. ?webmasterdomain.info? is the alias of virtual server ?www.webmasterdomain.info? so the ?passwd? directory is created on the location below.

?/var/www/webmasterdomain.info/passwd?

After creating the directory an user ?admin? has been added with bellow command:

?htpasswd ?c /var/www/webmasterdomain.info/passwd/passwords admin?

Then it required to enter the new password twice like the image below. Once the password is provided then the user has been added on that file. Now it could be used for Authentication.

For security reason the password is not visible but the password has been successfully added.

As the next part the HTTP Basic Authentication has been configured at the web server. Below is the virtual server configuration file example.

Then after restarting the apache2 web server by ?service apache2 restart? command, the new configuration will take effect.

8. SSL Certificates: SSL is the abbreviation of ?Secure Socket Layer?. It?s an application layer security protocol that has been used to encrypt data transfer between the server and the client end. According to the generated key by a server, the SSL Certificate is being signed by the particular authority. There are a lot of SSL Certificate provider/authority on the web who provides SSL Certificate for websites but this server is using a Self Signed SSL Certificate.

To generate a self-signed SSL Certificate in a server ?openssl? package should be present on the server. The ?openssl? package should be installed on ubuntu using ?apt-get install openssl? command.

After installing the openssl package on the server the next step is generating a ?private.key? for the server. This file is very secret and should not be shared with anyone else. To create this file the below command has been used:

?openssl genrsa -out private.key 1024?

Here private.key is the servers private key file name and 1024 is the encryption bit.

The next step is generating a server.csr file. CSR means Certificate Signing Request. To create a valid SSL Certificate CSR is required. During the process of the CSR Generation this command has been used

?openssl req -new -key private.key -out server.csr?

And some information about the organization has been provided like:

Country Name, State or Province Name, Locality Name, Organization Name, Organizational Unit Name, Common Name, Email Address, A challenge password, An optional company name.

After generating the CSR finally it?s time to generate the SSL Certificate file. To do this below command has been used.

openssl x509 -req -days 365 -in server.csr -signkey private.key -out webmaster.crt

-req -days 365 means this certificate has been requested for 365 days. -in server.csr -signkey private.key -out webmaster.crt is the syntax used to create the ssl certificate webmaster.crt file by according to the CSR and using the servers private.key.

Once certificate generation is successful the SSL Certificate files has been stored on the directory:

?/var/www/ssl/www/?

Then the SSL certificate file location has been defined and enabled SSL from the web server and the server has been configured to listen on port 443.

Finally the SSL was enabled on the server by using the below command:

?a2enmod ssl?

9. PHP Application: The PHP Application has been installed on a separate Virtual Server and in Virtual Directory. The application is using php7.2, mysql server, php7.2-mysql module, And bunch of PHP and HTML codes.

This application is accessible using the below url:

http://app.webmasterdomain.info/index.php

With this application the administrator could easily add students data to the database just like the below example.

And then clicking the Submit button.

With this application it?s also possible to pull any student data by using only student ID just like the below example:

On the bottom of this PHP Application its showing all the student data in raw format.

Error Log

In the PDF embedded.

System Log

In the PDF embedded.

Access Log

In the PDF embedded.