

WAPH-Web Application Programming and Hacking

Instructor: Dr. Phu Phung

Student

Name: Bhargavi Murari

Email: muraribi@mail.uc.edu

Lab 1 - Foundations of the Web

Part I: The Web and HTTP Protocol

1. Task 1 : I used Mozilla Firefox to search for “http://example.com” on a website and started Wireshark using the “any” option. Once the HTTP stream was accessible, Wireshark showed the HTTP requests and responses. Through the capture and presentation of the network traffic related to the web browsing activity, this made it possible to examine in detail the communication that took place between my machine and the web server.
2. Task 2 : To use Telnet to perform HTTP tests, I entered “\$telnet example.com 80” on my terminal. Following that, I typed the instructions’ “GET /index.html HTTP/1.0” and “Host: example.com” commands. The HTTP request I had made and the response I received from the server were then shown by the command prompt, giving me a glimpse of the exchange that was taking place between the computer and the example.com web server.

REQUEST: The HTTP request sent in Wireshark Task 1 is similar to the snapshot that was taken using Telnet. Although Wireshark offers more detailed information, Telnet queries still include necessary components such as the host site, HTTP version, and “GET” method. Beyond the fundamental elements observed in Telnet, Wireshark’s expert details include information regarding the “GET” request, severity levels, and arranged groupings.

RESPONSE: 1. Task 1 (10 pts). CGI Web applications in C 2. Task 2 (10 pts). A simple PHP Web Application with user input. 3. Task 3 (10 pts). Understanding HTTP GET and POST requests

Follow the detailed instructions in Lecture 3’s slides with hands-on demonstration during the class to complete this lab.

Report and deliverables

As prepared in Lab 0, you need to create a sub-folder `labs/lab1` with a `README.md` file to write a report in Markdown format and generate the

report to PDF using the **pandoc** application. All of the code from this lab must also be stored in this folder and included in the report if required. **Please note that the required screenshots must include your virtual machine name or your name with proper captions and be visible, e.g., not too blurry, for grading.** Your report should follow the template provided in Lecture 2 (<https://github.com/phungph-uc/waph/blob/main/README-template.md>) which should include the course name and instructor, your name and email together with your headshot (150x150 pixels), and the following sub-sections:

The lab's overview

(0 pts, -3pts if missing)

Write an overview of the lab and the outcomes you learned from this lab.

Also, include a direct clickable link to the lab folder on GitHub.com so that it can be viewed when grading, for example, <https://github.com/phungph-uc/waph-phungph/tree/main/labs/lab1>.

Part I - The Web and HTTP Protocol

Task 1. Familiar with the Wireshark tool and HTTP protocol

Write a summary of how you used the Wireshark tool to examine the HTTP protocol **(2.5 pts)** and demonstrate with three screenshots of the HTTP Request, Response messages, and HTTP Stream with proper captions **(2.5 pts each)**.

Task 2. Understanding HTTP using telnet and Wireshark

Summarize how you used the telnet program to send a minimal HTTP Request and the Wireshark tool to examine the HTTP messages **(2.5 pts)**. Demonstrate the tasks with the following screenshots with proper captions and explanations:

1. A screenshot of your terminal showing the HTTP Request (you typed) and HTTP response from the server. **(2.5 pts)**
2. A screenshot of the HTTP Request message (you typed in telnet above) in Wireshark as in Task 1. Is there any difference between this HTTP Request message and the one the browser sent in Task 1? Hints: What fields are missing in this request compared to the one the browser sent? **(2.5 pts)**
3. A screenshot of the HTTP Response message in Wireshark that the server responded to your request. Is there any difference between this HTTP Response message and the one in Task 1? **(2.5 pts)**

Part II - Basic Web Application Programming

Task 1. (10 pts) CGI Web applications in C

- a. Summarize how you developed a Hello World CGI program in C and compiled and deployed the program on the web server. **(2.5 pts)**. Demonstrate the task with a screenshot showing the CGI program is invoked properly in a browser. **(2.5 pts)**
- b. **(5 pts)** Summarize and demonstrate with a screenshot that you can write another C CGI program and deploy it with a simple HTML template provided on <https://www.w3schools.com/html/> with proper title, heading, and paragraph, i.e., the course and your information should be there. Include the source code of the file in the report. An example of code inclusion is below.

```
Included file helloworld.c:  C      #include <stdio.h>      int
main(void) {                printf("Content-Type: text/plain; charset=utf-8\n\n");
printf("Hello World CGI! From Phu Phung, WAPH\n\n");      return
0;                }
```

Task 2 (10 pts). A simple PHP Web Application with user input.

- a. **(2.5 pts)** Summarize and demonstrate with a screenshot that you have successfully developed a simple `helloworld.php` PHP page with your name and PHP configuration as guided in Lecture 3.
- b. Demonstrate that you developed and deployed an echo Web application in PHP, e.g., `echo.php` with a screenshot with your name in the data **(2.5 pts)**. Include the source code of the file in the report and discuss if there are any security risks in this simple web application. **(5 pts)**

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

- a. Briefly describe how you used Wireshark to examine the HTTP GET Request and Response for the `echo.php` page with your name in the data. Demonstrate with two corresponding screenshots in Wireshark. **(2.5 pts)**
- b. Summarize using `curl` to create an HTTP POST request with your name in the data. Demonstrate the outcome with a screenshot from the `curl` program **(2.5 pts)**, and a screenshot of the corresponding HTTP Stream in Wireshark. **(2.5 pts)**
- c. Compare the similarity/difference between HTTP POST Request and HTTP GET Request and between the two HTTP Responses above. **(2.5 pts)**

Submission

Use the `pandoc` tool to generate the PDF report for submission from the `README.md` file, make sure that the report and contents are rendered properly.

The PDF file should be named `your-username-waph-lab1.pdf`, e.g., `phungph-waph-lab1.pdf`