

Sri Lanka Institute of Information Technology



Assignment 02 Report  
IE3022

**Penetration testing report based on the lab  
work performed for the module.**

**Applied Information Security – IE3022  
B.Sc. (Hons) in Information Technology**

### Student Details

Student ID	Student Name
IT22560094	Ranasinghe R.A.D.T.M

## Contents

Overview .....	4
Introduction .....	4
Methodology.....	5
Scope of limitation .....	5
Red team Findings .....	6
Findings and Insights.....	6
1.    Software Vulnerabilities.....	7
2.    Network Vulnerabilities .....	19
3.    Web application vulnerabilities.....	25
Blue Team Analysis.....	30
Countermeasures against Software vulnerabilities .....	30
Countermeasures against Network Vulnerabilities .....	31
Countermeasures against Web Application Vulnerabilities .....	32
Purple Team Assessment .....	34
Effectiveness of Defensive Tactics and Controls .....	34
Protection Against Exploited Vulnerabilities .....	34
Business Impact Assessment .....	35
Impact of Software Vulnerabilities .....	35
Impact of Network Vulnerabilities .....	36
Impact of Web Application Vulnerabilities .....	36
Effectiveness of Present Controls .....	37
Strengths in the Existing Security Posture.....	37
Conclusion.....	38
References .....	39

## Overview

In this report, we present the findings and assessments resulting from a comprehensive penetration testing conducted by PentestRus on behalf of Mayo Industries. The test, which includes network and application assessments, was performed by respective teams which are red team, blue team and purple team each with different roles and responsibilities. This report offers insight into the potential vulnerabilities identified, anticipated threats, and recommendations for enhancing Mayo industry's security posture.

## Introduction

Mayo industries, a prominent organization within the sector, recognized the importance of securing its digital infrastructure and enlisted the expertise of PentestRus, which is a powerful digital security company provides Vulnerability Assessment and Penetration Testing (VAPT) services. This report marks a pivotal milestone in the collaboration between PentestRus and Mayo industries, aimed at safeguarding the organization's digital assets and intellectual properties.

In the preparation of the comprehensive penetration testing, our engagement was initiated with the assumption that a myriad of vulnerabilities may exist within the Mayo Industries network and applications. To ensure an accurate reflection of the potential risks faced by the organization, we selected several vulnerabilities to assess.

Software vulnerabilities like insecure designs, cryptographic failures, buffer overflows, and the unrestricted uploads of dangerous file types. Network vulnerabilities, implementation of weak firewalls, insecure wireless networks, and vulnerabilities related to incoming phishing emails and social engineering tactics withing those mails. Web applications vulnerabilities such as, SQL injections, cross-site scripting, cross-site request forgery, security misconfiguration, broken authentication, and path traversals.

It is important to note that while these assumed vulnerabilities are a significant focus of our assessment, the digital landscape is fraught with numerous other potential weaknesses that organizations typically face. This may include misconfigured systems, out-of-date network components, weak password policies, lack of security awareness training, and the possibility of insider threats. Our engagement with Mayo industries aims to holistically assess and identify these aspects as well, although the primary focus remains on the above-mentioned technical vulnerabilities.

Through this comprehensive report, we strive equip mayo industries withing the insights and recommendations required to improve its security measures and safeguard against both known and challenges in the ever-evolving digital landscape.

## Methodology

The red team is prepared to execute a multifaced approach to access potential technical vulnerabilities withing Mayo industries infrastructure. The assumption is that these vulnerabilities might exist, and we aim to simulate real world scenarios to examine the organization's security resilience. The following is the brief overview of the methodologies we assume will be employed,

*Software vulnerabilities*-we anticipate examining potential software vulnerabilities such as insecure design, cryptographic failure, buffer overflows, and the unrestricted uploads of dangerous file types. Vulnerable and outdated components will be checked for potential exploitation.

*Network vulnerabilities*-The assessment will extend to network security, withing the focus on the assumption of poor firewall configurations, insecure wireless networks, and security vulnerabilities related to incoming phishing mails and social engineering attacks. We will assume the presence of outdated or unpatched networks and will assess them for weaknesses.

*Web application vulnerabilities*-In the section of web application security, we assume the presence of vulnerabilities including SQL injections, cross-site scripting, cross-site request forgery, security misconfigurations, broken authentication and path traversals. Each of these vulnerabilities will be assumed for potential exploitations.

## Scope of limitation

The scope of this penetration test process helps to evaluate Mayo industries network and application security, by assuming the potential vulnerabilities exists. No specific zones are deemed off-limits, allowing for a thorough examination of the entire network. However, it's important to note that the focus is on assuming these vulnerabilities rather than risk management reporting at this stage. Additionally, this test is conducted with the following constraints: and limitations in mind.

**Time constraints**- this test is executed withing a specific timeframe.

**Scope limitation**-External factors beyond the network, such as physical security, are not assumed to be assessed in this report.

**Legal and ethical considerations**-all activities are assumed to be carried out within legal and ethical boundaries to ensure no disruption to Mayo industries operations.

These modified sections take into account that the vulnerabilities being addressed are assumptions rather than known issues and provide a clear framework for the rest of the report.

## Red team Findings

The red team assumed a curtail role in conducting a comprehensive assessment of Mayo industries infrastructure, focusing on the identification and analysis of potential vulnerabilities. Red team involves simulating a cyber-attack against an organization's network, applications, and systems to identify vulnerabilities and potential weaknesses. The goal of cyber red teaming is to objectively assess an organization's cyber security posture, identify gaps and vulnerabilities, and provide recommendations to improve the organization's security posture. Red Teaming is a proactive security assessment methodology that helps organizations to identify and mitigate security risks by simulating real-world attacks. In the context of Cyber Security and Ethical Hacking, Red Teaming involves a team of security experts attempting to penetrate an organization's systems and networks using the same tactics, techniques, and procedures (TTPs) as real-world attackers.[1]

Additionally, the red team assessed the severity of each vulnerability, considering the factors such as potential impact and ease of exploitation. They document their findings, providing a detailed description of each vulnerability's location and potential exploitation risks. The red team offered recommendations for remediation, guiding Mayo industries in strengthening their security posture and preventing future exploits. Their realistic threat simulation mirrored the tactics of actual threat actors, collaborating closely with the blue and purple teams to ensure holistic assessment.[2]

## Findings and Insights

The findings presented in this section reveal the vulnerabilities identified by the red team during their assessment of Mayo industries. These vulnerabilities can be categorized by software, network and web application, which are the main three digital parts of the organization. Each of these vulnerabilities are accompanied by a severity assessment, offering insights on the potential impact of exploitation. The read team's recommendations for remediation provide actionable steps to bolster the organization's security measures. These below red team findings will help to understand Mayo industries security strengths and weaknesses, which helps them to navigate ever evolving landscape of potential cyber threats and security challenges.[3]

## 1. Software Vulnerabilities

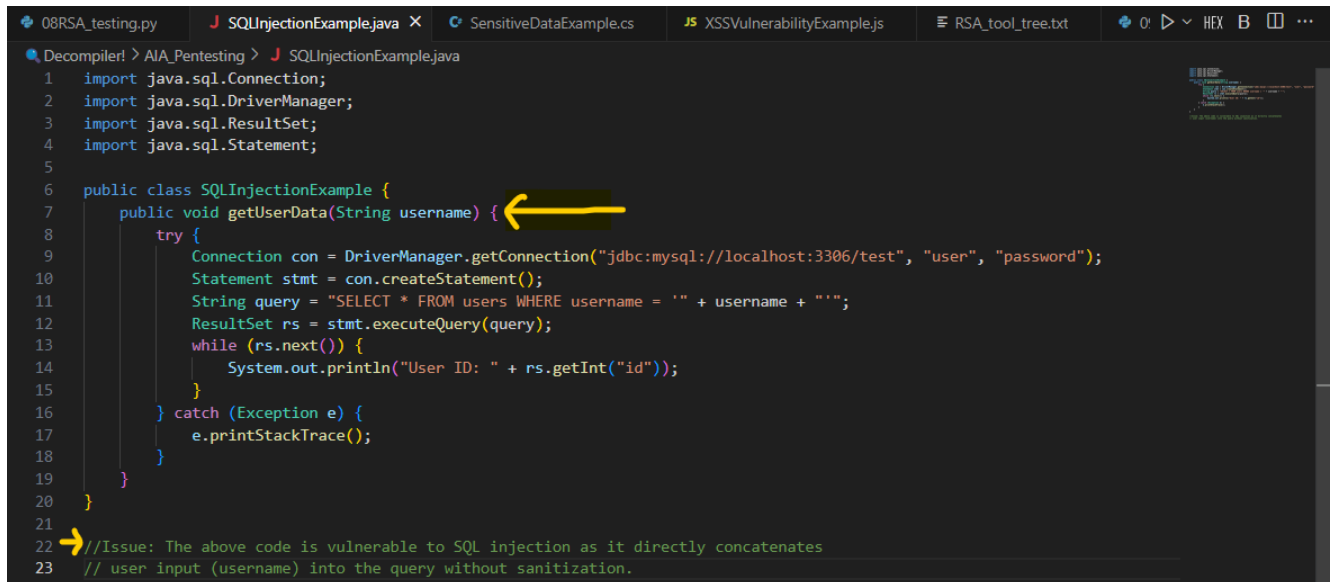
Vulnerability	Description	severity	Detecting tools and techniques
Insecure design	<p>Insecure design means weaknesses stemming from flawed software architecture and initial planning.[4]</p> <p>This vulnerability may occur because of improper access controls, inadequate threat modeling and insufficient security practices during the software design and maintain phases.</p>	<p><b>High.</b></p> <p>Insecure designs can lead to pervasive security issues that are challenging to mitigate once software is developed.</p>	<ul style="list-style-type: none"> <li>• Manual code reviews. (Image 01) (image 02)</li> <li>• Secure code review tools like Checkmarx and Fortify.</li> <li>• Maltego tool for information gathering and reconnaissance. (Image 03) (image 04)</li> <li>• Recon-ng tool for data discovery and reconnaissance. (Image 05)</li> <li>• Dn-spy tool for modify insecure software codes (image 06)</li> </ul>
Cryptographic failures	<p>Cryptographic failures include issues with encryption and decryption processes.</p> <p>This can result from using weak encryption algorithms, improper key management, or inadequate protection of cryptographic keys.</p>	<p><b>High.</b></p> <p>Cryptographic failures can expose sensitive data, leading to data breaches and potential regulatory violations.</p>	<ul style="list-style-type: none"> <li>• Nessus. (Image 07)</li> <li>• Wireshark [5] for analyzing SSL/TLS traffic. (Image 08)</li> <li>• John the ripper for password cracking. (Image 09)</li> <li>• Hydra for password cracking including SSH. (Image 10)</li> </ul>
Buffer Overflow	<p>Buffer overflow vulnerabilities occur when a program writes more data to a buffer than it can hold.</p> <p>That potentially leads to memory corruption or the execution of malicious code.</p>	<p><b>High.</b></p> <p>Exploiting buffer overflows can result in unauthorized code execution and also leads to compromise the software systems.</p>	<ul style="list-style-type: none"> <li>• Valgrind (image 11)</li> <li>• Immunity Debugger (image 12)</li> <li>• Maltego for information gathering and reconnaissance (image 03)</li> </ul>
Unrestricted Upload of	<p>This vulnerability arises when an application allows the unrestricted</p>	<p><b>Moderate to high.</b></p>	<ul style="list-style-type: none"> <li>• OWASP ZAP for analyzing file upload vulnerabilities. (Image 13)</li> </ul>

Dangerous File Types	uploading of files without proper validation. Attackers can upload malicious files to compromise the system.	Depending on the files uploaded, this vulnerability can lead to various levels of compromise.	<ul style="list-style-type: none"> <li>• Burp Suite [6] for intercepting and modifying file uploads (image 14) (image 15) (image 16)</li> <li>• Metasploit for testing file upload vulnerabilities (image 17)</li> <li>• The Harvester for gathering email addresses and information (image 18)</li> </ul>
Vulnerable and Outdated Components	Vulnerable and outdated components refer to the usage of software libraries, plugins, or modules with known vulnerabilities. Those haven't been updated or patched so it can lead to trigger security flaws in software's.	<p><b>High.</b></p> <p>Exploiting known vulnerabilities in components can lead to system compromise.</p>	<ul style="list-style-type: none"> <li>• OWASP Dependency-Check</li> <li>• Retire.js for JavaScript libraries</li> <li>• Snyk for identifying and fixing vulnerable dependencies</li> <li>• Netcraft for website scanning and analysis (image 19)</li> <li>• Shodan [7] for identifying exposed systems and vulnerabilities (image 20)</li> </ul>



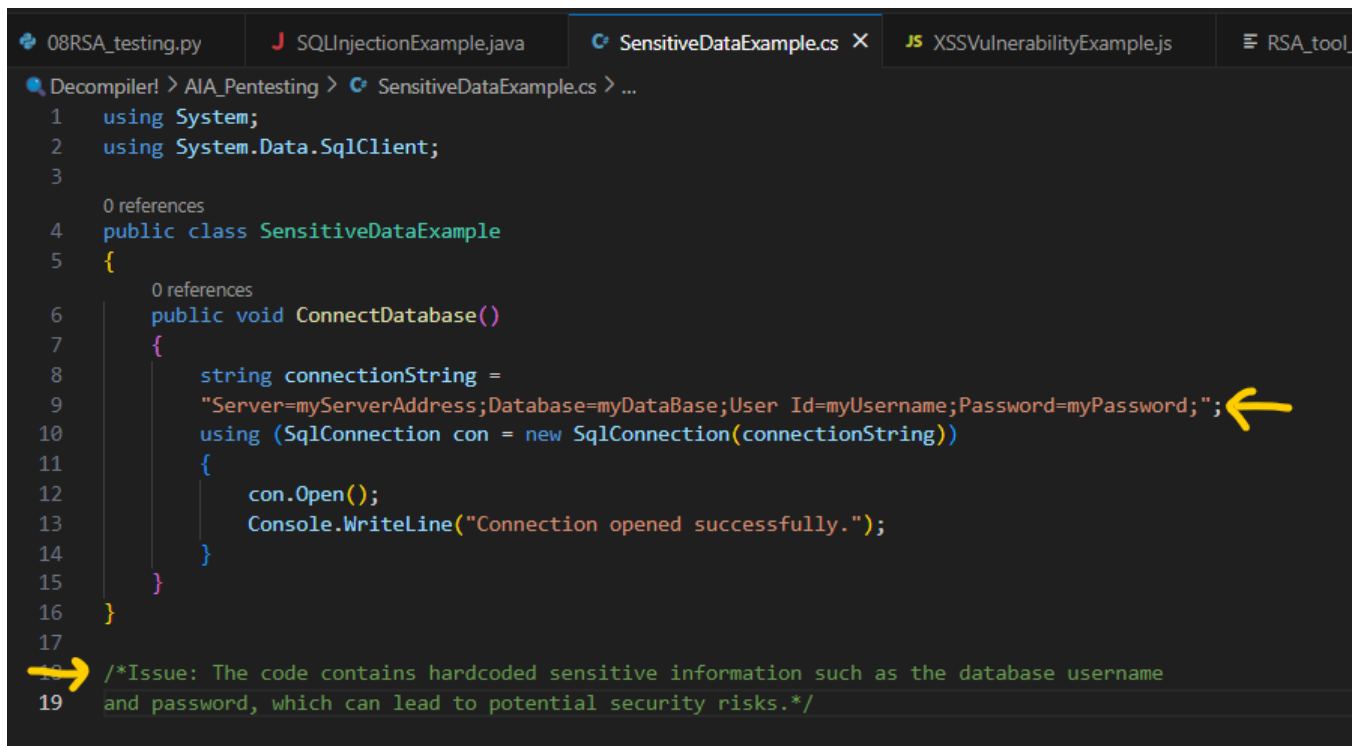
## Software vulnerabilities Detecting tools and techniques

### Code Review



```
08RSA_testing.py | SQLInjectionExample.java | SensitiveDataExample.cs | JS XSSVulnerabilityExample.js | RSA_tool_tree.txt
Decompiler! > AIA_Pentesting > SQLInjectionExample.java
1  import java.sql.Connection;
2  import java.sql.DriverManager;
3  import java.sql.ResultSet;
4  import java.sql.Statement;
5
6  public class SQLInjectionExample {
7      public void getUserData(String username) {
8          try {
9              Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/test", "user", "password");
10             Statement stmt = con.createStatement();
11             String query = "SELECT * FROM users WHERE username = " + username + "'";
12             ResultSet rs = stmt.executeQuery(query);
13             while (rs.next()) {
14                 System.out.println("User ID: " + rs.getInt("id"));
15             }
16         } catch (Exception e) {
17             e.printStackTrace();
18         }
19     }
20 }
21
22 //Issue: The above code is vulnerable to SQL injection as it directly concatenates
23 // user input (username) into the query without sanitization.
```

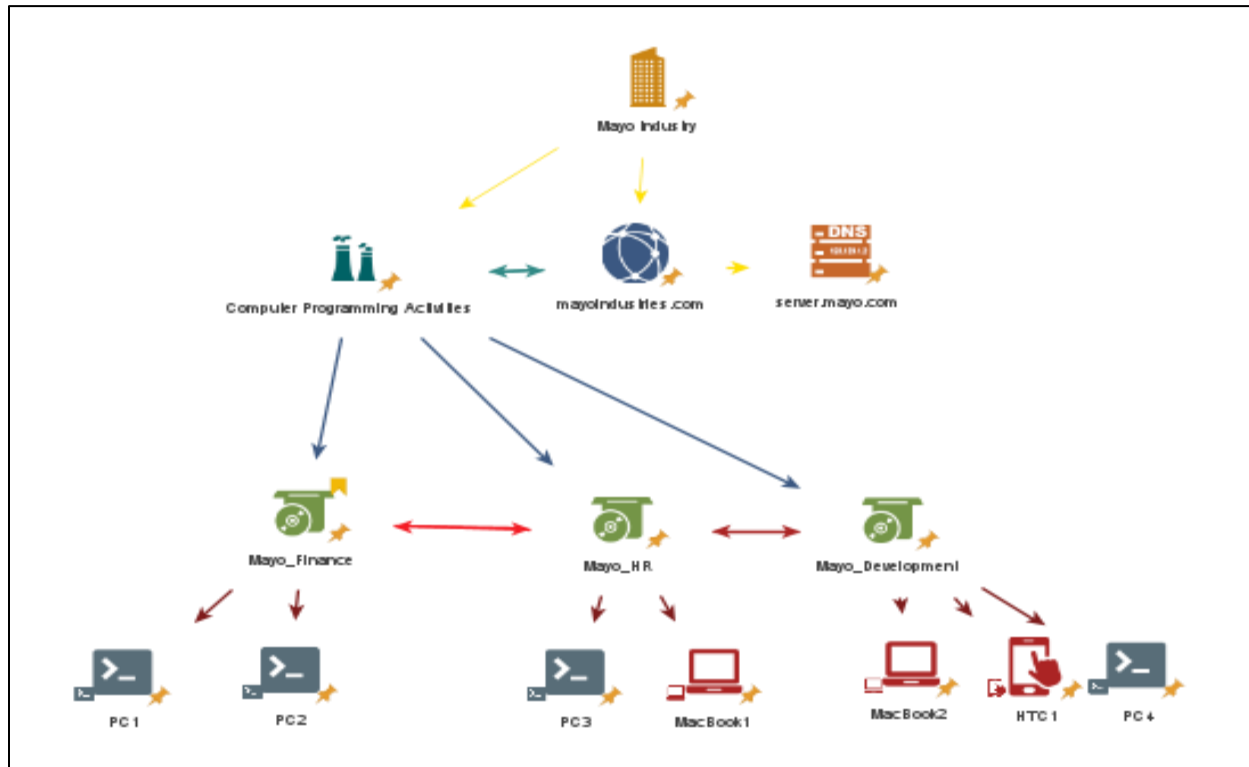
(Image 01)



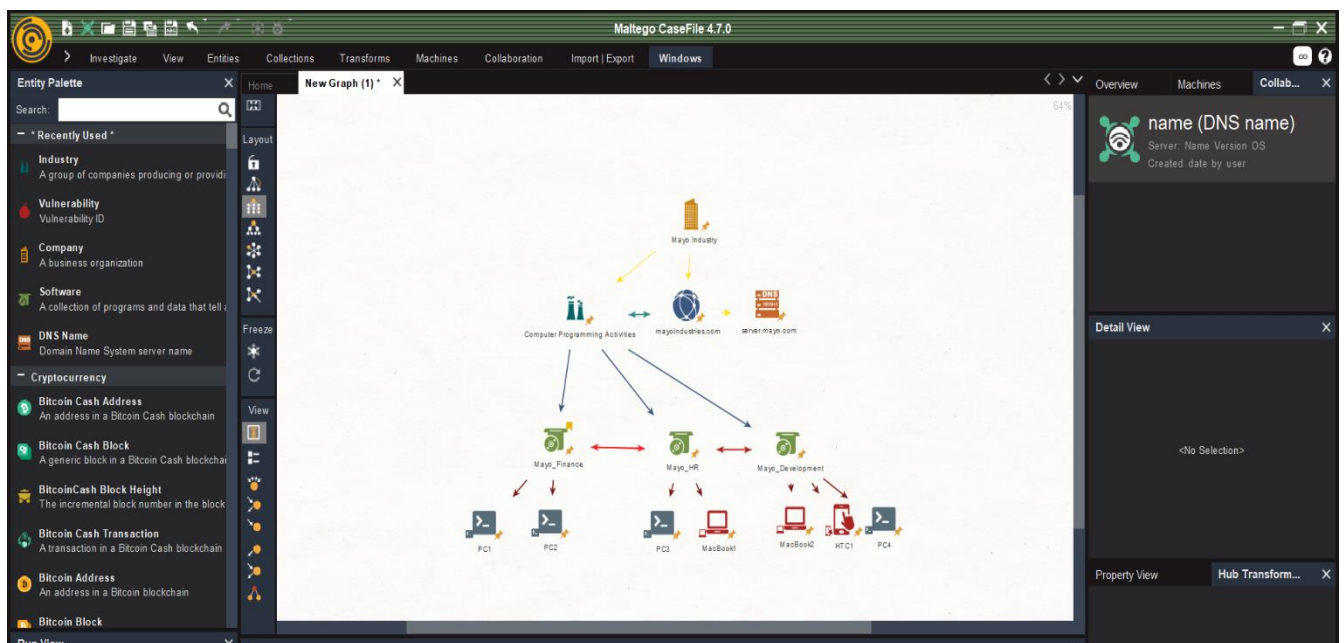
```
08RSA_testing.py | SQLInjectionExample.java | SensitiveDataExample.cs | JS XSSVulnerabilityExample.js | RSA_tool_
Decompiler! > AIA_Pentesting > SensitiveDataExample.cs > ...
1  using System;
2  using System.Data.SqlClient;
3
4  0 references
5  public class SensitiveDataExample
6  {
7      0 references
8      public void ConnectDatabase()
9      {
10         string connectionString =
11         "Server=myServerAddress;Database=myDataBase;User Id=myUsername;Password=myPassword;";
12         using (SqlConnection con = new SqlConnection(connectionString))
13         {
14             con.Open();
15             Console.WriteLine("Connection opened successfully.");
16         }
17     }
18
19     /*Issue: The code contains hardcoded sensitive information such as the database username
    and password, which can lead to potential security risks.*/
```

(Image 02)

## Maltego tool



(Image 03)



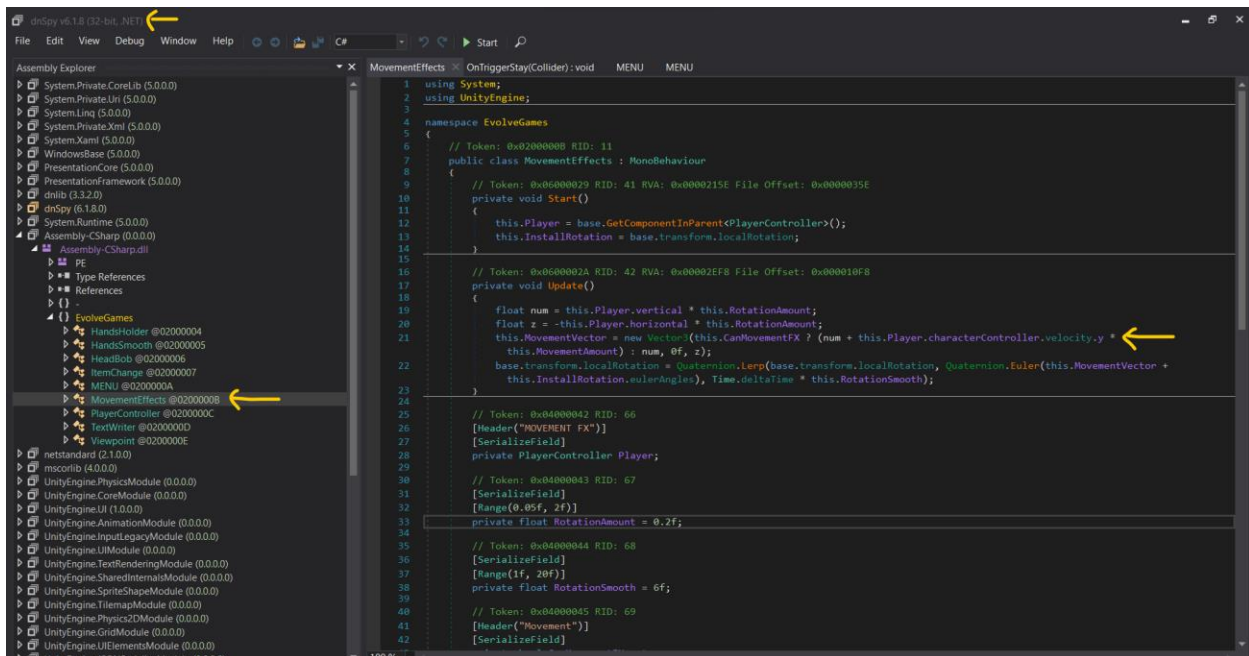
(Image 04)

## Recon-ng reconnaissance

[illegible]

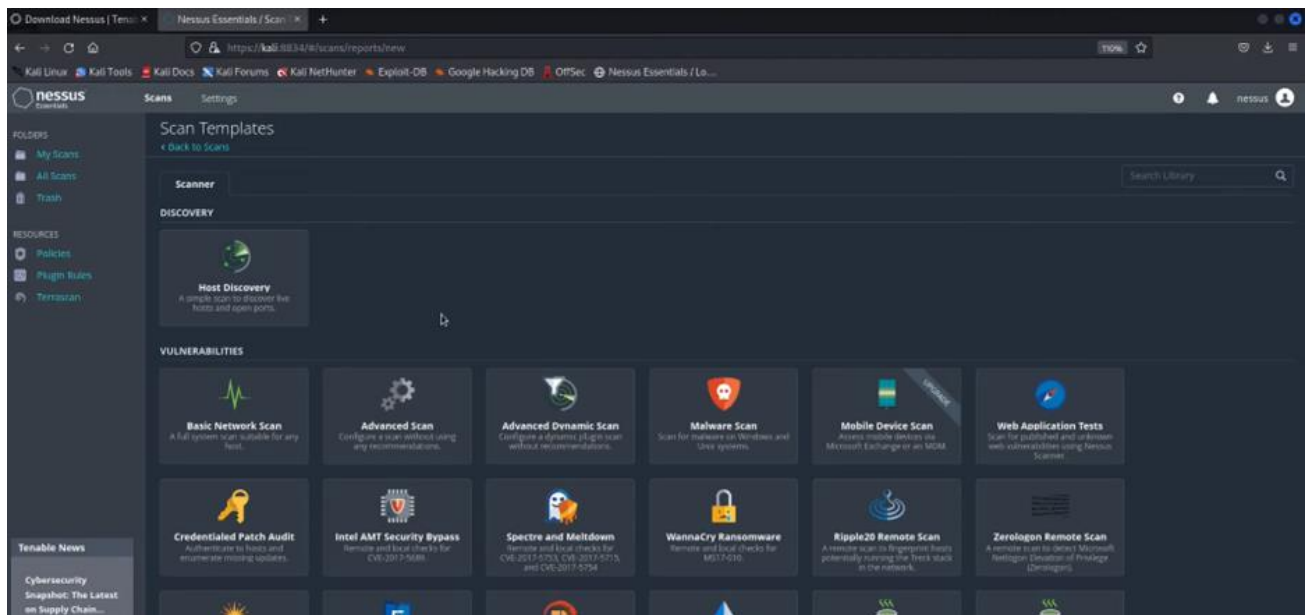
(Image 05)

## Dn-spy software cracker



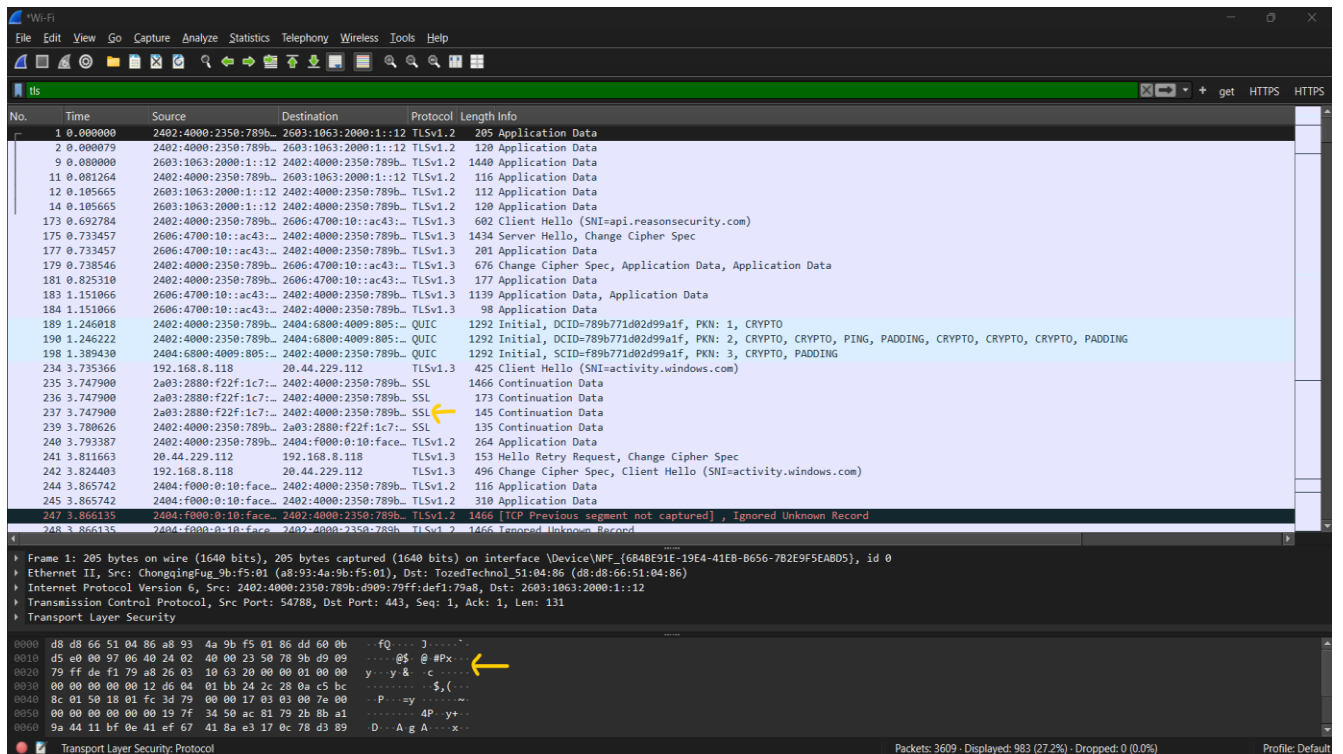
(Image 06)

## Nessus scanner



(Image 06)

## Wireshark



(Image 07)

## John the ripper

```
(kali@kali)-[~]
$ nano test_hashes.txt

(kali@kali)-[~]
$ john test_hashes.txt
```

Warning: detected hash type "LM", but the string is also recognized as "dynamic-md5(\$p)"  
Use the "--format=dynamic-md5(\$p)" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "HAVAL-128-4"  
Use the "--format=HAVAL-128-4" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "MD2"  
Use the "--format=MD2" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "mdc2"  
Use the "--format=mdc2" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "mscash"  
Use the "--format=mscash" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "mscash2"  
Use the "--format=mscash2" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "NT"  
Use the "--format=NT" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "Raw-MD4"  
Use the "--format=Raw-MD4" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "Raw-MD5"  
Use the "--format=Raw-MD5" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "Raw-MD5u"  
Use the "--format=Raw-MD5u" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "Raw-SHA1-AxCrypt"  
Use the "--format=Raw-SHA1-AxCrypt" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "ripemd-128"  
Use the "--format=ripemd-128" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "Snefru-128"  
Use the "--format=Snefru-128" option to force loading these as that type instead  
Warning: detected hash type "LM", but the string is also recognized as "ZipMonster"  
Use the "--format=ZipMonster" option to force loading these as that type instead  
Using default input encoding: UTF-8  
Using default target encoding: CP850  
Loaded 4 password hashes with no different salts (LM [DES 128/128 SSE2])  
Warning: poor OpenMP scalability for this hash type, consider --fork=8  
Will run 8 OpenMP threads  
Proceeding with single, rules:Single  
Press 'q' or Ctrl-C to abort, almost any other key for status  
Almost done: Processing the remaining buffered candidate passwords, if any.  
Warning: Only 756 candidates buffered for the current salt, minimum 1024 needed for performance.  
Proceeding with wordlist:/usr/share/john/password.lst  
Proceeding with incremental:LM\_ASCII

(Image 09)



## Hydra

```
File Actions Edit View Help
root@kali: ~/home/kali
$ hydra -l msfadmin -P passwords.txt 192.168.99.7 ftp
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-05-25 11:47:48
[DATA] max 12 tasks per 1 server, overall 12 tasks, 12 login tries (l:1/p:12), ~1 try per task
[DATA] attacking ftp://192.168.99.7:21/
[21][ftp] host: 192.168.99.7 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-05-25 11:47:52

root@kali: ~/home/kali
$ hydra -l users.txt -p msfadmin 192.168.99.7 ftp
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-05-25 11:48:47
[DATA] max 13 tasks per 1 server, overall 13 tasks, 13 login tries (l:13/p:1), ~1 try per task
[DATA] attacking ftp://192.168.99.7:21/
[21][ftp] host: 192.168.99.7 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-05-25 11:48:51
```

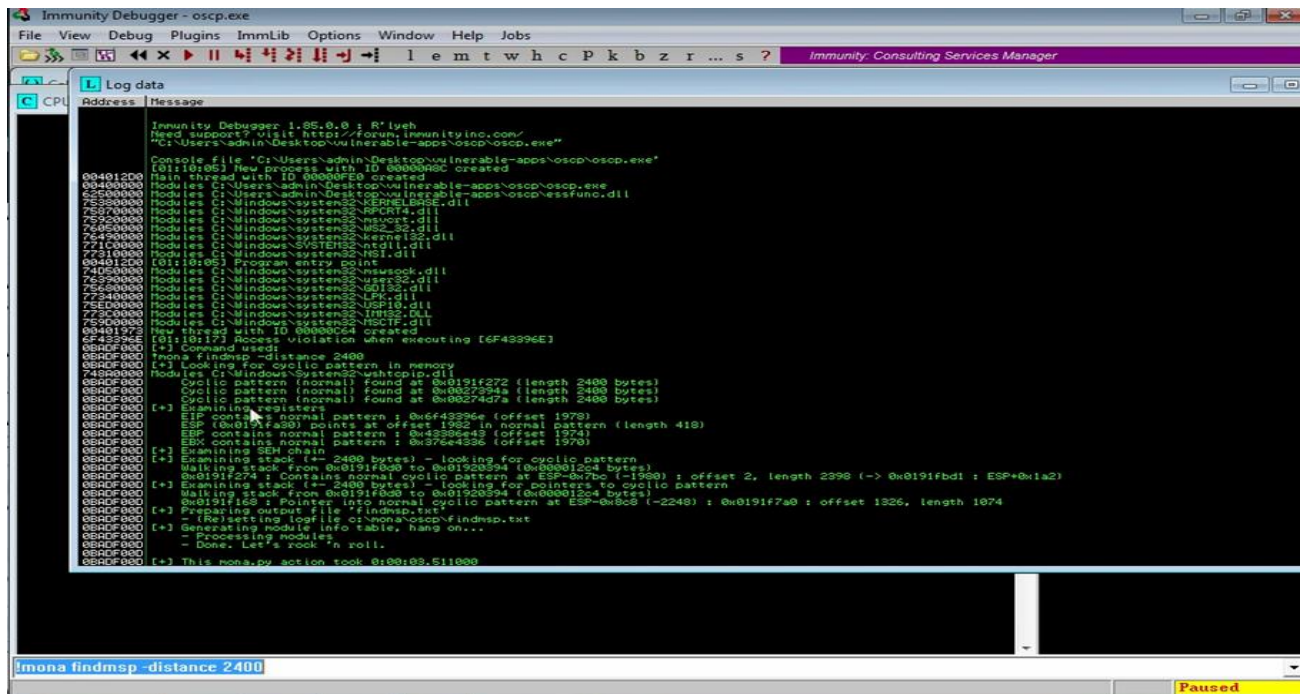
(Image 10)

## Valgrind

```
[X]-[parrot@MSI]-[/mnt/d/YEAR 3 SEM2 01 Cyber Security/01_Applied Information Assurance - IE3022/00 Assignment 1]
$ ls
buffer_overflow.c buffer_overflow.exe
[parrot@MSI]-[/mnt/d/YEAR 3 SEM2 01 Cyber Security/01_Applied Information Assurance - IE3022/00 Assignment 1]
$ valgrind --leak-check=full --track-origins=yes ./buffer_overflow.c
==1148== Memcheck, a memory error detector
==1148== Copyright (C) 2002-2022, and GNU GPL'd, by Julian Seward et al.
==1148== Using Valgrind-3.19.0 and LibVEX; rerun with -h for copyright info
==1148== Command: ./buffer_overflow.c
==1148==
./buffer_overflow.c: 1: //: Permission denied
==1149==
==1149== HEAP SUMMARY:
==1149==    in use at exit: 1,386 bytes in 34 blocks
==1149==   total heap usage: 36 allocs, 2 frees, 4,410 bytes allocated
==1149==
==1149== LEAK SUMMARY:
==1149==    definitely lost: 0 bytes in 0 blocks
==1149==    indirectly lost: 0 bytes in 0 blocks
==1149==    possibly lost: 0 bytes in 0 blocks
==1149==    still reachable: 1,386 bytes in 34 blocks
==1149==    suppressed: 0 bytes in 0 blocks
==1149== Reachable blocks (those to which a pointer was found) are not shown.
==1149== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==1149==
==1149== For lists of detected and suppressed errors, rerun with: -s
==1149== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
./buffer_overflow.c: 4:
./buffer_overflow.c: 5: Syntax error: "(" unexpected
==1148==
==1148== HEAP SUMMARY:
==1148==    in use at exit: 1,370 bytes in 33 blocks
==1148==   total heap usage: 39 allocs, 6 frees, 3,450 bytes allocated
==1148==
```

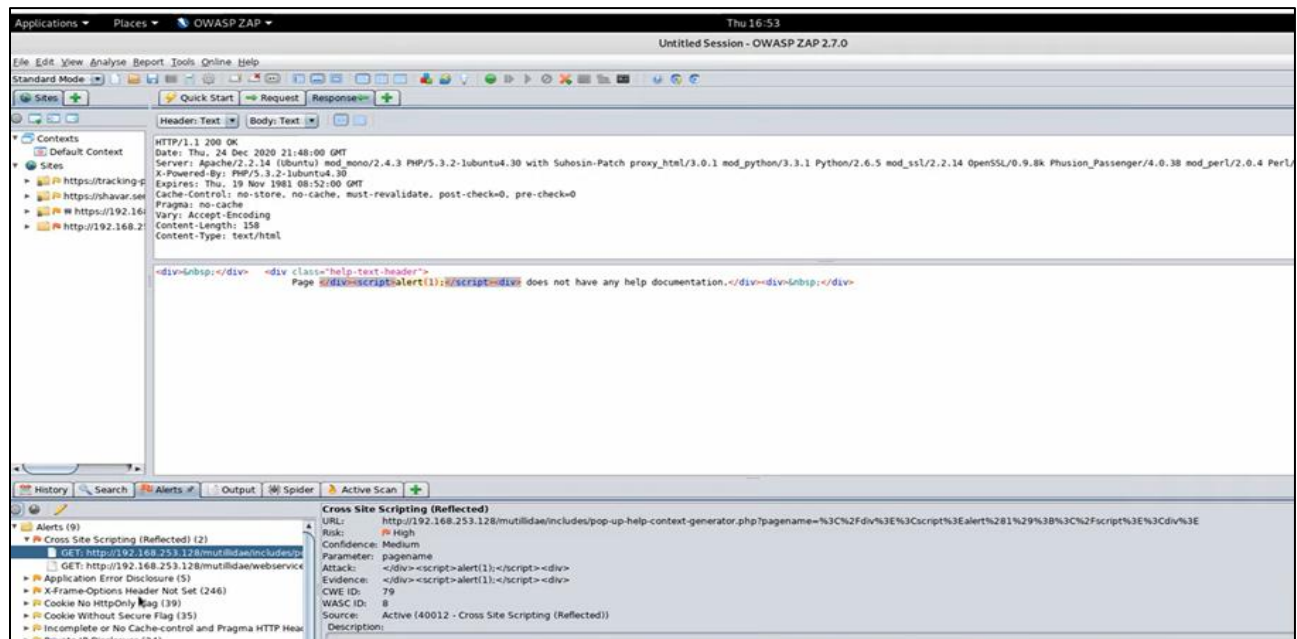
(Image 11)

## Immunity Debugger



(Image 12)

## OWASP ZAP



(Image 13)

## Burp Suite

The screenshot shows the Burp Suite interface with the HTTP history tab selected. A table lists several HTTP requests. The 7th request is highlighted with a yellow arrow pointing to its URL: `https://0aa7006603866cb4833e8c180029007c.web-security-academy.net/files/avatars/PROF%20PIC%20.jpg`. Below the table, the 'Request' and 'Response' sections are visible. The 'Request' section shows the raw HTTP request, and the 'Response' section shows the raw HTTP response. The 'Inspector' panel on the right shows the request attributes, cookies, headers, and response headers.

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies	Time	Listener port	Start response
1	https://portswigger.net	GET	/academy/labs/launch/df0cce82acda8dae507a0f...			302	1767						34.240.117.4	SessionId=CfD1...	22:53:28.6...	8080	12270
2	https://0aa7006603866cb4833e8c180029007c.web-security-academy.net	GET	/my-account			200	4344	HTML		Web shell upload via...			34.246.129.62		22:56:05.6...	8080	198
3	https://0aa7006603866cb4833e8c180029007c.web-security-academy.net	GET	/files/avatars/PROF%20PIC.jpg			200	102563	JPEG	jpg				34.246.129.62		22:56:06.6...	8080	225
4	https://0aa7006603866cb4833e8c180029007c.web-security-academy.net	GET	/academy/labHeader			101	147						34.246.129.62		22:56:06.6...	8080	173
5	https://0aa7006603866cb4833e8c180029007c.web-security-academy.net	POST	/my-account/avatar			200	334	HTML					34.246.129.62		22:56:12.6...	8080	271
6	https://0aa7006603866cb4833e8c180029007c.web-security-academy.net	GET	/my-account			200	4346	HTML		Web shell upload via...			34.246.129.62		22:56:14.6...	8080	177
7	https://0aa7006603866cb4833e8c180029007c.web-security-academy.net	GET	/files/avatars/PROF%20PIC%20.jpg			200	96716	JPEG	jpg				34.246.129.62		22:56:14.6...	8080	187
8	https://0aa7006603866cb4833e8c180029007c.web-security-academy.net	GET	/academy/labHeader			101	147						34.246.129.62		22:56:14.6...	8080	189

(Image 14)

The screenshot shows a web browser window with the address bar displaying `0aa7006603866cb4833e8c180029007c.web-security-academy.net/my-account/avatar`. The page content shows a message: "The file avatars/PROF PIC.jpg has been uploaded." Below the message, there is a link labeled "Back to My Account" with a yellow arrow pointing to it.

(Image 15)

The screenshot shows a web browser window with the address bar displaying `0a8d00cc0447d52d815dd4f3004c00fb.web-security-academy.net/my-account/avatar`. The page content shows an error message: "Sorry, file type application/octet-stream is not allowed Only image/jpeg and image/png are allowed Sorry, there was an error uploading your file." Below the message, there is a link labeled "Back to My Account" with a yellow arrow pointing to it.

(Image 16)



## Metasploit

```
File Actions Edit View Help  
  
kali@kali-[-~]  
msfconsole
```

it looks like you're trying to run a module

@ @  
| |  
|| ||  
\\ \\

```
[ metasploit v6.3.16-dev ]  
+ --[ 2315 exploits - 1208 auxiliary - 412 post ]  
+ --[ 975 payloads - 46 encoders - 11 nops ]  
+ --[ 9 evasion ]
```

Metasploit tip: Metasploit can be configured at startup, see msfconsole --help to learn more  
Metasploit Documentation: https://docs.metasploit.com/  
msf6 >

(Image 17)

## The Harvester

```

root@kali: ~
File Actions Edit View Help
*
* theHarvester 4.0.2
* Coded by Christian Martorella
* Edge-Security Research
* cmartorella@edge-security.com
*
*****

usage: theHarvester [-h] -d DOMAIN [-l LIMIT] [-S START] [-g] [-p] [-s] [--screenshot SCREENSHOT] [-v] [-e DNS_SERVER] [-t DNS_TLD] [-r] [-n] [-c] [-f FILENAME] [-b SOURCE]

theHarvester is used to gather open source intelligence (OSINT) on a company or domain.

optional arguments:
  -h, --help            show this help message and exit
  -d DOMAIN, --domain DOMAIN
                        Company name or domain to search.

  -l LIMIT, --limit LIMIT
                        Limit the number of search results, default=500.

  -S START, --start START
                        Start with result number X, default=0.

  -g, --google-dork      Use Google Dorks for Google search.
  -p, --proxies          Use proxies for requests, enter proxies in proxies.yaml.
  -s, --shodan           Use Shodan to query discovered hosts.
  --screenshot SCREENSHOT
                        Take screenshots of resolved domains specify output directory: --screenshot output_directory

  -v, --virtual-host     Verify host name via DNS resolution and search for virtual hosts.
  -e DNS_SERVER, --dns-server DNS_SERVER
                        DNS server to use for lookup.

  -t DNS_TLD, --dns-tld DNS_TLD
                        Perform a DNS TLD expansion discovery, default False.

  -r, --take-over        Check for takeovers.
  -n, --dns-lookup       Enable DNS server lookup, default False.
  -c, --dns-brute        Perform a DNS brute force on the domain.
  -f FILENAME, --filename FILENAME
                        Save the results to an XML and JSON file.

  -b SOURCE, --source SOURCE
                        anubis, baidu, bing, binaryedge, Bingapi, bufferoverun, cysys, certspotter, crtsh, dnsdumpster, duckduckgo, fullhunt, github-code, google, hackertarget,
                        hunter, intelx, linkedin, linkedin links, n45ht, omnisint, otx, pentesttools, projectdiscovery, qwant, rapiddns, rocketreach, securityTrails, spyse,
                        sublist3r, threatcrowd, threatminer, trello, twitter, uriscan, virustotal, yahoo, zoomeye

```

(Image 18)

## Netcraft

The Netcraft website interface displays information for TryHackMe. At the top, there's a navigation bar with the Netcraft logo, a 'LEARN MORE' button, and a 'REPORT FRAUD' button. Below this is a social media share section with icons for GitHub, Twitter, Facebook, LinkedIn, and YouTube. The main content is divided into two sections: 'Background' and 'Network'.

**Background**

Site title	TryHackMe   Cyber Security Training	Date first seen	September 2018
Site rank	113	Netcraft Risk Rating	0/10
Description	TryHackMe is a free online platform for learning cyber security, using hands-on exercises and labs, all through your browser!		
Primary language	English		

**Network**

Site	https://tryhackme.com	Domain	tryhackme.com
Netblock Owner	Cloudflare, Inc.	Nameserver	kip.ns.cloudflare.com
Hosting company	Cloudflare	Domain registrar	namecheap.com

(Image 19)

## Shodan

The Shodan search results page for the query 'cctv' shows a total of 3,331 results. The top countries are listed as Indonesia (778), Thailand (641), United Kingdom (232), United States (152), and Bangladesh (110). The search results are displayed in a table with columns for IP address, location, and details.

**TOTAL RESULTS**

3,331

**TOP COUNTRIES**

Country	Count
Indonesia	778
Thailand	641
United Kingdom	232
United States	152
Bangladesh	110

**Search Results**

IP Address	Location	Details
220.231.90.234	Viettel Group Viet Nam, Ho Chi Minh City	Microsoft RPC Endpoint Mapper d95afe70-a6d5-4259-822e-2c84da1ddb0d version: v1.0 protocol: [MS-RSP]: Remote Shutdown Protocol provider: wininit.exe ncacn_ip_tcp: 192.168.141.39:49664 ncalrpc: WindowsShutdown ncacn_np: \\CCTV-SERVER1\PIPE\InitShutdown ncalrpc: WtsgKRpc0B6EF0 76f226...
161.0.191.51	NETLAND CHILE S.A. Chile, Santiago	SNMP: Unitime: 321598900

(Image 20)

## 2. Network Vulnerabilities

Vulnerability	Description	severity	Detecting tools and techniques
Poor Firewall Configurations[8]	<p>Poor firewall configurations involve misconfigurations of firewall rules that can result in unauthorized access or data leakage.</p> <p>These errors may lead to openings in the network's defenses.</p>	<p><b>High.</b></p> <p>Misconfigured firewalls can expose sensitive systems and data to external threats.</p>	<ul style="list-style-type: none"> <li>• Wireshark for network analysis. (Image 07)</li> <li>• Nmap for network scanning (Image 21) (Image 22)</li> <li>• Network Topology Mapper for mapping network structures (Image 23)</li> </ul>
Insecure Wireless Networks	<p>Insecure wireless networks result from inadequate security measures in wireless communication protocols.</p> <p>Weak encryption and authentication can lead to unauthorized access to the network.</p>	<p><b>High.</b></p> <p>Insecure wireless networks can allow unauthorized users to intercept and compromise data.</p>	<ul style="list-style-type: none"> <li>• Aircrack-ng (Image 24)</li> <li>• Wireshark for wireless network traffic analysis (Image 07)</li> <li>• Kismet for wireless network detection (Image 25)</li> <li>• Angry IP Scanner for IP address and port scanning (Image 26) (Image 27)</li> <li>• Hydra for password cracking, including SSH (Image 10)</li> </ul>
Insecure Incoming Emails [9]	<p>Insecure email configurations may lead to phishing attacks, spam, and malware infections through those malicious mails.</p> <p>Failure to implement email security measures can result in malicious emails entering the organization's network.</p>	<p><b>Moderate to high.</b></p> <p>Insecure emails can lead to various security incidents, including malware infections and data breaches.</p>	<ul style="list-style-type: none"> <li>• MailScanner</li> <li>• OpenEMM for email marketing security</li> <li>• The Harvester for gathering email addresses and information (Image 18)</li> <li>• GoPhish for phishing simulations (Image 28)</li> <li>• Setoolkit for social engineering attacks (Image 29)</li> </ul>
Social Engineering	<p>Social engineering attacks manipulate individuals to divulge confidential information or perform</p>	<p><b>Moderate to high.</b></p> <p>Social engineering can exploit human</p>	<ul style="list-style-type: none"> <li>• Social-Engineer Toolkit (SET) (Image 29)</li> <li>• Phishing Frenzy</li> <li>• GoPhish (Image 28)</li> </ul>

	<p>actions that compromise security.</p> <p>Techniques may include pretexting, baiting, or phishing.</p>	<p>vulnerabilities, potentially leading to data breaches or system compromises.</p>	<ul style="list-style-type: none"> <li>• Maltego for information gathering and reconnaissance (Image 03)</li> <li>• Recon-ng for data discovery and reconnaissance (Image 05)</li> </ul>
Outdated Or Unpatched Networks	<p>Outdated or unpatched networks result from failing to apply security updates and patches.</p> <p>This can leave systems vulnerable to known exploits.</p>	<p><b>High.</b></p> <p>Attackers frequently target unpatched systems, as known vulnerabilities are readily exploitable.</p>	<ul style="list-style-type: none"> <li>• Nessus <i>Image 03</i></li> <li>• OpenVAS (Open Vulnerability Assessment System) (Image 30)</li> <li>• Nexpose</li> <li>• Nmap for network scanning (Image 21) (Image 22)</li> <li>• Network Topology Mapper for mapping network structures (Image 23)</li> </ul>

## Nmap

```
(kali@kali)-[~]
$ nmap --version
Nmap version 7.94SVN ( https://nmap.org )
Platform: x86_64-pc-linux-gnu
Compiled with: liblua-5.4.6 openssl-3.2.2-dev libssh2-1.11.0 libz-1.3 libpcres2-10.42 libpcap-1.10.4 nmap-libdnet-1.12 ipv6
Compiled without:
Available nsock engines: epoll poll select
```

(Image 21)

```
(root@kali)-[/home/kali]
# nmap -sS -p 53 192.168.8.1
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-30 04:22 EDT
Nmap scan report for 192.168.8.1
Host is up (0.00080s latency).

PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds

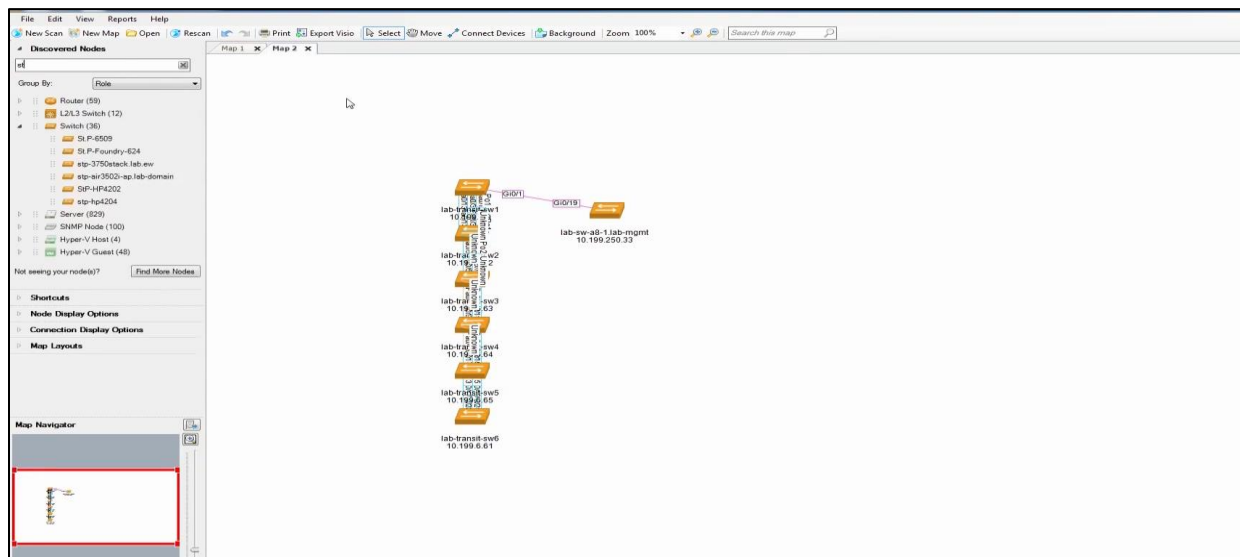
(root@kali)-[/home/kali]
# nmap -sS 192.168.8.1 -p 53
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-30 04:23 EDT
Nmap scan report for 192.168.8.1
Host is up (0.0010s latency).

PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.22 seconds
```

(Image 22)

## Network Topology Mapper



(Image 23)

## Aircrack-ng

```
(root@kali)-[/home/kali]
# aircrack-ng start wlan0

PHY      Interface      Driver      Chipset
phy0     wlan0           88XXau      Realtek Semiconductor Corp. RTL8812AU 802.11a/b/g/n/ac 2T2R DB WLAN Adapter
          (mac80211 monitor mode already enabled for [phy0]wlan0 on [phy0]10)

(root@kali)-[/home/kali]
```

(Image 24)

## Kismet

```
root@localhost kismet]# pacman -S kismet
warning: kismet-2021_08_R1-2 is up to date -- reinstalling
resolving dependencies...
looking for conflicting packages...

packages (1) kismet-2021_08_R1-2
Total Installed Size: 24.25 MiB
Net Upgrade Size: 0.00 MiB

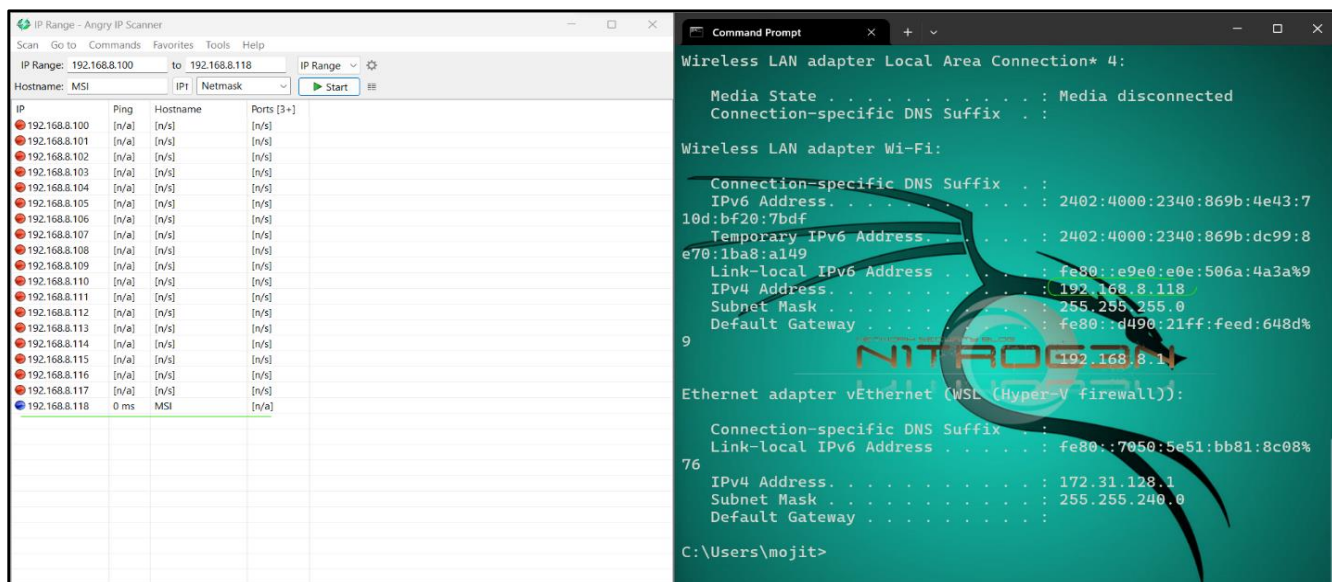
Proceed with installation? [Y/n] ^C
interrupt signal received

root@localhost kismet]# cd /etc/kismet
root@localhost kismet]# ls\
^C
root@localhost kismet]# ls
kismet-20211122-04-11-43-1.kismet  kismet_alerts.conf  kismet_filter.conf  kismet_logging.conf  kismet_uav.conf
kismet_80211.conf                  kismet.conf         kismet_httpd.conf  kismet_memory.conf
root@localhost kismet]#
```

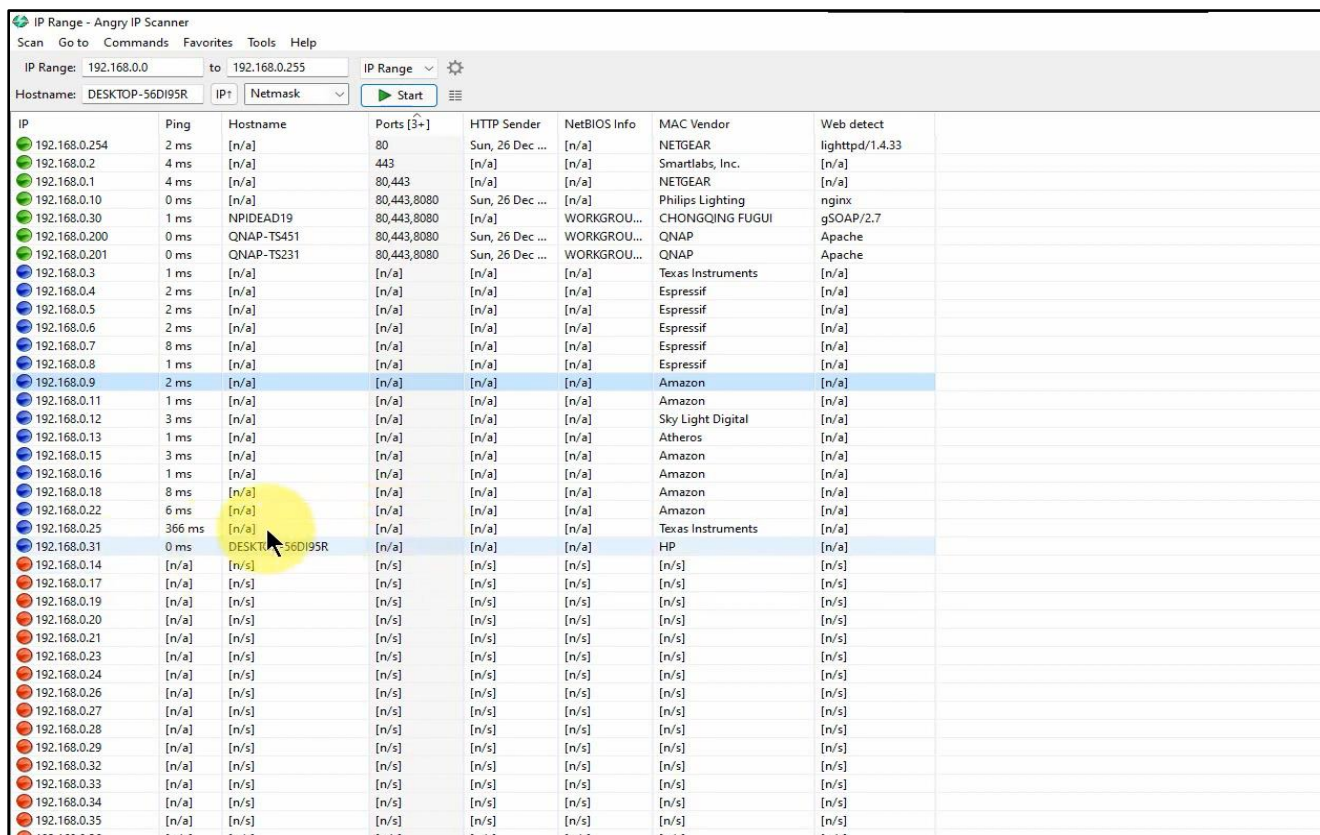
(Image 25)



## Angry IP



(Image 26)



(Image 27)

## Go phish

The screenshot shows the Gophish web interface. The sidebar on the left contains navigation links: Dashboard, Campaigns, Users & Groups, Email Templates, Landing Pages, Sending Profiles, Account Settings, User Management, Webhooks, User Guide, and API Documentation. The main content area is titled "Sending Profiles" and features a "New Profile" button and a message: "No profiles created yet. Let's create one!". A modal form is open for creating a new profile. The form includes fields for Name, Interface Type, SMTP From, Host, Username, Password, and a checkbox for "Ignore Certificate Errors". There is also a section for "Email Headers" with a table and a "Send Test Email" button.

(Image 28)

## SET

```
XX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
.o88o.      o8o      .
888  "      "      "
o888oo      .oooo.o  .ooooo.  .ooooo.  oooo  .ooooo.  .o888oo  oooo  ooo
888  d88(  "8  d88'  `88b  d88'  `Y8  `888  d88'  `88b  888  `88.  .8'
888  `""Y88b.  888  888  888  888  888  888oo888  888  `88..8'
888  o.  )88b  888  888  888  .o8  888  888  .o  888  .  `888'
o888o  8""888P' `Y8bod8P' `Y8bod8P' o888o `Y8bod8P' "888"  d8'
                                     .o...P'
                                     `XERO'

[---]      The Social-Engineer Toolkit (SET)      [---]
[---]      Created by: David Kennedy (ReL1K)      [---]
[---]      Version: 8.0.3
[---]      Codename: 'Maverick'
[---]      Follow us on Twitter: @TrustedSec      [---]
[---]      Follow me on Twitter: @HackingDave    [---]
[---]      Homepage: https://www.trustedsec.com   [---]
Welcome to the Social-Engineer Toolkit (SET).
The one stop shop for all of your SE needs.

The Social-Engineer Toolkit is a product of TrustedSec.

Visit: https://www.trustedsec.com

It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!

Select from the menu:

1) Social-Engineering Attacks
2) Penetration Testing (Fast-Track)
3) Third Party Modules
4) Update the Social-Engineer Toolkit
5) Update SET configuration
6) Help, Credits, and About

99) Exit the Social-Engineer Toolkit
```

(Image 29)

## OpenVAS

The screenshot shows the OpenVAS web interface in a browser window. The address bar displays `https://127.0.0.1:9392/targets`. The page title is "Greenbone Security Assistant". The main content area is titled "New Target" and contains the following fields and options:

- Name:** A text input field containing "Unnamed".
- Comment:** An empty text input field.
- Hosts:** A section with two radio buttons: "Manual" (selected) and "From file" (with a "Browse..." button and the text "No file selected.").
- Exclude Hosts:** A section with two radio buttons: "Manual" (selected) and "From file" (with a "Browse..." button and the text "No file selected.").
- Port List:** A dropdown menu showing "All IANA assigned TCP" with a plus icon.
- Alive Test:** A dropdown menu showing "Scan Config Default".
- Credentials for authenticated checks:** A section with three rows, each with a dropdown menu and a plus icon:
  - SSH: dropdown shows "--", followed by "on port" and a text input field containing "22".
  - SMB: dropdown shows "--".
  - ESXi: dropdown shows "--".

At the bottom of the form are two green buttons: "Cancel" on the left and "Save" on the right.

(Image 30)



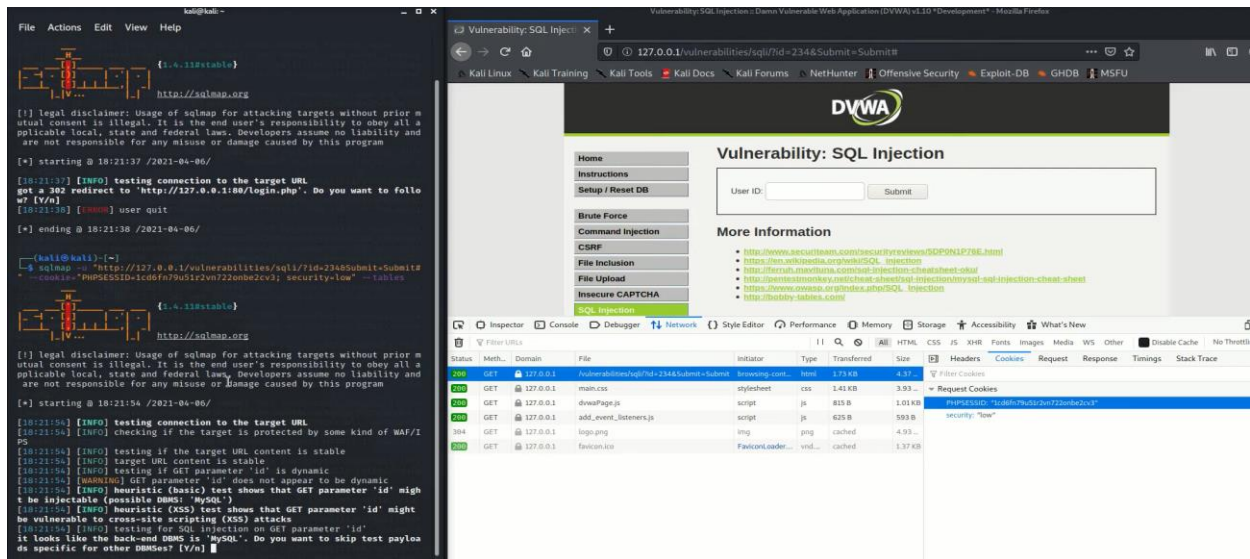
### 3. Web application vulnerabilities

Vulnerability	Description	severity	Detecting tools and techniques
XQL Injections	XQL injections involve manipulating XML query languages to extract unauthorized information or execute malicious code on web applications.[10]	<b>High.</b>  Exploiting XQL injections can lead to data theft or unauthorized system access.	<ul style="list-style-type: none"> <li>• SQL Map (Image 30)</li> <li>• OWASP ZAP(Image 13)</li> <li>• Burp Suite (Image 30) (Image 31) (Image 32)</li> <li>• Maltego for information gathering and reconnaissance.</li> <li>• Recon-ng for data discovery and reconnaissance.</li> </ul>
Cross-Site Scripting (XSS)	<p>XSS vulnerabilities allow attackers to inject malicious scripts into web pages viewed by other users.</p> <p>This can lead to session hijacking, data theft, and more.</p>	<b>High.</b>  XSS can compromise user data and privacy.	<ul style="list-style-type: none"> <li>• OWASP ZAP(Image 13)</li> <li>• Burp Suite professional</li> <li>• Acunetix</li> <li>• Maltego for information gathering and reconnaissance.</li> <li>• Recon-ng for data discovery and reconnaissance.</li> </ul>
Cross-Site Request Forgery (CSRF)	CSRF attacks trick users into executing unwanted actions on a web application, often without their knowledge or consent.[11]	<b>Moderate to high.</b>  CSRF attacks can lead to unauthorized actions and data modifications.	<ul style="list-style-type: none"> <li>• OWASP ZAP(Image 13)</li> <li>• Burp Suite</li> <li>• CSRFTester</li> <li>• Maltego for information gathering and reconnaissance.</li> <li>• Recon-ng for data discovery and reconnaissance.</li> </ul>
Security Misconfiguration	Security misconfiguration occurs when security settings are improperly configured, leaving vulnerabilities in the application's defenses.	<b>Moderate to high.</b>  Depending on the specific misconfiguration.  It can lead to unauthorized access or data exposure.	<ul style="list-style-type: none"> <li>• Nessus</li> <li>• OpenVAS</li> <li>• Qualys</li> <li>• Maltego for information gathering and reconnaissance.</li> <li>• Recon-ng for data discovery and reconnaissance.</li> </ul>
Broken Authentication.	Broken authentication vulnerabilities can occur	<b>High.</b>	<ul style="list-style-type: none"> <li>• OWASP</li> <li>• Amass (Image 36).</li> </ul>

	when user authentication and session management are flawed, allowing unauthorized access to accounts.	Broken authentication can result in unauthorized access to sensitive user data.	<ul style="list-style-type: none"> <li>• Nessus</li> <li>• OpenVAS</li> <li>• Maltego for information gathering and reconnaissance.</li> <li>• Recon-ng for data discovery and reconnaissance.</li> </ul>
Directory Traversal	Directory traversal attacks exploit vulnerabilities in web applications, enabling attackers to access files and directories outside the intended scope.	<p><b>Moderate to high.</b></p> <p>Directory traversal can lead to unauthorized access.</p>	<ul style="list-style-type: none"> <li>• DirBuster</li> <li>• OWASP ZAP(Image 13)</li> <li>• Burp Suite (Image 34) (Image 35) (Image 36)</li> <li>• Maltego for information gathering and reconnaissance.</li> <li>• Recon-ng for data discovery and reconnaissance.</li> </ul>

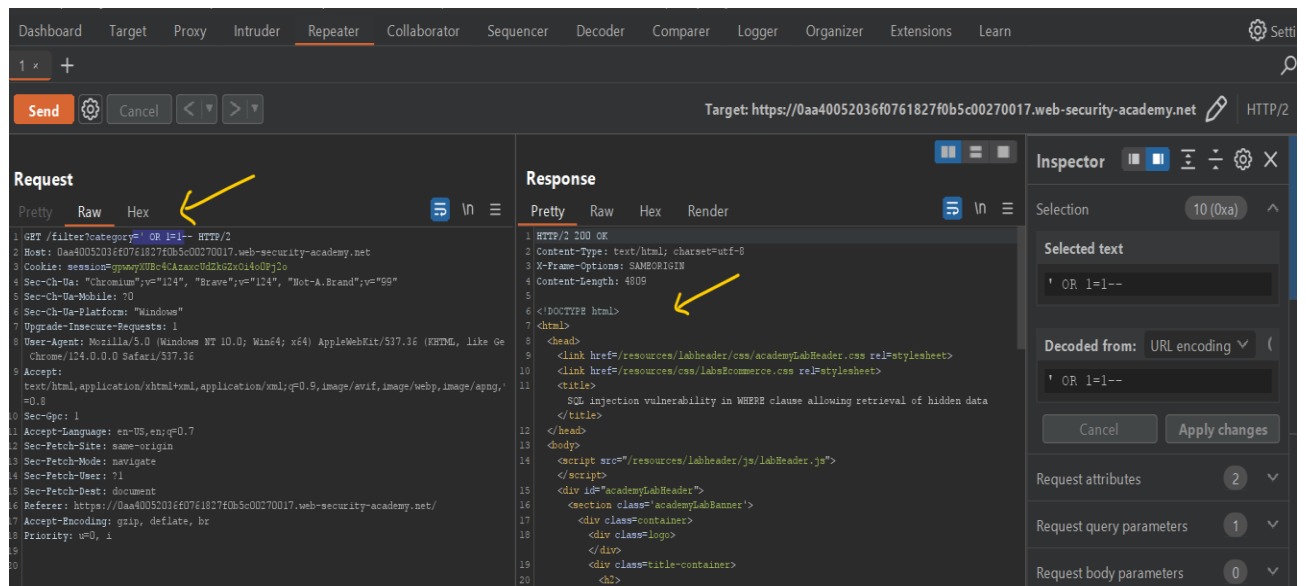
## Web application Vulnerabilities Detecting tools and techniques.

### SQL map

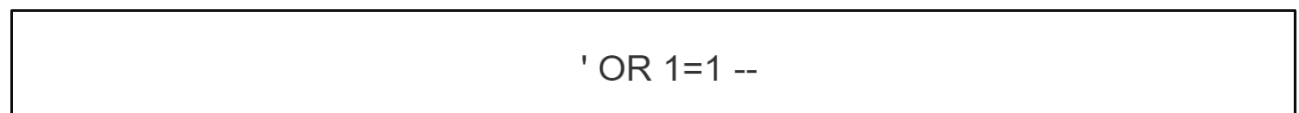


(Image 30)

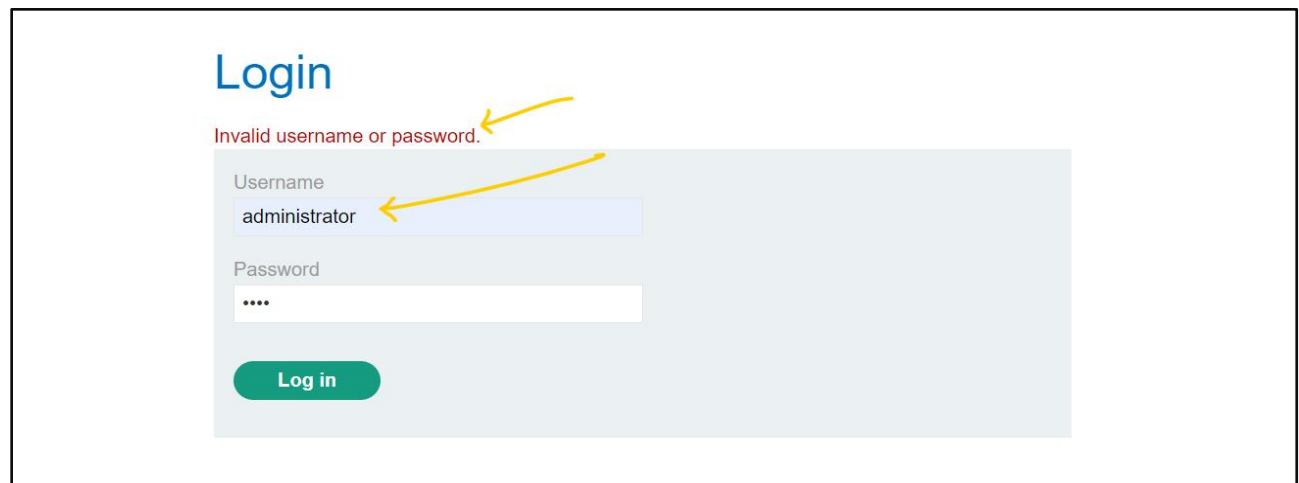
## Burp Suite for SQL injections



(Image 31)



(Image 32)



(Image 33)

[illegible]

The screenshot shows a web browser's developer tools. The 'Request' tab is selected, displaying the raw HTTP request. A red arrow points to the 'Cookie' field in the request headers, which contains a session cookie. The 'Response' tab is also visible, showing the raw HTTP response. The 'Inspector' panel on the right shows the request and response details.

Dashboard
Target
Proxy
Intruder
Repeater
Collaborator
Sequencer
Decoder
Compiler
Logger
Organizer
Extensions
Learn

1 ×
2 ×
3 ×
4 ×
+

Send
Cancel
<
>

Target: https://0a7c000d04bdd3685ea03c7002f000a.web-security-academy.net/


Request

Pretty
Raw
Hex

1 GET /image?filename=39.jpg HTTP/2  
2 Host: 0a7c000d04bdd3685ea03c7002f000a.web-security-academy.net  
3 Cookie: session=4f5aBVFHUF08K0LGHryu0hsh0L37x1e  
4 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"  
5 Sec-Ch-Ua-Mobile: ?0  
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)  
7 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.160  
8 Safari/537.36  
9 Sec-Ch-Ua-Platform: "Windows"  
10 Accept: image/avif,image/webp,image/apng,image/svg+xml,image/\*,\*/\*;q=0.8  
11 Sec-Fetch-Site: same-origin  
12 Sec-Fetch-Mode: no-cors  
13 Sec-Fetch-Dest: image  
14 Referer: https://0a7c000d04bdd3685ea03c7002f000a.web-security-academy.net/product  
15 Accept-Encoding: gzip, deflate, br  
16 Accept-Language: en-US,en;q=0.9  
17 Proxy: uel, 1

Response

Pretty
Raw
Hex
Render



Inspector

Request attr
Request que
Request bod
Request coo
Request hea
Response he

0 highlights

Done

## Amass

[illegible]

(Image 36)

## Blue Team Analysis

The Blue Team plays a pivotal role in ensuring the cybersecurity resilience of an organization. Its primary responsibility is to analyze, evaluate, and address vulnerabilities identified by the Red Team comprehensively and proactively. In the context of our engagement with Mayo Industries, the Blue Team's task is to strengthen the organization's defenses against a range of vulnerabilities encompassing software, network, and web applications. By strategically implementing countermeasures and best practices, the Blue Team aims to fortify the company's security posture, reducing the risk of exploitation and potential security incidents. In the following sections, we delve into the specific vulnerabilities discovered and the corresponding countermeasures employed to safeguard Mayo Industries from cyber threats.[12]

### Countermeasures against Software vulnerabilities

Software vulnerabilities pose a formidable challenge to organizations, potentially leading to data breaches and system compromises. The Blue Team takes a proactive stance in mitigating these risks by addressing software vulnerabilities head-on. To counteract insecure design, the Blue Team advocates a shift towards secure software development practices, emphasizing threat modeling, secure coding guidelines, and comprehensive code reviews. For cryptographic failures, they focus on adopting secure cryptographic libraries, implementing rigorous key management, and conducting regular cryptographic audits. In cases of buffer overflow, the Blue Team insists on secure coding practices, input validation, and code audits. They recommend scrutinizing file uploads for unrestricted and dangerous types and advocate robust, real-time scanning. To tackle vulnerable and outdated components, a stringent component management system is put in place. Vulnerability scans using tools like OWASP Dependency-Check, Retire.js, and Sync ensure that the software components remain up-to-date and free from known vulnerabilities. By adopting these measures, the Blue Team fortifies the organization's software infrastructure, ensuring a resilient defense against exploitation.

Vulnerability	Why It's a Risk	Countermeasures
Insecure Design	<ul style="list-style-type: none"><li>Insecure design can lead to persistent vulnerabilities that are difficult and costly to remediate.</li><li>Attackers can leverage these design flaws to gain unauthorized access to sensitive data or systems.</li></ul>	Implement secure software development practices such as threat modeling, secure coding guidelines, and code review to ensure designs are secure from the outset.
Cryptographic Failures	<ul style="list-style-type: none"><li>Cryptographic failures can result in data breaches and loss of confidentiality.</li></ul>	Use well-established cryptographic libraries, ensure proper key management, and perform regular cryptographic audits.



	<ul style="list-style-type: none"> <li>Attackers may decrypt encrypted data, exposing sensitive information.</li> </ul>	
Buffer Overflow	<ul style="list-style-type: none"> <li>Buffer overflows can lead to unauthorized code execution, which can compromise system integrity and provide attackers with control over a system.</li> </ul>	Implement code and input validation, utilize safe programming languages, and conduct regular code audits to identify and fix buffer overflow issues.
Unrestricted Upload of Dangerous File Types	<ul style="list-style-type: none"> <li>Unrestricted file uploads can result in malware infections, data breaches, and other security incidents.</li> </ul>	Implement content validation and filtering to restrict the types of files that can be uploaded and ensure they are thoroughly scanned for threats.
Vulnerable and Outdated Components	<ul style="list-style-type: none"> <li>Attackers often target known vulnerabilities in components to exploit systems.</li> <li>Failing to update or replace these components can lead to security breaches.</li> </ul>	Maintain a comprehensive inventory of software components, keep them up to date, and regularly scan for vulnerabilities using tools like Retire.js and Snyk.

## Countermeasures against Network Vulnerabilities

Network security stands as the bedrock of safeguarding an organization's digital assets. To address network vulnerabilities, the Blue Team formulates a comprehensive strategy aimed at bolstering Mayo Industries' network defenses. For poor firewall configurations, they meticulously review and update firewall rules, ensuring the principle of least privilege is adhered to. To secure wireless networks against potential threats, the Blue Team recommends strong encryption methods like WPA3, credential changes from default settings, and constant network assessment and monitoring. In response to insecure incoming emails, robust email filtering systems are implemented, shielding against phishing attempts. Furthermore, a comprehensive employee training program is adopted to educate and sensitize the workforce to email security. To combat the evolving tactics of social engineering, the Blue Team promotes security awareness training and reporting procedures for suspicious activities. In addressing the issue of outdated or unpatched networks, a stringent patch management system is adopted, coupled with network vulnerability scans to ensure timely updates and patch applications. With these measures in place, the Blue Team actively fortifies Mayo Industries' network infrastructure, ensuring its resilience against potential threats.

<b>Vulnerability</b>	<b>Why It's a Risk</b>	<b>Countermeasures</b>
Poor Firewall Configurations	<ul style="list-style-type: none"> <li>Misconfigured firewalls expose networks to threats, allowing malicious traffic to pass through undetected.</li> </ul>	Regularly review and update firewall rules, conduct access control lists (ACLs) audits, and implement the principle of least privilege.
Insecure Wireless Networks	<ul style="list-style-type: none"> <li>Insecure wireless networks can compromise data privacy and expose sensitive information to attackers.</li> </ul>	Use strong encryption (WPA3, for example), change default credentials, and regularly assess and monitor wireless network security.
Insecure Incoming Emails	<ul style="list-style-type: none"> <li>Insecure emails can lead to various security issues, including malware infections and data breaches, as malicious content can easily infiltrate the network.</li> </ul>	Implement robust email filtering systems and conduct employee training to recognize and report phishing attempts.
Social Engineering	<ul style="list-style-type: none"> <li>Social engineering exploits human vulnerabilities, potentially leading to unauthorized access or data exposure.</li> </ul>	Educate employees on social engineering tactics, employ user awareness training, and establish procedures for reporting suspicious activities.
Outdated Or Unpatched Networks	<ul style="list-style-type: none"> <li>Failing to apply security updates and patches allows attackers to exploit known vulnerabilities, resulting in security breaches and data loss.</li> </ul>	Regularly apply security patches and updates, implement patch management systems, and conduct network vulnerability scans.

## Countermeasures against Web Application Vulnerabilities

Web applications, serving as the frontline of digital engagement, are often targeted by adversaries. In response to web application vulnerabilities, the Blue Team implements a multifaceted strategy to enhance the security posture of Mayo Industries. For XQL injections, they advocate input validation, the use of prepared statements in database queries, and comprehensive security testing to identify and remediate injection vulnerabilities. In the case of cross-site scripting (XSS), the Blue Team emphasizes input validation and content sanitization, while also implementing security mechanisms like Content Security Policy (CSP). To combat cross-site request forgery (CSRF), the Blue Team recommends the deployment of anti-CSRF tokens in web forms, stringent request validation, and the implementation of secure session management practices. Addressing security misconfiguration, they focus on the development of secure configuration baselines, regular security audits, and automated configuration checks to identify and rectify misconfigurations. Broken authentication concerns are met with robust password policies, the adoption of multi-factor authentication (MFA), and continuous monitoring



and analysis of authentication logs. For dealing with directory traversal, the Blue Team advocates input validation, file system access restriction, and the use of security mechanisms like Web Application Firewalls (WAFs) to thwart traversal attempts. With these multi-layered measures in place, the Blue Team actively reinforces the integrity and security of Mayo Industries' web applications, significantly reducing the potential for exploitation and data breaches.[13]

<b>Vulnerability</b>	<b>Why It's a Risk</b>	<b>Countermeasures</b>
XQL Injections	<ul style="list-style-type: none"> <li>XQL injections can lead to data theft and unauthorized access, as attackers can view sensitive data and potentially compromise the application's integrity.</li> </ul>	Input validation, use prepared statements in database queries, and conduct security testing to identify and fix injection vulnerabilities.
Cross-Site Scripting (XSS)	<ul style="list-style-type: none"> <li>XSS vulnerabilities can compromise user data and privacy, as well as lead to the hijacking of user sessions.</li> </ul>	Employ input validation, sanitize user-generated content, and utilize security mechanisms like Content Security Policy (CSP).
Cross-Site Request Forgery (CSRF)	<ul style="list-style-type: none"> <li>CSRF attacks can lead to unauthorized actions on behalf of authenticated users, potentially modifying data or making unwanted transactions.</li> </ul>	Implement anti-CSRF tokens in web forms, validate and verify requests on the server side, and employ proper session management.
Security Misconfiguration	<ul style="list-style-type: none"> <li>Misconfigurations can result in unauthorized access and data exposure, as attackers can exploit these vulnerabilities.</li> </ul>	Develop secure configuration baselines, automate configuration checks, and perform regular security audits to identify and correct misconfigurations.
Broken Authentication	<ul style="list-style-type: none"> <li>Broken authentication can lead to unauthorized access to sensitive user data, which can have serious consequences for an organization.</li> </ul>	Implement strong password policies, enable multi-factor authentication (MFA), and continuously monitor and analyze authentication logs.
Directory Traversal	<ul style="list-style-type: none"> <li>Directory traversal can lead to unauthorized access to files and directories, potentially exposing sensitive data and compromising the web application's security.</li> </ul>	Employ input validation, restrict file system access, and use security mechanisms like Web Application Firewalls (WAFs) to block traversal attempts.

## Purple Team Assessment

The Purple Team assessment marks a critical juncture in the comprehensive cybersecurity evaluation of Mayo Industries. This phase serves as the pivot between the Red Team's offensive penetration attempts and the Blue Team's defensive measures. In particular, the purple Team assessment takes a laser-focused approach to dissect the effectiveness of the defensive tactics and controls proposed by the Blue Team, and how they respond to the vulnerabilities assumed to be exploited by the Red Team.[14]

### Effectiveness of Defensive Tactics and Controls

One of the primary objectives of the Purple Team assessment is to rigorously examine the effectiveness of the defensive tactics and controls put forth by the Blue Team. These measures, meticulously designed to safeguard the organization's digital assets, must undergo a thorough evaluation. We scrutinize the alignment of these controls with industry best practices and the extent to which they offer protection against specific vulnerabilities.

The assessment goes beyond mere compliance checks; it delves into the pragmatic implementation of these controls. It considers their adaptability to the dynamic threat landscape and evaluates their capacity to withstand evolving attack vectors. In this regard, the Purple Team conducts simulated tests and evaluates the real-world effectiveness of the proposed controls. It addresses critical questions such as how quickly the controls detect and respond to potential threats, the precision of intrusion detection systems, and their overall impact on fortifying the defensive posture.

### Protection Against Exploited Vulnerabilities

A central facet of the Purple Team assessment revolves around the evaluation of how well these defensive measures protect against the vulnerabilities assumed to have been exploited by the Red Team during the penetration test. This segment of the assessment delves into the specific vulnerabilities, particularly those rooted in software, network, and web applications, which were identified and subsequently addressed by the Blue Team.

Controlled testing scenarios, replicating the tactics employed by the Red Team, aim to gauge the resilience of the newly implemented measures. This evaluation factors in the time taken to detect and respond to attacks, the accuracy of intrusion detection and prevention systems, and the overall effectiveness of the defensive posture in the context of mitigating vulnerabilities.

The Purple Team assessment, thus, serves as a crucial milestone in enhancing Mayo Industries' cybersecurity preparedness. Identifying the strengths and pinpointing potential areas of improvement in the defensive strategies, facilitates data-driven decision-making, enabling the organization to adapt and bolster its security posture against real-world cyber threats.

# Business Impact Assessment

The Business Impact Assessment presents a comprehensive evaluation of the potential repercussions of each identified vulnerability and weakness across software, network, and web application domains. The assessment takes into account various factors, including financial impact, reputation damage, and operational disruptions, to provide a nuanced understanding of the potential consequences for Mayo Industries.

## Impact of Software Vulnerabilities

- ❖ **Insecure Design** - An insecure software design may lead to severe financial repercussions. Exploitation of this vulnerability could result in data breaches, legal ramifications, and financial losses, stemming from the theft of sensitive data or intellectual property. Additionally, the organization's reputation could be tarnished, leading to a loss of trust among clients and stakeholders.
- ❖ **Cryptographic Failures** - Cryptographic vulnerabilities can jeopardize sensitive data, potentially resulting in regulatory fines due to non-compliance, loss of intellectual property, and the erosion of customer trust. The financial impact, combined with potential legal consequences, can be substantial.
- ❖ **Buffer Overflow** - Buffer overflow vulnerabilities can be exploited to execute malicious code, leading to system crashes and operational disruptions. The financial costs include downtime for system recovery and damage to the organization's reputation for system reliability.
- ❖ **Unrestricted Upload of Dangerous File Types** - Allowing the unrestricted upload of dangerous file types poses a substantial risk of malware injection, leading to potential financial losses in the form of data loss, system downtime, and potentially damaged hardware. Furthermore, reputation damage is a significant concern.
- ❖ **Vulnerable and Outdated Components** - The use of outdated or vulnerable software components can result in data breaches, legal liabilities, and the potential loss of proprietary information. Financial consequences may arise from the need to address security breaches, legal disputes, and regulatory penalties.

## Impact of Network Vulnerabilities

- ❖ **Poor Firewall Configurations** - Inadequate firewall configurations can lead to unauthorized access and data breaches, incurring financial losses due to data theft, legal implications, and operational disruptions.
- ❖ **Insecure Wireless Networks** - Exploitation of insecure wireless networks can result in financial repercussions due to data breaches, operational disruptions, and potential legal actions. Reputation damage is also a concern, especially if customer data is compromised.
- ❖ **Insecure Incoming Emails** - Vulnerabilities in email security can lead to financial losses from phishing attacks, data breaches, and potential legal issues. Operational disruptions may occur if systems are compromised.
- ❖ **Social Engineering** - Social engineering exploits can result in severe financial losses, as they may lead to unauthorized access, data breaches, and operational disruptions. The loss of customer trust and reputation damage are additional concerns.
- ❖ **Outdated Or Unpatched Networks** - The financial impact of operating outdated or unpatched networks includes the costs associated with data breaches, system downtime, and potential regulatory fines.

## Impact of Web Application Vulnerabilities

- ❖ **SQL Injections** - SQL injections can lead to data breaches, financial losses, and potential legal actions. Reputation damage is also likely, as it may result in a loss of customer trust.
- ❖ **Cross-Site Scripting (XSS)** - XSS vulnerabilities can be exploited to steal sensitive data and execute malicious scripts, potentially leading to financial losses, data breaches, and reputation damage.
- ❖ **Cross-Site Request Forgery (CSRF)** - CSRF attacks can result in unauthorized actions on behalf of users, potentially leading to financial losses, reputation damage, and operational disruptions.
- ❖ **Security Misconfiguration** - Security misconfigurations can lead to unauthorized access, data breaches, and operational disruptions, incurring financial costs and reputation damage.

- ❖ **Broken Authentication** - Exploiting broken authentication may result in financial losses, data breaches, and reputation damage. The loss of customer trust is also a concern.
- ❖ **Directory Traversal** - Directory traversal vulnerabilities can be exploited to gain unauthorized access to sensitive data, potentially leading to financial losses, reputation damage, and operational disruptions.

The Business Impact Assessment provides a clear understanding of the multifaceted consequences of these vulnerabilities, guiding Mayo Industries in the prioritization of security measures and risk mitigation efforts.

## Effectiveness of Present Controls

The assessment of the effectiveness of the current security controls and measures in place at Mayo Industries plays a pivotal role in understanding the organization's existing security posture. This evaluation encompasses a comprehensive analysis of the strengths within the current security framework, providing valuable insights for enhancing cybersecurity preparedness.

### Strengths in the Existing Security Posture

**Employee Training and Awareness** - Mayo Industries invests in comprehensive employee training and awareness programs. Staff members are well-informed about security best practices and potential threats, serving as a crucial line of defense against social engineering and phishing attacks.[15]

**Incident Response Plan** - The organization boasts a well-defined incident response plan. This plan outlines clear steps for detecting, responding to, and recovering from security incidents, minimizing downtime and data loss.

**Network Monitoring** - Mayo Industries employs robust network monitoring tools to detect suspicious activities and potential threats in real time, facilitating rapid response and mitigation.

**Data Backup and Recovery** - Mayo Industries maintains a reliable data backup and recovery system. This capability ensures that in the event of data loss due to unforeseen incidents like hardware failures or cyberattacks, critical information can be swiftly restored, minimizing operational disruptions.

**Multi-Factor Authentication (MFA)** - The organization enforces the use of multi-factor authentication for accessing sensitive systems and data. MFA enhances authentication security by requiring multiple forms of verification, making it significantly more challenging for unauthorized users to gain access.

**Collaborative Security Culture** - The organization fosters a culture of collaboration when it comes to security. Employees are encouraged to report security concerns and incidents promptly, enabling rapid incident response and reducing the dwell time of potential threats.

**Security Compliance Adherence** - Mayo Industries adheres to relevant security compliance standards and regulations, ensuring that its security practices align with industry best practices and legal requirements. This commitment to compliance enhances data protection and risk mitigation.

**Security Incident Logging** - Comprehensive logging practices are in place to record security events and incidents. This aids in forensic analysis, enabling the organization to investigate and learn from security breaches and incidents.

## Conclusion

In the ever-evolving landscape of cybersecurity, organizations must remain proactive, vigilant, and adaptive to the myriad of threats that loom on the digital horizon. Mayo Industries, through the comprehensive assessment conducted by PentRus, has taken significant steps toward bolstering its security posture. Throughout this assessment, we have identified several strengths and potential areas for improvement in the organization's security landscape.

As we conclude this report, it is essential to acknowledge the assumptions made during this evaluation. The vulnerabilities and weaknesses assessed within this report were assumed based on known threat vectors and common security pitfalls. While these assumptions offer valuable insights into potential risks, they may not comprehensively capture every nuance of the organization's unique security landscape.

The organization's readiness to embrace these findings and proactively address them is commendable. By focusing on mitigating vulnerabilities, enhancing existing controls, and fostering a culture of security awareness, Mayo Industries can position itself as a resilient fortress against the ever-present and ever-adaptive cyber threats.

In an era where data breaches and cyberattacks have far-reaching consequences, investing in cybersecurity is an investment in the continuity and integrity of the organization. Mayo Industries has embarked on a path toward safeguarding its digital assets and preserving its reputation. As new threats emerge, the organization's willingness to adapt and fortify its security measures will be a key determinant of its resilience in the face of an ever-changing threat landscape.

## References

- [1] Sarang Tumne, "Practical Red teaming Field-Tested strategies for cyberwarfare", 2023, [online], Available: [1725939597913 \(licdn.com\)](https://www.linkedin.com/company/1725939597913)
- [2] S. Foster, "Vulnerabilities Definition: Top 10 Software Vulnerabilities," Perforce, July 2020. [Online]. Available: <https://www.perforce.com/blog/kw/common-software-vulnerabilities>.
- [3] N. Veerasamy, "High-level Methodology for Carrying out Combined Red and Blue Teams," IEEE, 2009.
- [4] N-able, "Top computer security & network vulnerabilities," N-able, 2023. [Online]. Available: <https://www.n-able.com/features/computer-security-vulnerabilities>.
- [5] Viit jain, "Wireshark Fundamentals A Network Engineer's Handbook to Analyzing Network Traffic", 2022, [Online] Available: <https://dl.ebooksworld.ir/books/Wireshark.Fundamentals.Vinit.Jain.Apress.9781484280010.EBooksWorld.ir.pdf>
- [6] Mahnoor Intizar, "Burp Suite User Manual Penetration Testing Report", 2024 Aug 05.
- [7] Michael Schearer, "SHODAN for Penetration Testers", [Online] Available: <https://media.defcon.org/DEF%20CON%2018/DEF%20CON%2018%20presentations/DEF%20CON%2018%20-%20Schearer-SHODAN.pdf>
- [8] M. a. M. S. C. C. C. Jason Firch, "Common Types Of Network Security Vulnerabilities," Purplesec, September 2023. [Online]. Available: <https://purplesec.us/common-network-vulnerabilities/>.
- [9] Vumetric, "10 Most Common Network Vulnerabilities," Vumetric Cybersecurity, 2023. [Online]. Available: <https://www.vumetric.com/blog/10-most-common-network-vulnerabilities/>.
- [10] O. Moradov, "8 Critical Web Application Vulnerabilities and How to Prevent Them," Brightsec, May 2022. [Online]. Available: <https://brightsec.com/blog/web-application-vulnerabilities/>.
- [11] G. Kalman, "10 Common Web Security Vulnerabilities," Developers, 2022. [Online]. Available: <https://www.toptal.com/cyber-security/10-most-common-web-security-vulnerabilities>.
- [12] D. R. V. a. S. Mayukha, "Port Scanning Mitigation Strategies for Penetration Testing: Blue Team Perspective," IEEE, 2022.

[13] A. Bio, "What are the Best Security Practices to Protect Against the Main Types of Attacks on Web Applications?," Indusface, October 2021. [Online]. Available: <https://www.indusface.com/blog/what-are-the-best-security-practices-to-protect-against-the-main-types-of-attacks-on-web-applications/>.

[14] Crowdstrike, "Purple Teaming Explained," Crowdstrike, February 2023. [Online]. Available: <https://www.crowdstrike.com/cybersecurity-101/purple-teaming/>.

[15] International Journal of Advances in Scientific Research and Engineering, "Evaluating the Security Posture and Protection of Critical Assets of Industrial Control Systems", 2022, . [Online]. Available: [file:///D:/YEAR%203%20SEMZ%2001%20Cyber%20Security/01\\_Applied%20Information%20Assurance%20-%20IE3022/00%20Assignment/Ass%20two/EvaluatingtheSecurityPostureandProtectionofCriticalAssetsofIndustrialControlSystemsinZambia.pdf](file:///D:/YEAR%203%20SEMZ%2001%20Cyber%20Security/01_Applied%20Information%20Assurance%20-%20IE3022/00%20Assignment/Ass%20two/EvaluatingtheSecurityPostureandProtectionofCriticalAssetsofIndustrialControlSystemsinZambia.pdf)