|  |
| --- |
| **PENETRATION TEST REPORT** |
| **Information Gathering and Vulnerability Assessment**  **Report** |
| Done by– Mohammed Amaan  Email: [mdamaan023@gmail.com](mailto:mudavathsindhu93@gmail.com) |

# Document Control

|  |  |  |
| --- | --- | --- |
| **Document Version** | **Owner & Role** | **Status & comments** |
| v1.0 | Mohammed Amaan – Penetration  Tester | Prepared the Internal Draft |

# Disclaimer

The content of this report is highly confidential and may include critical information on Information Gathering and Vulnerability Assessment. The report should be shared only with intended parties.

Although maximum effort has been applied to make this report accurate, Mohammed Amaan cannot be held responsible for inaccuracies or systems changes after the report has been issued since new vulnerabilities may be found o nce the tests are completed.

Moreover, Mohammed Amaan cannot be held responsible on how the report is implemented and changes made to DVWA Vulnerabilities . systems based on the recommendations of this report. Guidance should be taken from a network and security expert on how best to implement the recommendations.

All other information and the formats, methods, and reporting approaches is the intellectual property of Mohammed Amaan and is considered proprietary information and is provided in confidence to DVWA vulnerabilities and Information gathering for the purpose of internal use only.

# Table of Contents

Document Control

Disclaimer

Executive Summary

Scanning on Nmap

Scanning Methods

DVWA Vulnerability Tools

Detailed Findings

10.0.2.4

Conclusion

Recommendations

Additional Items

# Executive Summary

# SYNOPSIS

This report summarises the penetration test that was conducted by **Mohammed Amaan** as an assignment for the Cyber Security Bootcamp **Institute of aeronautical engineering , Hyderabad ,India .** This report is an attempt at showcasing my skillset and methodology of conducting a Web Application Penetration Test. The goal of this "pentest" was to act as a threat-actor by performing cyber-attacks and different vulnerabilities, an open-to-all web application. This will serve to discover any present vulnerabilities that could result in a breach and be leveraged to access sensitive data by a real-world attacker. All issues discovered by the pentester( Mohammed Amaan **)** are achieved and verified through network evaluation, system vulnerability scanning and assessment, and both automated and manual exploitation (where applicable) of found vulnerabilities.

# FINDINGS OVERVIEW

While conducting the external penetration test, there were several critical vulnerabilities discovered in the Information gathering. I was able to scan the Nmap vulnerabilities , DVWA vulnerabilities, and exploit XSS vulnerabilities. This could lead to grave information leaks and insecure sessions for users.

# SEVERITY SCALE

**CRITICAL Severity Issue:** Poses an immediate danger to systems, network, and/or data security and should be addressed as soon as possible. Exploitation requires little to no special knowledge of the target. Exploitation doesn’t require highly advanced skills, training, or tools.

**HIGH Severity Issue:** Poses a significant danger to systems, network, and/or data security. Exploitation commonly requires some advanced knowledge, training, skill, and/or tools. Issue(s) should be addressed promptly

**MEDIUM Severity Issue:**Vulnerabilities should be addressed in a timely manner. Exploitation is usually more difficult to achieve and requires special knowledge or access. Exploitation may also require social engineering as well as special conditions

**LOW Severity Issue:** The danger of exploitation is unlikely as vulnerabilities offer little to no opportunity to compromise the system, network, and/or data security. Can be handled as time permits.

**INFORMATIONAL Issue:** Meant to increase the client’s knowledge. Likely no actual threat

#### Penetration Test Report

**Final Report**

# Methodology

I utilized a widely adopted approach to performing penetration testing during the tests to test how well the target environment is secured. Below, a breakdown of the applied methodology is provided.

Information Gathering

Vulnerability Analysis

Exploitation

Post Exploitation

House Cleaning

* Information Gathering – Reconnaissance [Footprinting, Scanning and Enumeration]
* Vulnerability Analysis – Researching Potential Vulnerabilities and Analyzing them
* Exploitation – Using Exploits in order to validate the vulnerabilities of the target
* Post Exploitation – Everything that should be performed after successful exploitation
* House Cleaning – Ensuring that the Remnants of the Penetration Test are removed

# Tools Utilized

Tools used by me were Industry Grade in a combination of Open Source and Commercial Licenses.

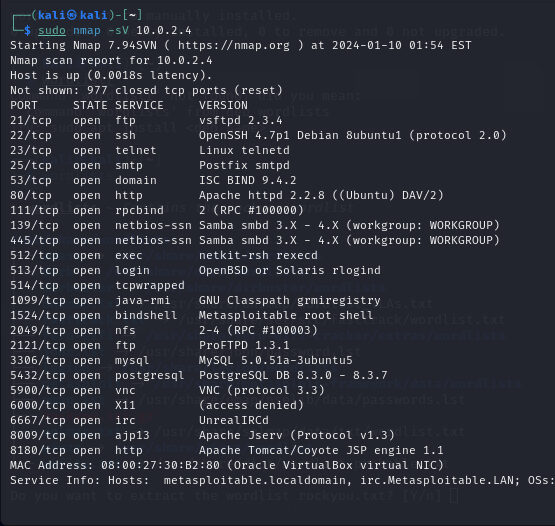
* 1. Nmap – Industry’s Most Commonly used Open-Source Scanning Tool
  2. Metasploit Framework – Industry Grade Most Popular Pen-Testing Framework Toolset
  3. BurpSuite Professional – Best in Class Suite of Tools for Web Application Assessment
  4. Linux terminal – Best tool to find the information about a**l**l the tools

# ENUMERATION

## STEP 1: INFORMATION GATHERING

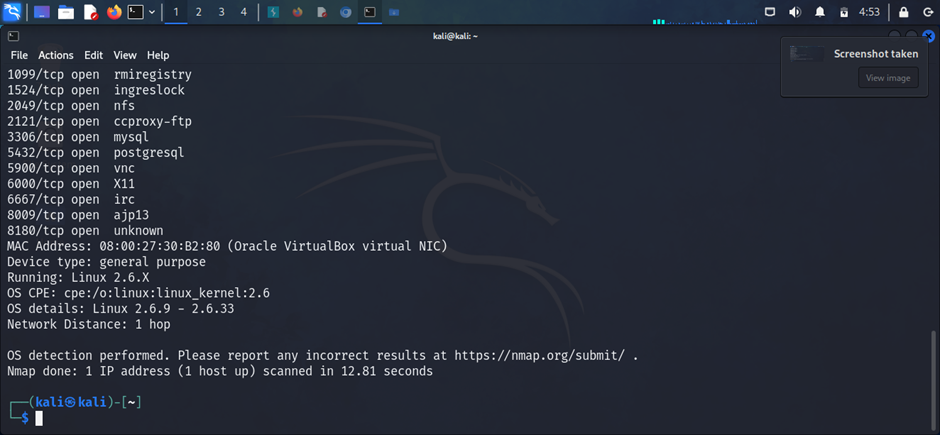
Performed a map service version enumeration. Using the command

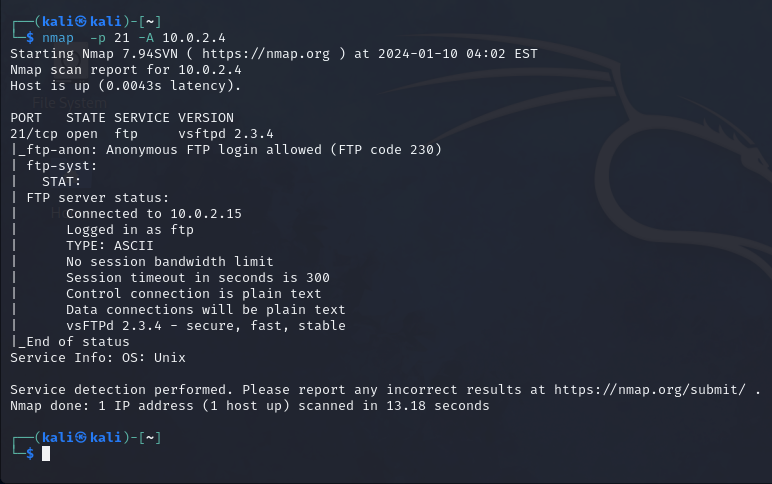
“nmap -sV 10.0.2.4” and have found out the ftp port 21 was active.



# Step-2: OS DETECTION ON Nmap

Found out the OS detection using the command “nmap -O 10.0.2.4”



**Step-3: SCANNING OF PORT**

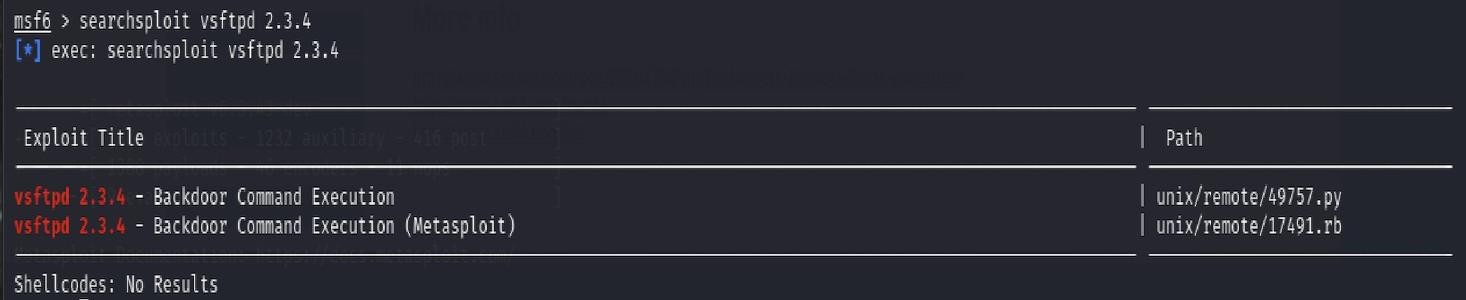
# Step-4: EXPLOITING VULNERABILITY

Exploiting Vulnerability in port no 21 by using “msf console”.



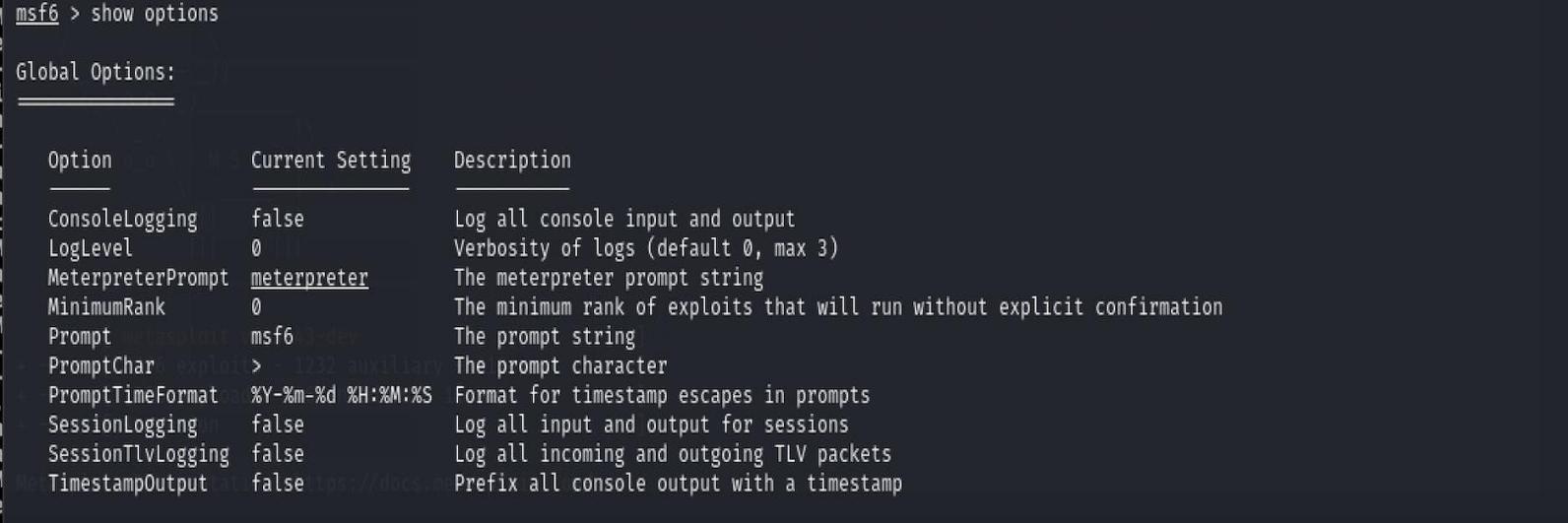
**Step-5: SEARCHING VSFTPD VERSION**

When conducting information gathering in cybersecurity, it can be beneficial to determine the version of VSFTPD (Very Secure FTP Daemon) running on a target system, as this information can help in identifying potential vulnerabilities or exploits associated with specific versions.

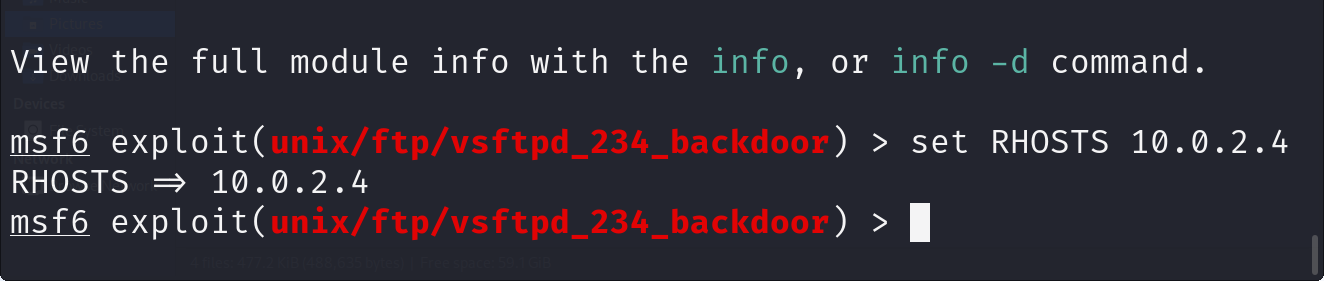
 

After msf console it shows the global options , those options are called as

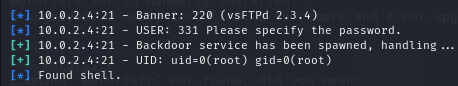
options or modules . After running "msfconsole," you will be presented with a console interface for the Metasploit Framework. This framework is widely used for penetration testing, vulnerability assessment, and exploitation.

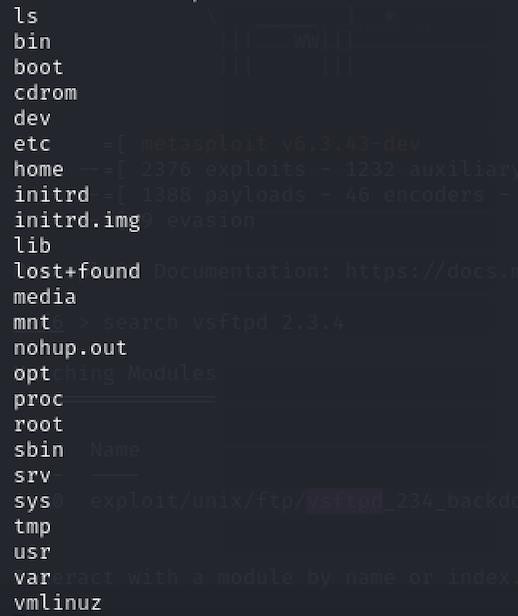


* When you set the RHOSTS parameter, you are essentially specifying the IP address of the remote system that you are attempting to exploit or interact with using Metasploit. This command sets the RHOSTS parameter to the specified IP address, enabling subsequent modules or exploits to target the specific remote host.



* Exploiting the vulnerability present in the port-21





**DVWA Vulnerability Scanning**

1. **COMMAND INJECTION :**
   * DVWA (Damn Vulnerable Web Application) is intentionally designed to be vulnerable for educational purposes, allowing users to practice and enhance their penetration testing skills in a legal and controlled environment.
   * **Command Injection** is one of the common vulnerabilities that can be found in web applications, and DVWA provides a platform to practice exploiting such vulnerabilities.
   * In DVWA, the Command Injection vulnerability allows an attacker to execute arbitrary commands on the underlying system by manipulating input parameters.

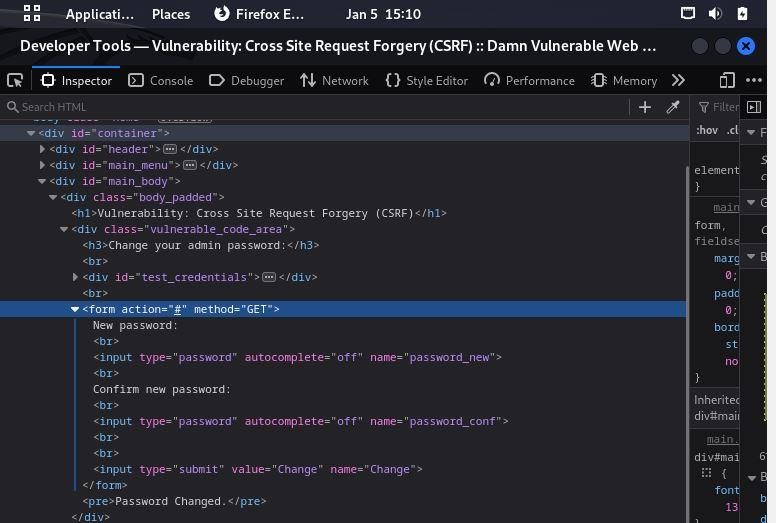


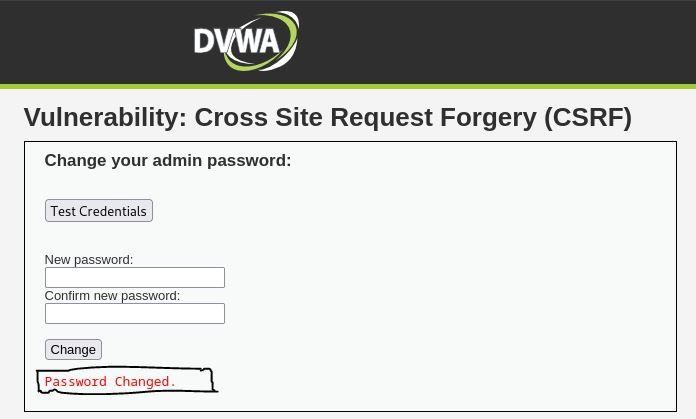
**DVWA Vulnerability Scanning**

1. **Cross Site Request Forgery (CSRF):**

Cross-Site Request Forgery (CSRF) is a web security vulnerability that allows an attacker to trick a user into performing actions they did not intend to perform. In the context of DVWA, CSRF vulnerabilities are intentionally included to allow users to practice identifying and exploiting such issues.

“”””http://10.0.2.4/dvwa/vulnerabilities/csrf/?password\_new=password&password\_conf=password&Change=Change#“””

By sending this url to an authenticated email user he/she can will click on the link , they try to login . If the url gets worked then actual user will get the message as “ pasword changed”.



**DVWA Vulnerability Scanning**

1. **Brute Force**

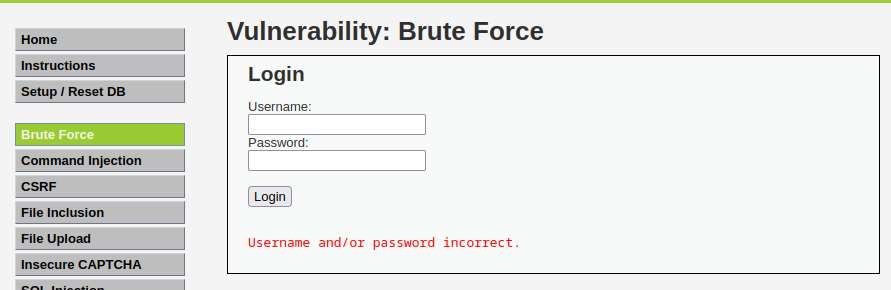
I have scanned for vulnerabilities by brute force attack

an attacker attempts to guess the correct username and password

combination by trying all possible combinations until they find the

correct one. This type of attack can be time-consuming and

resource-intensive, but it can also be effective in certain situations.

.

In the Burp Suite tool, I followed the path: **Target** → **Site**

**map** → **http://localhost →** URL Containing the

following: **/DVWA/vulnerabilities/brute/?username=admin&passwor**

**d=admin&Login=Login HTTP/1.1**

I selected the **Intruder** tab → **Positions**. And we can **Choose an**

**attack type, Add** or **Clear** payload markers, and **Start attack**.

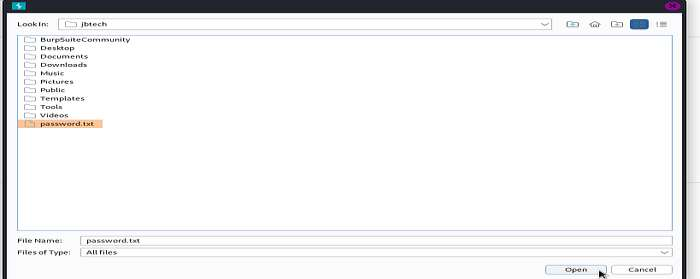
The password=**admin** is marked as payloads



**DVWA Vulnerability Scanning**

I selected the **password.txt** file, a list of common passwords I

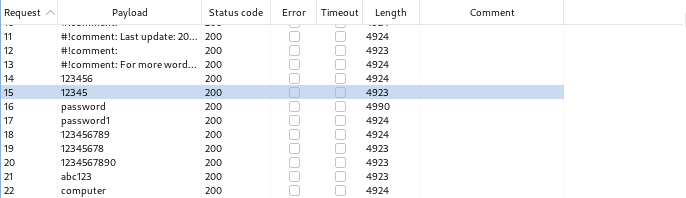
created.



The list of passwords inside the password.txt file is set.

Click **Start attack ,** here Burp Suite is performing a Brute Force

attack on DVWA.

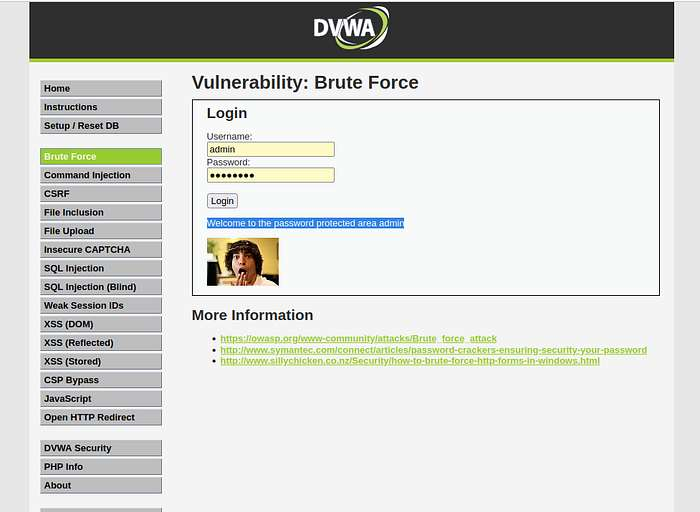


Payload = Password: **12345**

I typed the following username and password and clicked **Login**.

Username: **admin**

Password: **12345**



The Brute Force attack was successful.

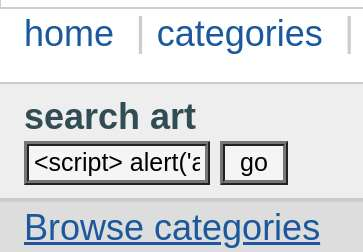
**4.XSS (Reflected)**

**Reflected XSS:**

I've tried inserting the string **<script> alert ('a')**

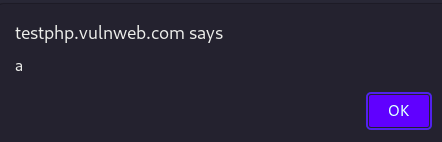
**</script>** into the text fields in the web app. The search box showed

that it could process the string



The following proves that the web application is processing XSS

requests without any sanitization.

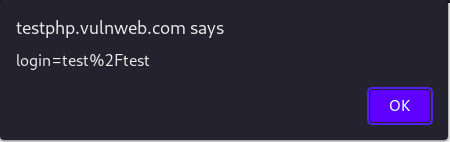


Creating an alert with the session cookie with the string **<script>**

**alert(document.cookie) </script>** reveals that the login credentials

are being revealed. This makes it extremely easy for a hacker to

steal login credentials through persistent XSS.



**Finding**: It was found that the search field was forwarding and

processing XSS requests without any sanitization. It was also found

that the session cookie reveals the login credentials of the logged in

user.

# Conclusion

Example Organization suffered a series of control failures, which led to a complete compromise of many in-scope machines. These failures would have had a dramatic effect on the company’s operations if a malicious party had exploited them.

The overall risk identified to Example Organization as a result of the penetration test is High. A direct path from external attacker to full network compromise was discovered. The fact that all 5 systems in scope were compromised makes it clear that these systems were not tested from a long time and since, they are all placed at the DMZ area, It’s a risky situation.

The primary goal of this penetration test was stated as identifying if there is any weakness in Example Organization’s Network that could potentially be used by attackers to access sensitive health (PHI) or payment data which would violate **HIPPA or PCI-DSS** compliances.

These goals of the pentest were met and in-fact much more than this. Many critical vulnerabilities were found during the test that directly affect confidentiality, integrity a nd availability of the information and systems. Majority of the findings have occasio nal prevalence, easy exploitability, and devasting impact with simple prevention.

It was found that your security architecture has few patterns:

* + Operating Systems are Outdated and Unpatched.
  + Softwares and Services are Outdated.
  + Passwords are either defaults or very weak.
  + Security Controls are either not defined or implemented in most cases.
  + All the vulnerabilities found have easy mitigation

In conclusion, these vulnerabilities should not be there in the first place. Example Corporation needs to redefine their Information Security Management Program and rethink their processes.

# Recommendations

Due to the impact to the overall organization as uncovered by this penetration test, appropriate actions should be taken to remediate and safeguard your IT infrastructure.

Though mitigation for specific vulnerabilities has already been given in this report,

#### Additionally, we recommend the following:

1. Establishment of Updates & Patch Management Program
2. Implementation of WAF and IPS
3. Source Code Review of Deployed Applications and Sanitization
4. Alignment of Security Policies with Industry’s Best Practices
5. Use a Custom 404 (Not Found Error) Page
6. Social Engineering training for every employee
7. Vulnerability Scanning on at least monthly basis (Scan – Patch – Scan Again)
8. Insta**l** a HIPS and DLP to stop common attacking payloads like meterpreter

# Additional Items

### Appendix A - References:

There are some concepts and special tools Iused, to which Ihave given the links below -

* + **Kali Linux -** [https://www.kali.org/do wnloads/](https://www.kali.org/downloads/)
  + **Vsftpd Exploit -** [rapid7.com/db/modules/exploit/unix/ftp/vsftpd\_234\_backdoor/](https://rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor/)
  + **Rooting Guide -** [blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/](https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/)

### Appendix B - Glossary:

There are some technical terms in the report which are important to be explained here -

* + **Black Box Penetration Test -** In penetration testing, black-box testing refers to a method where an ethical hacker has no knowledge of the system being attacked. The goal of a black-box penetration test is to simulate an external hacking. It is the most unreliable form of penetration testing.
  + **Social Engineering –** It is the art of using deceptio n to co n someone into providing information or access they would not normally have provided. It’s the “human side” of breaking into a network and preys on the qualities of human nature, such as the desire to be helpful, the tendency to trust peo ple and the fear of getting in trouble. According to recent statistics, 98%of all cyber-attacks rely on social engineering.