

waph-payyavsa

WAPH-Web Application Programming and Hacking

Instructor: Dr. Phu Phung

Student

Name: Sumanth Naga Payyavula

Email: payyavsa@ucmail.uc.edu



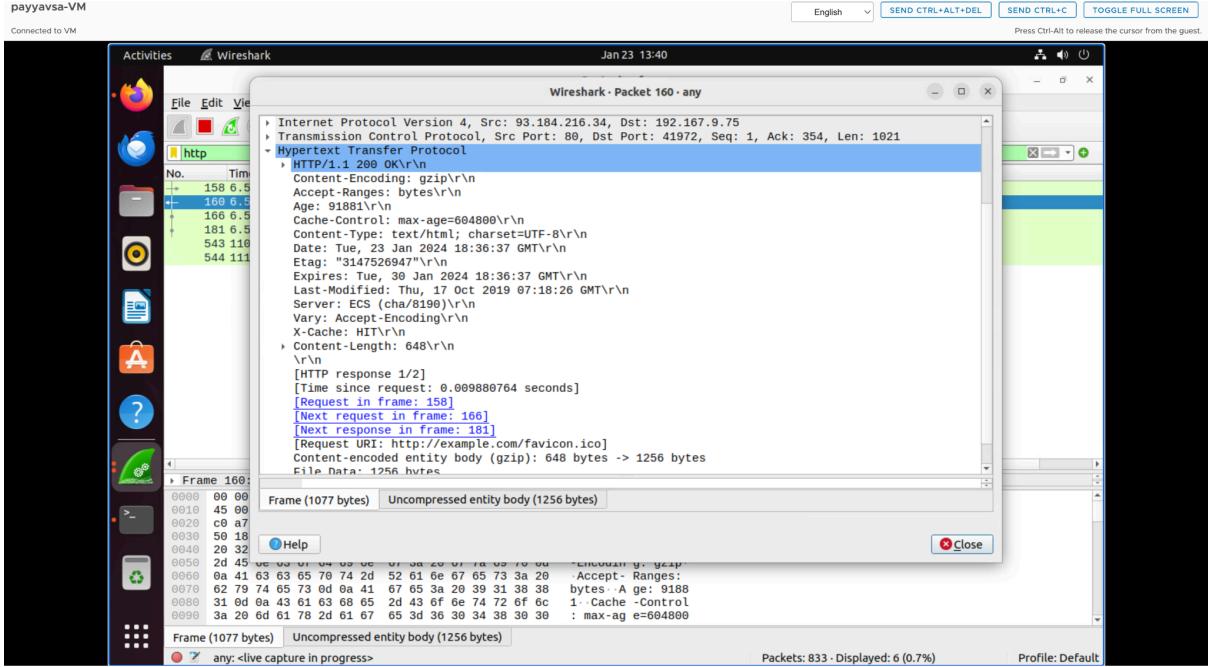
Lab 1 - Foundations of the Web

Part I: The Web and HTTP Protocol

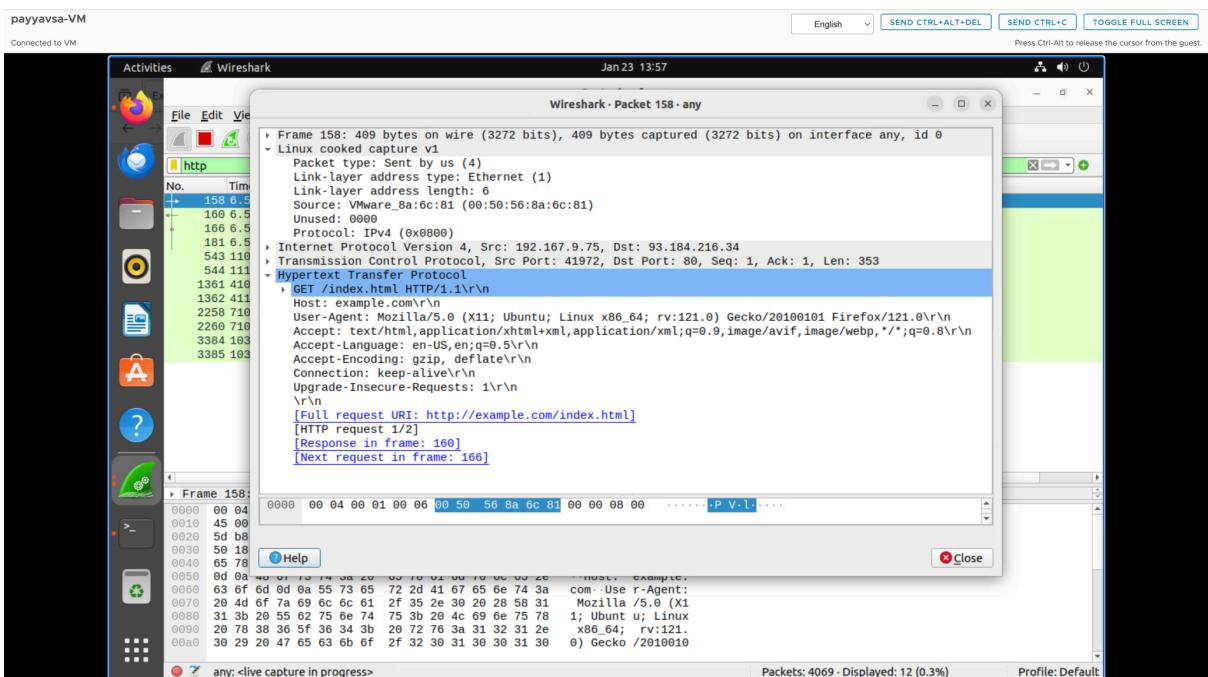
Task1: Familiar with Wireshark tool and http protocol

I selected the "any" option when I first launched the Wireshark application. After that, I could open any browser and search for any website using, say, Mozilla Firefox. Currently displayed are some http requests and responses.

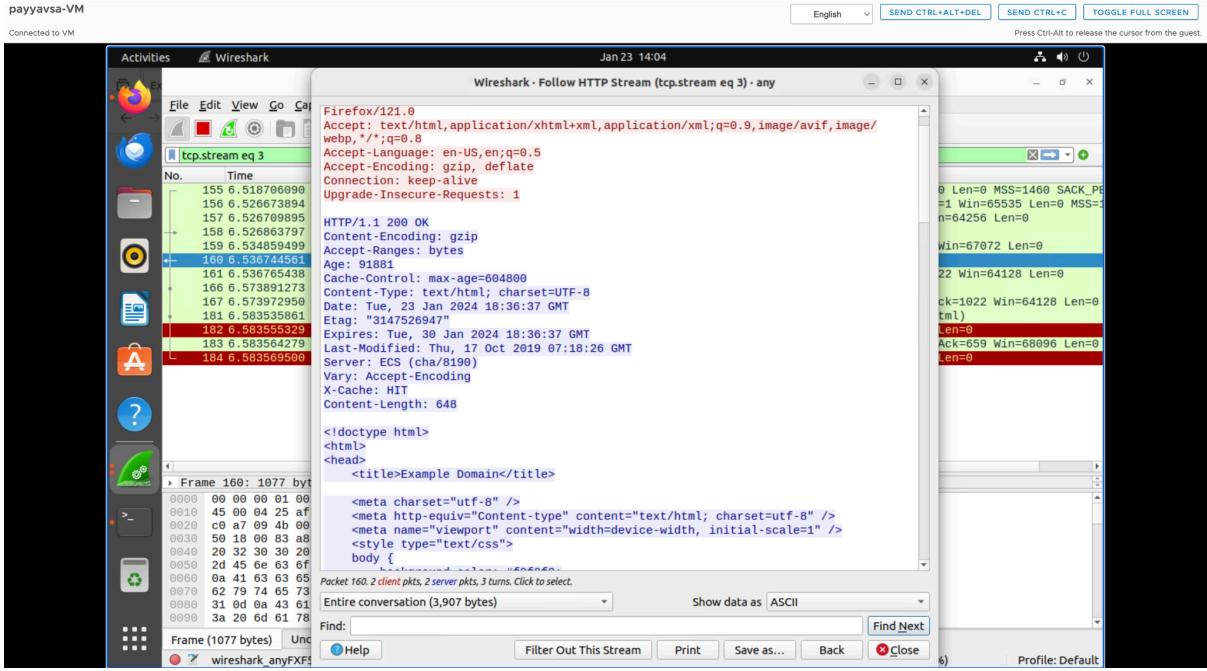
After that, you can access http stream as well. I've included a screenshot of the stream, the request, and the response below.



[1.1] http request



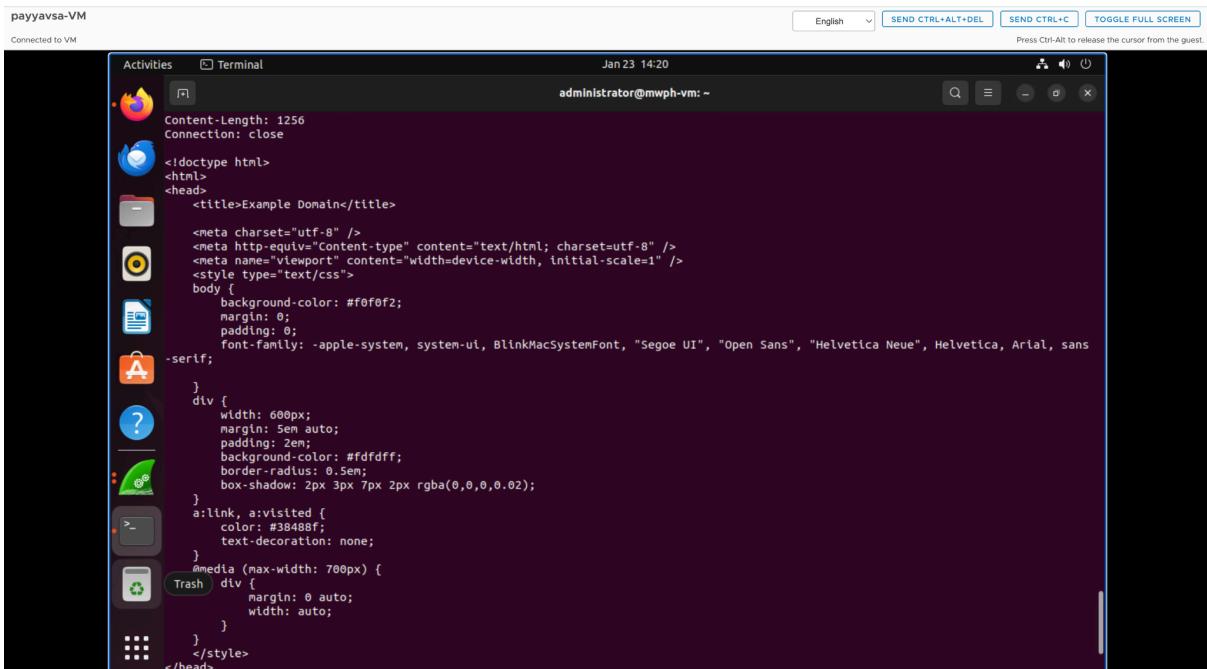
[1.2] http response



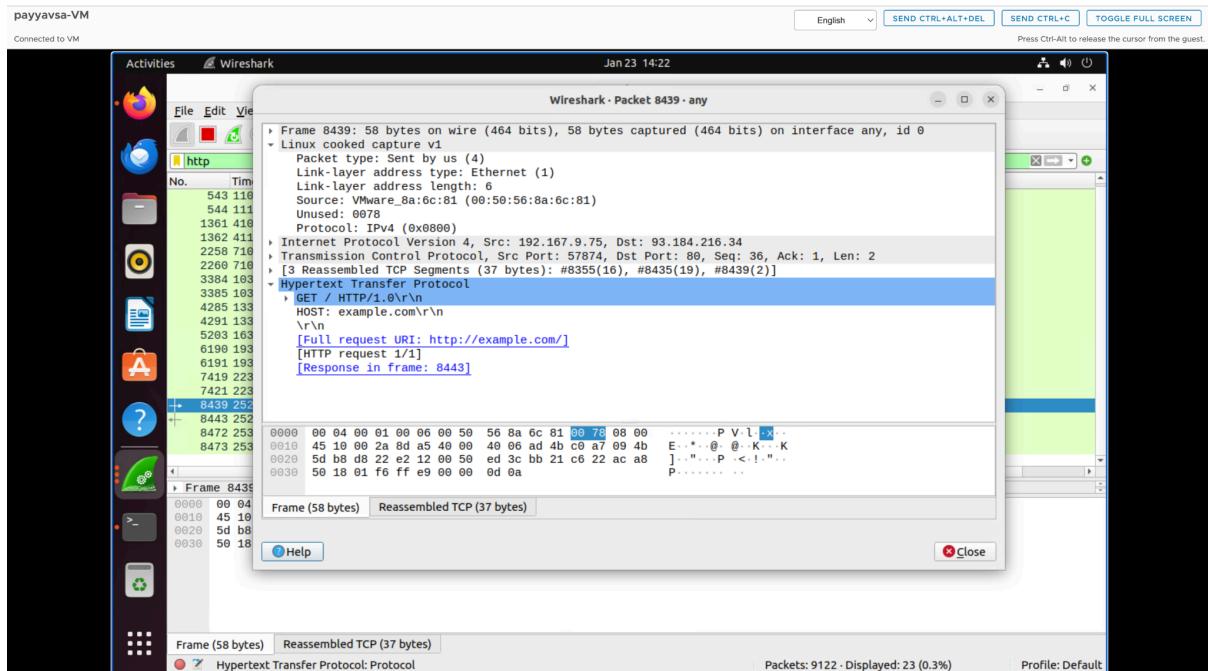
[1.3] http stream

Task 2

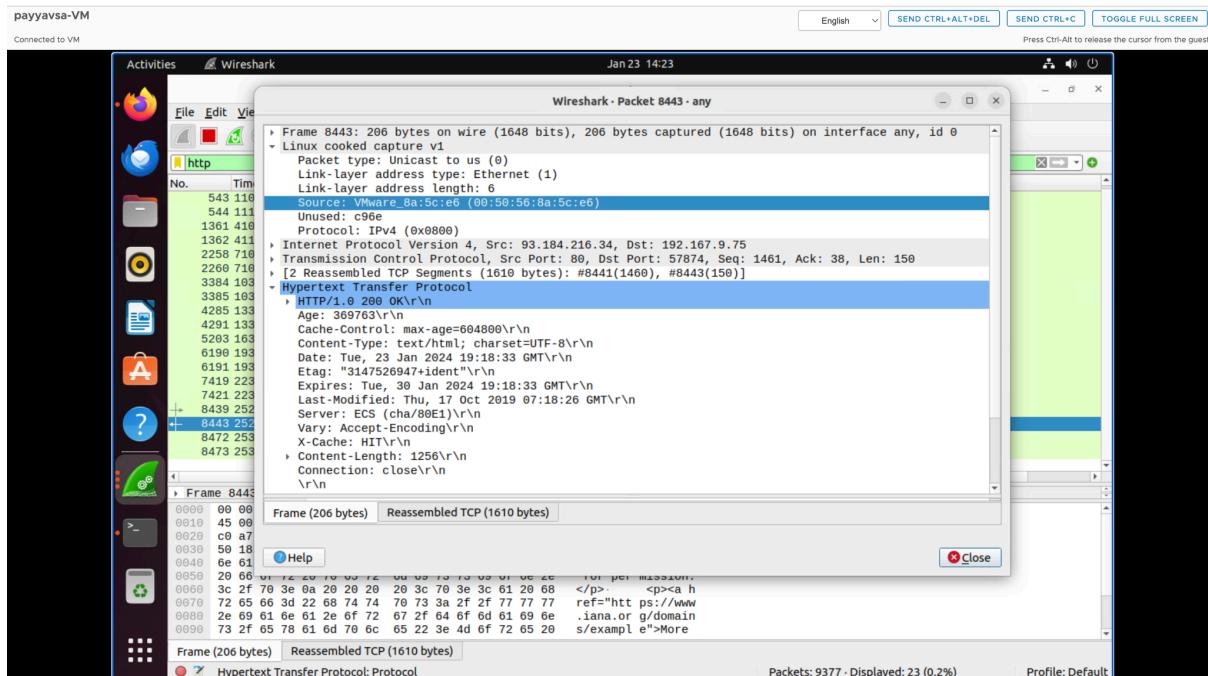
To utilize Telnet for HTTP assessments, launch a terminal and input the command "\$telnet example.com 80". Subsequently, input commands such as "GET /index.html /HTTP/1.0" and "Host: example.com"; this will display the HTTP request and response in the command prompt. These images are stored below.



[2.1] The output of running the commands listed in the description above is shown in the above snapshot.



[2.2] This snapshot resembles the http request that we completed in task 1 using Wireshark. We are given requests in telnet that contain only the GET and HTTP version along with the host site. However, wireshark contains some expert information in addition to request method, request URI, and version, such as GET information, severity level, and group.



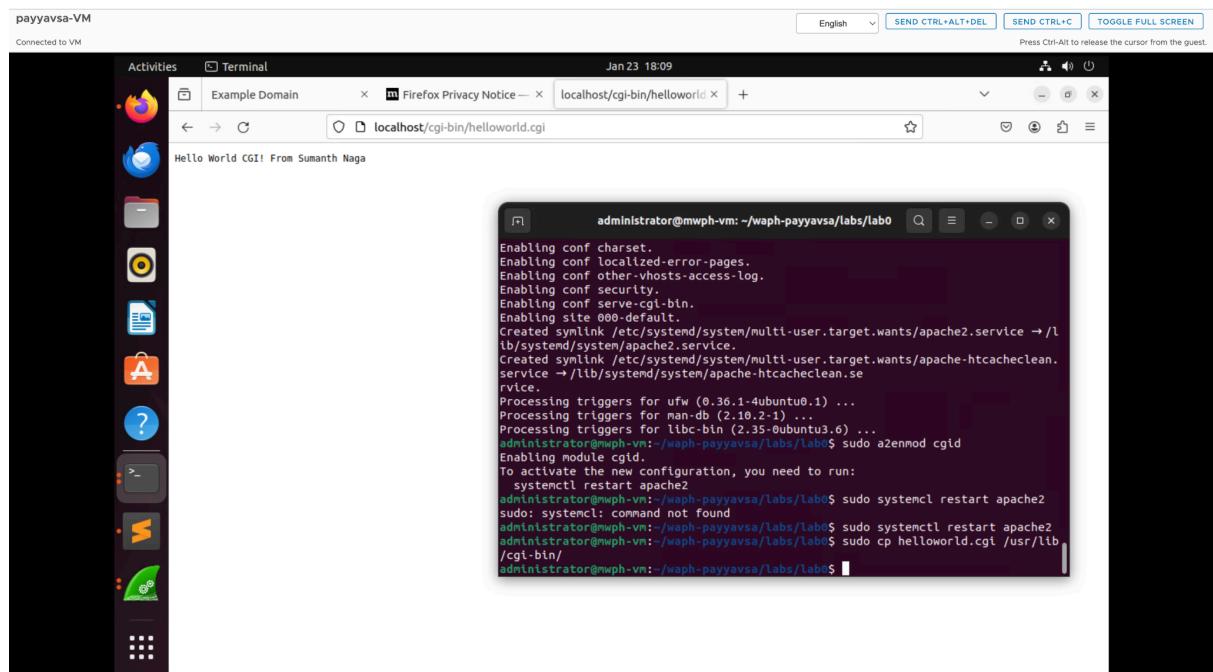
This snapshot resembles the http response that we completed in task 1 using Wireshark. We received a status code of "200" and an OK response over the telnet, along with the host

site's HTTP version. However, some expert information is available in Wireshark, including Response version, Status code, and Response Phrase. In essence, Wireshark provides us with comprehensive information about every single line.in addition to the content type, last modification date, expiration date, and time, along with the content length

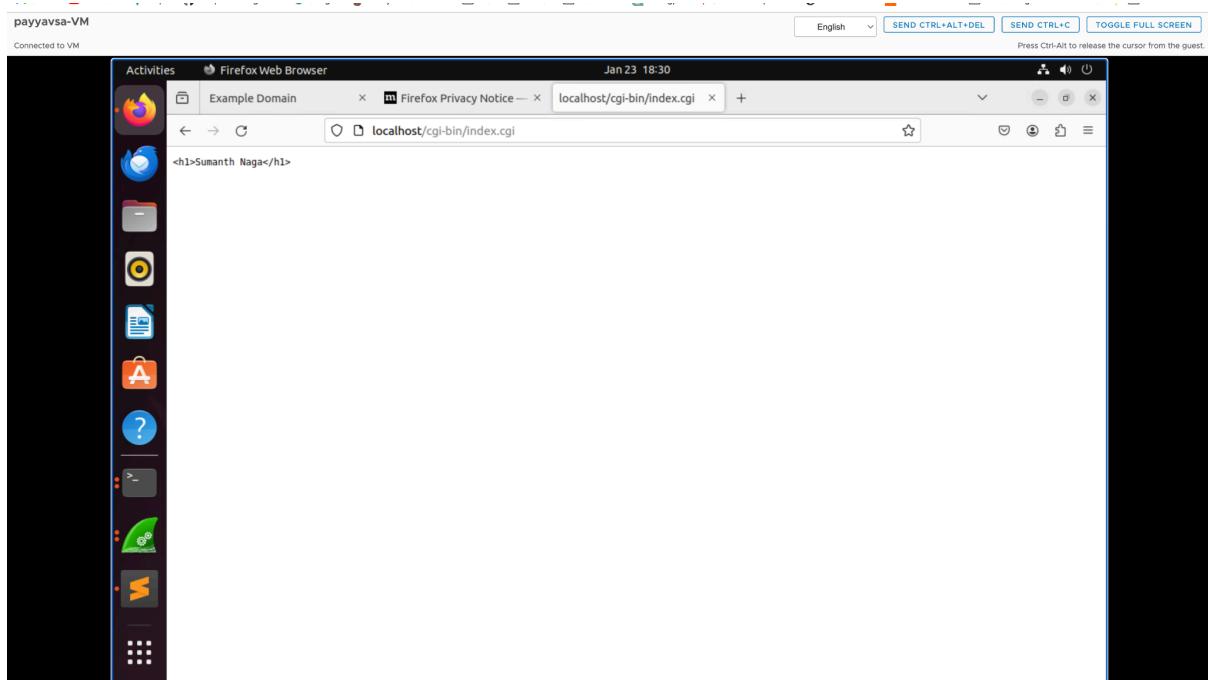
Part II: Basic Web Application Programming

Task 1:

The "\$ subl helloworld.c" command must be used to edit the CGI program in sublime before typing the code to deploy it on a web server. and use the gcc command and "/helloworld.cgi" to execute the code.The commands, along with \$ sudo systemctl restart apache2, are initially kept in a separate folder.We now need to type "localhost/cgi-bin/helloworld.cgi" in order to deploy the C code to the web server and view the output there.

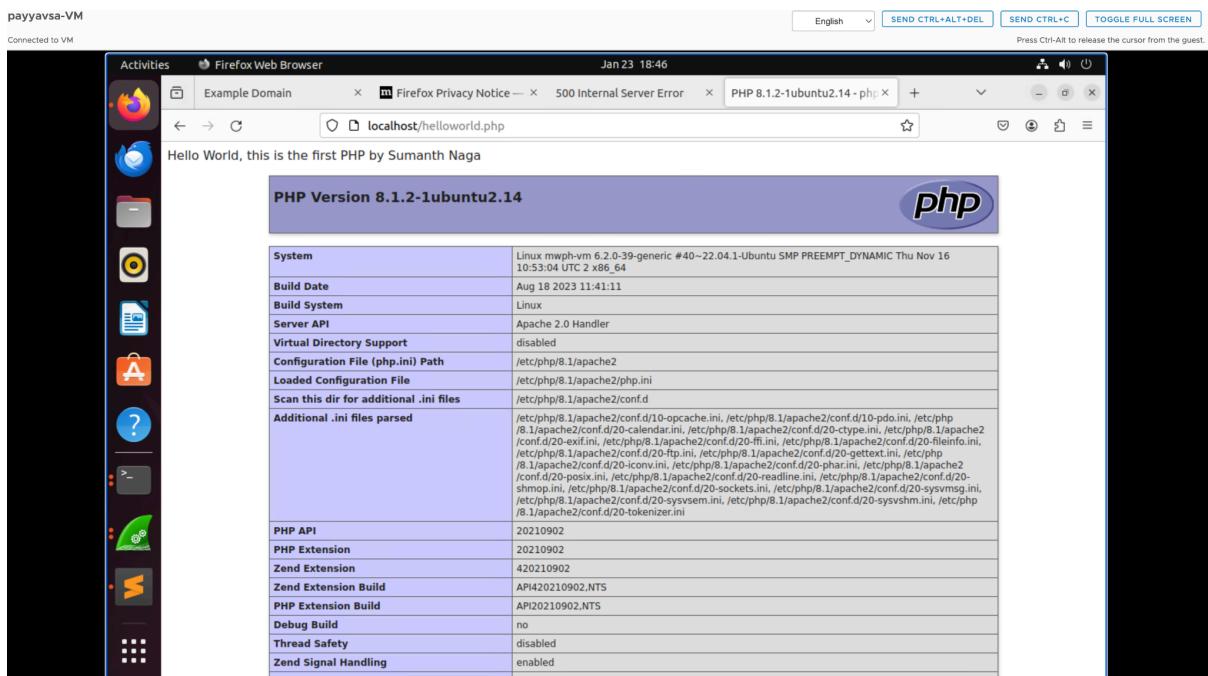


I've included a screenshot of the index.c file below, which contains a template and the link that our professor sent us.



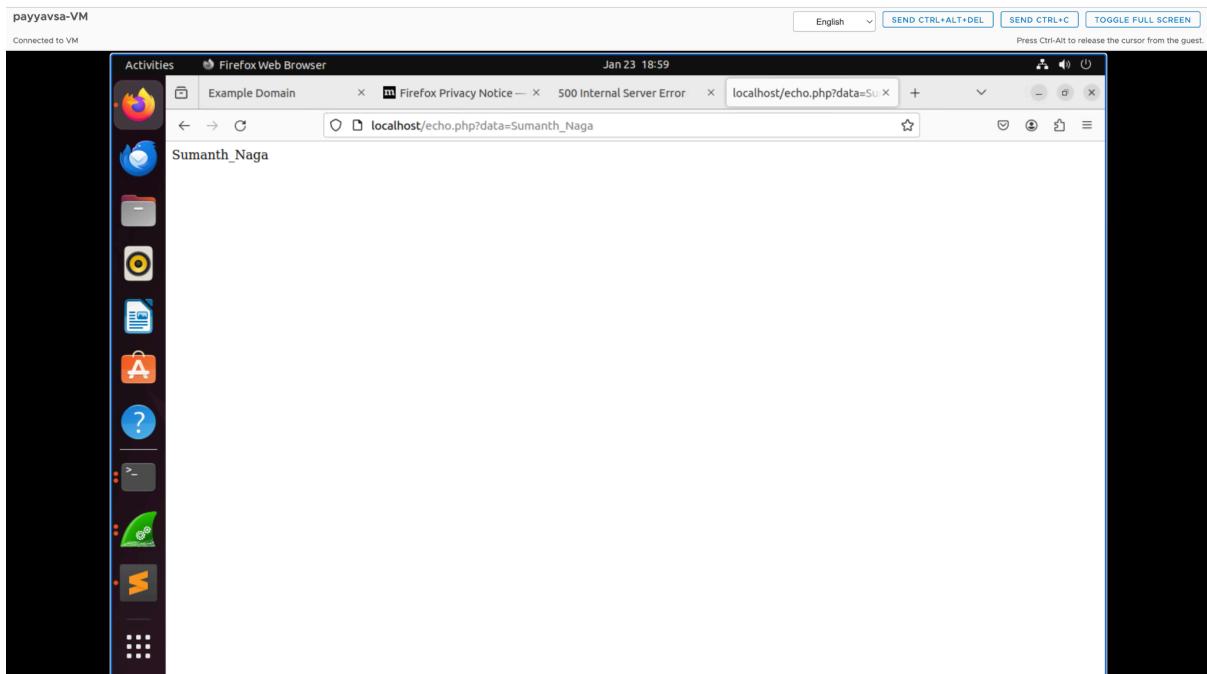
2. Task 2

Using Sublime Text, I created the basic helloworld.php file and uploaded it to the web server. To accomplish that, I need to type the following command in the terminal: "\$ sudo cp helloworld.php /var/www/html." Afterwards, open the "helloworld.php" file located on the localhost in any browser. I took a screenshot below.



Using Sublime Text, I created the basic helloworld.php file and uploaded it to the web server. To accomplish that, I need to type the following command into the terminal: "\$ sudo cp

helloworld.php /var/www/html." Afterwards, open the "helloworld.php" file located on the localhost in any browser. I've included a screenshot and the source code below.



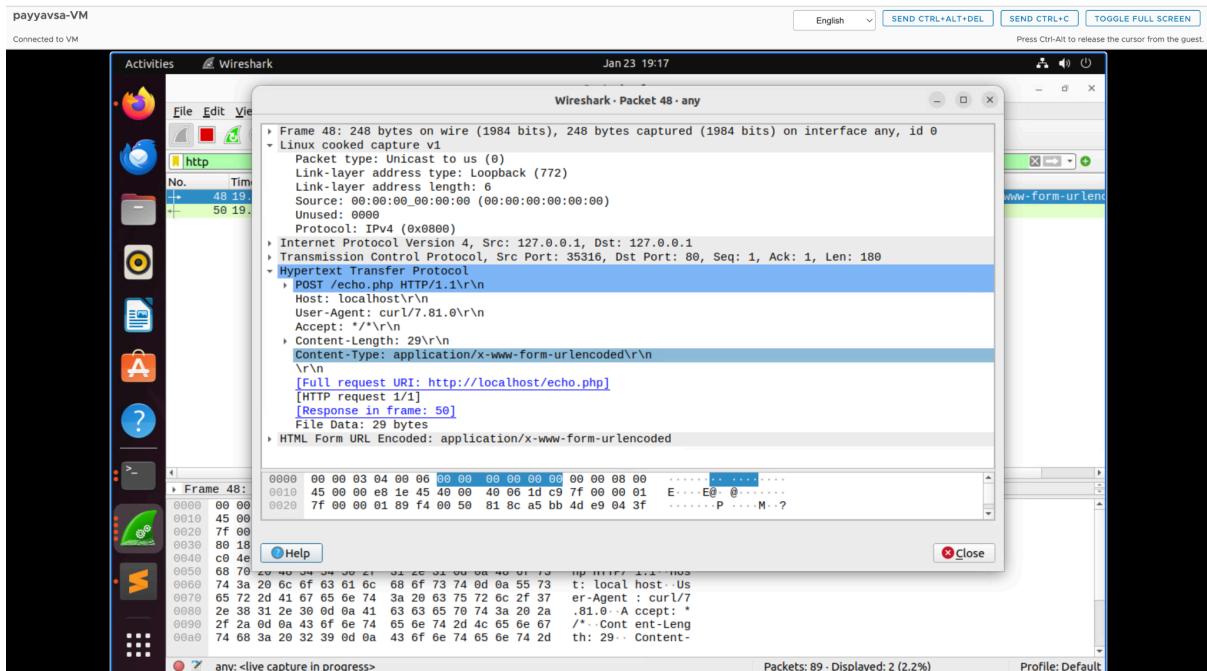
Source code: echo \$_REQUEST['data'];

Problem with REQUEST METHOD: This code or web application has a security risk since it can be attacked by hackers or attackers because their address is publicly visible.

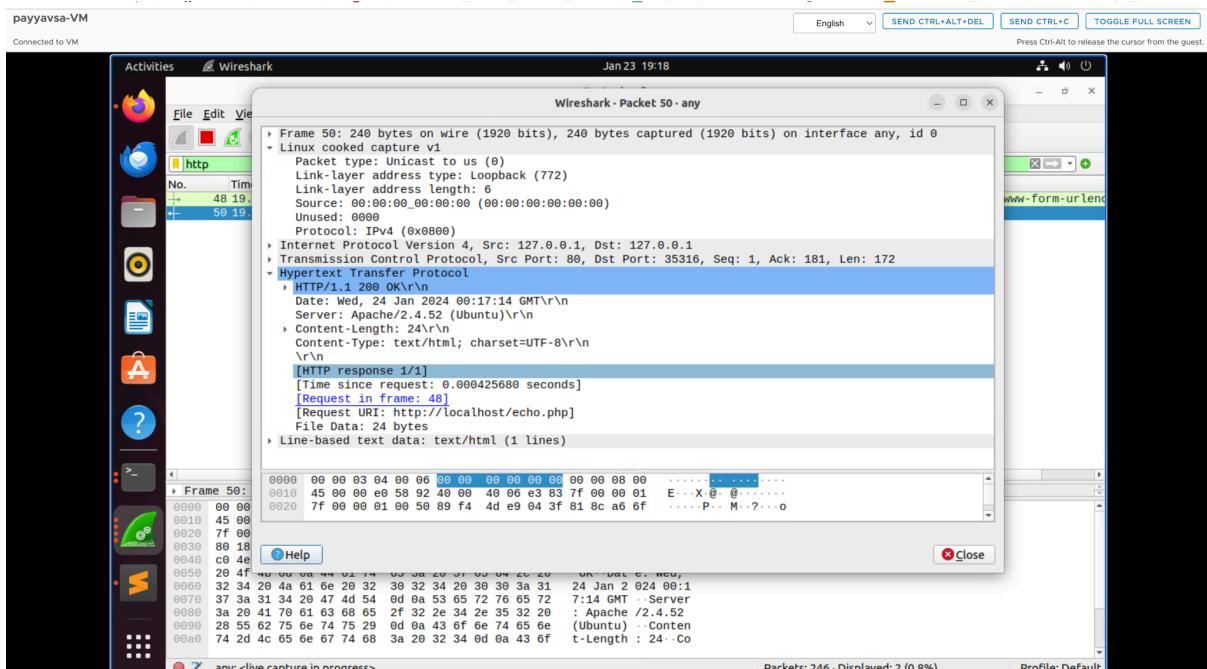
3. Task 3 (10 pts). Understanding HTTP GET and POST requests

a. I have examined the "echo.php" program's get request and response using wireshark.

Before starting the server, I selected any network option in Wireshark for that purpose. Later, I was able to obtain the request and response from Wireshark by using the http filter. I have retained the screenshots down below.

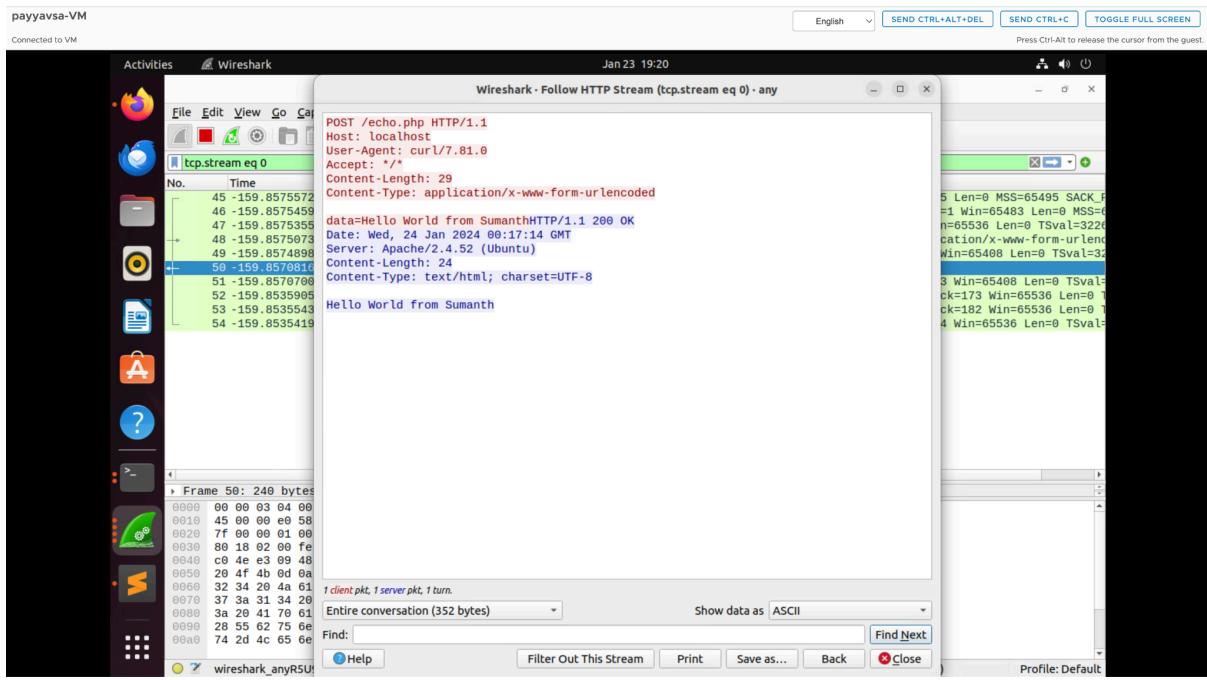


Here is http request of echo.php file



Http Response of echo.php file

b. I have initially set up the Ubuntu Curl application. Before opening Wireshark, I used a command that I previously described in the command prompt by using "sudo apt install curl." The http request and response of the application by using curl can later be clearly seen by right-clicking on it, which is followed by the HTTP stream. Below, I have included a screenshot.



Both the HTTP POST Request and Response follow the same cycle of requests and responses, and they both permit the addition of headers to the request to carry out extra information. Regarding the Differences. We have a lot of them such as-

- > We can observe that users can see the GET method in the browser address bar, while the POST method is not displayed there.
- > When it comes to capacity, GET is lower than POST.
- > The POST method is used for data submission that is sent in the request body, whereas the GET method contains data that was used for retrying when parameters like name are provided in the URL.