TME6-WSM-3323130-Learning Journal Wiki

A PHP Application has installed on the server which is accessible from anywhere with an active internet connection. To run this application there has been used an Apache2 web server in a Ubuntu Linux 18.4. PHP Version has been used in this server is PHP 7.2. This is the same web server that has been installed and configured in Unit 5.

In this demonstration I will try to manipulate the web server by sending some raw HTTP command using the application.

The location of this application on the server is: ?/var/www/app.webmasterdomain.info/public_html/rawport.php?

and it?s accessible by using below URL.

?http://app.webmasterdomain.info/rawport.php?

1. Using HTTP 0.9

During this part I?ve filled the command box like the example below:

Host: localhost

Port: 80

Action: GET /

The output does not contain any header information, Content Type, HTTP Version so it?s not possible for the web browsers to display the content of the webpage. Most of the popular web browsers like Chrome has already been removed HTTP 0.9 from the updated versions.

2. Using HTTP 1.1

During this part the host was set to localhost just like the previous time. The Port was set to 80. And the below command has entered on the Action field and after that I did hit return twice.

GET / HTTP/1.1

Host: localhost

Accept: */*

There are a huge difference between the previous output and this output. It has returned all the details that?s very useful for a web browser to know in order to read the reply and giving output. This output contains content type, Server Version information, Last modified date and lot other information.

I?m providing the output below:

```
HTTP/1.1 200 OK
Date: Sat, 05 Jan 2019 22:59:44 GMT
Server: Apache/2.4.29 (Ubuntu)
Last-Modified: Sat, 05 Jan 2019 17:56:59 GMT
ETag: "78-57eb9b8e617bc"
Accept-Ranges: bytes
Content-Length: 120
Vary: Accept-Encoding
Content-Type: text/html
```

3. Redirection and default documents.

In this part I didn?t change anything on the Host and Port box. I?ve just changed the command on the action field and did hit return twice to keep the cursor at the right place.

```
GET /newdir HTTP/1.1

Host: localhost

Accept: */*
```

But instead of saying HTTP/1.1 200 OK it?s saying HTTP/1.1 301 Moved Permanently. To access the directory and load data from inside the directory there should one more ?/? to be added after newdir. Just like the below example:

```
GET /newdir/ HTTP/1.1

Host: localhost

Accept: */*
```

4. Security:

Till unit 5 I?ve created couple of virtual hosts and had implemented 1 or 2 exercise in the same document root. On that server there is a live domain name to reach the protected directory. The protected directory has been configured at the main document root. So here only the domain name has entered instead of entering localhost/dirname

For this exercise I did put the below command at the action field with hitting return 2 times at the end.

```
GET / HTTP/1.1

Host: webmasterdomain.info

Accept: */*
```

And expectedly the server didn?t give me access to the relevant content of that website. I got a reply from the server like below:

```
HTTP/1.1 401 Unauthorized

Date: Sun, 06 Jan 2019 16:13:14 GMT

Server: Apache/2.4.29 (Ubuntu)

WWW-Authenticate: Basic realm="Restricted Files"
```

```
Content-Length: 467

Content-Type: text/html; charset=iso-8859-1
```

The authentication method the server using is Basic. So, on the next step I?ve encoded the username and password in base64 and tried again. This time the data sent to server is provided in raw format below.

```
Port: 80

Location: webmasterdomain.info

Last data sent: GET / HTTP/1.1 Host: webmasterdomain.info Accept: */* Authorization: Basic
YWRtaW46YWRtaW4xMQ==
```

And the reply from the server was just like below:

```
HTTP/1.1 200 OK

Date: Sun, 06 Jan 2019 16:59:59 GMT

Server: Apache/2.4.29 (Ubuntu)

Vary: Accept-Encoding

Transfer-Encoding: chunked

Content-Type: text/html; charset=UTF-8
```

Actually base64 is not an encryption. It?s actually an encoding system instead, that has nothing to do with the security. It represents binary data. So, to escape the special characters on the password it?s important to encode the password first.

5. Non existent page:

In this part I?ve tried with a location of a page that actually does not exists on the server. Below I?m providing the commands I used and the output I got from the server:

Command on action field:

```
GET /nosuchfileexists.html HTTP/1.1

Host: localhost

Accept: */*
```

Return from server:

```
HTTP/1.1 404 Not Found

Date: Sun, 06 Jan 2019 17:10:53 GMT

Server: Apache/2.4.29 (Ubuntu)

Content-Length: 294
```

Content-Type: text/html; charset=iso-8859-1

That is a right reply from the server. It?s replying with 404 Not Found. Literally there is no file named nosuchfileexists.html on the provided location so the server has returned 404 Not Found. Actually this is a very common error code we do see on the web. When a requested web page or content does not really exists on the server then the server replies with 404 Not Found header.

6. Getting Pictures:

7. Caching:

In this exercise I?ve sent data to server by replacing the ?GET / HTTP/1.1? to ?HEAD / HTTP/1.1? and submitted.

This was the raw data that has been sent to server:

HEAD / HTTP/1.1 Host: localhost Accept: */*

And this is the command that was used to send this data to server:

```
HEAD / HTTP/1.1

Host: localhost

Accept: */*
```

The server replied interestingly with all the expected header but there were no page source and didn?t load any file from the server. For this command server has just replied with a HTTP/1.1 200 OK header along with other header information.

As mentioned I?ve tried this again by changing the content of that page and submitted again. This time the server replied with the same header but the Last-Modified value has been changed to the time when I was modified that file. That?s 5 minutes past from now. I?ve modified the content again and expectedly the Last-Modified value is changing every time if I change any content on the website.

Now I?ve tried with the host of www.athabascau.ca with Port: 80. The header says it has been 301 Permanently Removed. But when I visit this URL from the browser it forward me to the correct webpage. It means the http://www.athabascau.ca this URL has been permanently forwarded to https://www.athabascau.ca with HTTP 301 Permanent Redirection.

Now I?ve tried the same host with port 443. But it replied, Your browser sent a request that this server could not understand. Reason: You're speaking plain HTTP to an SSL-enabled server port. Instead use the HTTPS scheme to access this URL, please.

That really makes sense. This is a SSL enabled server so it won?t reply such requests.

8. What is this ?host? thing?

On the server there is only 1 document root for localhost and all default requests like any request with the server?s public IP address. So when I tried with localhost and 127.0.0.1 it has replied with the same content that it supposed to do.

So I?ve created two more Virtual Host. On unit 5 I?ve already created virtual hosts so I?m not showing how did I create the Virtual Hosts. I did create virtual hosts one for localhost and another for the loopback Ip address 127.0.0.1 and uploaded two different pages on both web root. Now tried again. Now the result was very interesting. When I tried with localhost then the server has replied with the new Virtual Host that?s used for the Virtual Host localhost. And when I tried again with the loopback address 127.0.0.1 it has replied with another web page that has been hosted on the loopback document root.