CyberShield

Security Assessment Findings Report

Business Confidential

Date: June 1st, 2024
Project: Praktikum 3
Version 1.0

Confidentiality Statement

This document is the exclusive property of CyberShield and TCM Security (JAY'S BANK). This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent of both CyberShield and JAY'S BANK.

CyberShield may share this document with auditors under non-disclosure agreements to demonstrate penetration test requirement compliance.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. JAY'S BANK prioritized the assessment to identify the weakest security controls an attacker would exploit. JAY'S BANK recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

Contact Information

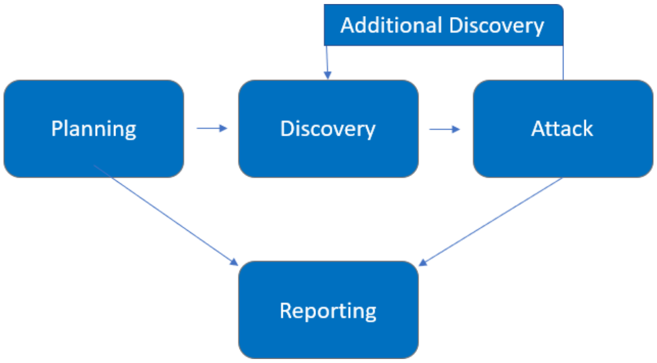
Name	Title	Contact Information
CyberShield		
John	Global Information Security Manager	Email: jsmith@democorp.com
Jay's Bank Security		
Jeany Aurellia	Lead Penetration Tester	Email: heath@tcm-sec.com

Assessment Overview

From June 28th, 2024 to 1st, 2024, CyberShield engaged JAY'S BANK to evaluate the security posture of its infrastructure compared to current industry best practices that included an internal network penetration test. All testing performed is based on the NIST *SP 800-115 Technical Guide to Information Security Testing and Assessment, OWASP Testing Guide (v4), and customized testing frameworks*.

Phases of penetration testing activities include the following:

- Planning – Customer goals are gathered and rules of engagement obtained.
- Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.



Assessment Components

Internal Penetration Test

An internal penetration test emulates the role of an attacker from inside the network. An engineer will scan the network to identify potential host vulnerabilities and perform common and advanced internal network attacks, such as: LLMNR/NBT-NS poisoning and other man- in-the-middle attacks, token impersonation, kerberoasting, pass-the-hash, golden ticket, and more. The engineer will seek to gain access to hosts through lateral movement, compromise domain user and admin accounts, and exfiltrate sensitive data.

Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V3 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Moderate	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Risk Factors

Risk is measured by two factors: Likelihood and Impact:

Likelihood

Likelihood measures the potential of a vulnerability being exploited. Ratings are given based on the difficulty of the attack, the available tools, attacker skill level, and client environment.

Impact

Impact measures the potential vulnerability's effect on operations, including confidentiality, integrity, and availability of client systems and/or data, reputational harm, and financial loss.

Scope

Assessment	Details
Jay's Bank Application Penetration Testing	167.172.75.216

Scope Exclusions

As per the client's directive, JAY'S BANK conscientiously abstained from engaging in the following attacks throughout the testing phase:

- All application functions.
- User account mechanisms and authentication.
- Web interfaces and APIs.
- Database interactions and data handling processes.

Client Allowances

CyberShield provided JAY'S BANK with the following allowances:

- Authorization is granted to search for and identify vulnerabilities within Jay's Bank application.
- Emphasis should be placed on vulnerabilities such as SQL injection, XSS, and authentication/authorization issues within the application.
- If feasible, discovered vulnerabilities may be exploited to access other user accounts, albeit restricted solely to the application (not the server).

Executive Summary

JAY'S BANK evaluated CyberShield's internal security posture through penetration testing from to June 28th, 2024 June 1st, 2024. The following sections provide a high-level overview of vulnerabilities discovered, successful and unsuccessful attempts, and strengths and weaknesses.

Scoping and Time Limitations

Scoping during the engagement did not permit denial of service or social engineering across all testing components.

Time limitations were in place for testing. Internal network penetration testing was permitted for five (5) days.

Testing Summary

The testing process, encompassing both Black-box Testing (BAC) and Cross-Site Scripting (XSS) evaluations, uncovered crucial insights into the security landscape of Jay's Bank application. Utilizing Black-box Testing, conducted with tools such as Burp Suite, vulnerabilities were unveiled, notably weaknesses within user authentication mechanisms and inadequately validated data inputs. The Cross-Site Scripting assessment aimed to identify vulnerabilities susceptible to script injection attacks.

This investigation pinpointed various entry points within the application vulnerable to XSS, potentially leading to session hijacking or user information compromise. These findings underscore the pressing need for immediate remediation efforts to bolster the application's security posture. Recommendations include fortifying authentication mechanisms, enhancing input data validation protocols, and implementing robust security controls to mitigate the risk of XSS attacks and safeguard user data.

Tester Notes and Recommendations

A vulnerability has been discovered in the application that allows access to other users' accounts solely by using their usernames. Furthermore, a Cross-Site Scripting (XSS) vulnerability has been uncovered, enabling the injection of malicious scripts that would trigger a pop-up alert when the affected page is loaded. These findings pose significant risks to the application's security and user data integrity.

The username-based account access vulnerability exposes a critical flaw in the authentication system, granting unauthorized individuals access to sensitive user information and potentially compromising data privacy. It is imperative to implement robust authentication measures, such as requiring both a username and a strong password, to mitigate this risk effectively.

Moreover, the XSS vulnerability creates an entry point for attackers to inject and execute malicious code on the client-side, potentially leading to data theft, session hijacking, or even further exploitation of the application's vulnerabilities. This issue underscores the importance of properly sanitizing and validating all user input, as well as implementing secure coding practices to prevent XSS attacks.

Key Strengths and Weaknesses

For scoop 167.172.75.216

The following identifies the key strengths identified during the assessment: The program's endpoints are highly secure, making it difficult for SQL injection and XSS attacks to penetrate.

The following identifies the key weaknesses identified during the assessment:

1. Vulnerability allowing account access using only the username.
2. Cross-Site Scripting (XSS) vulnerability enabling the injection of malicious scripts.

Vulnerability Summary & Report Card

The following tables illustrate the vulnerabilities found by impact and recommended remediations:

Internal Penetration Test Findings

13	5	6	0	1
Critical	High	Moderate	Low	Informational

Finding	Severity	Recommendation
Internal Penetration Test		
Finding 1: Broken Authentication and Access Control Vulnerability	High	Implement robust authentication measures by requiring both a username and a strong password for user accounts. Additionally, ensure proper access control mechanisms are in place to prevent unauthorized access to user accounts and sensitive information.
Finding 2: Cross-Site Scripting (XSS) Vulnerability	High	Properly sanitize and validate all user input to prevent the injection of malicious scripts. Implement secure coding practices, such as output encoding and input validation, to mitigate the risk of XSS attacks. Additionally, consider deploying a Web Application Firewall (WAF) to provide an additional layer of defense against XSS and other web-based attacks.

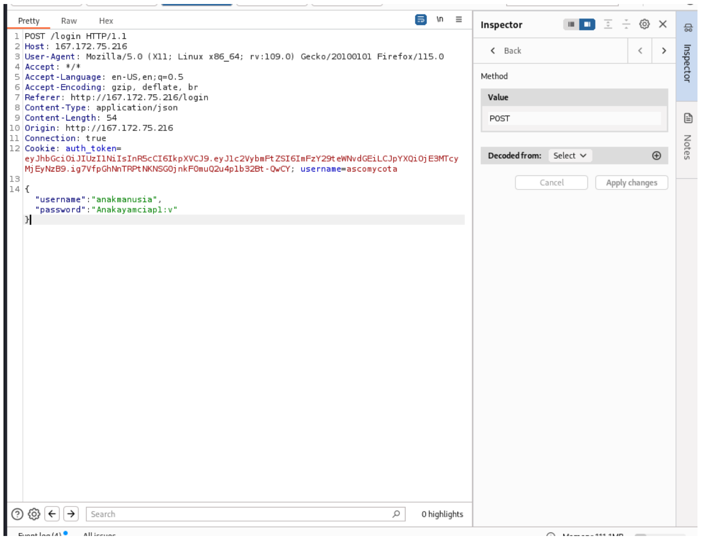
Technical Findings

Internal Penetration Test Findings

Finding 1: Broken Authentication and Access Control Vulnerability(High)

Description:	The application allows access to other users' accounts solely by using their usernames, bypassing the authentication process. This vulnerability exposes sensitive user information and potentially compromises data privacy.
Risk:	Likelihood: High - This vulnerability is highly likely to be exploited, as attackers can easily bypass authentication by simply using usernames. Impact: Very High - Successful exploitation can lead to unauthorized access to user accounts, data breaches, identity theft, and other malicious activities, resulting in severe financial and reputational damage.
System:	10.15.42.36
Tools Used:	Burp suite, Manual testing, web browser
References:	OWASP Top 10 2021 - A07:2021 – Identification and Authentication Failures

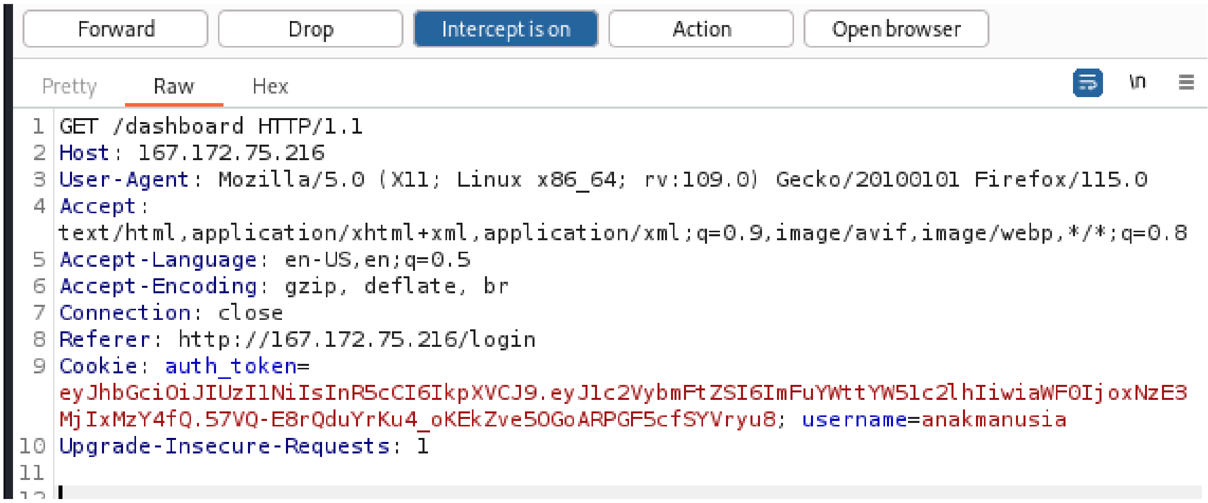
Evidence



```
POST /login HTTP/1.1
Host: 167.172.75.216
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: http://167.172.75.216/login
Content-Type: application/json
Content-Length: 54
Origin: http://167.172.75.216
Connection: true
Cookie: auth_token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFzY29teWNvdGEiLCJpYXQiOjE3MTcy
MjEyNzB9.ig7VfpGhNnTRPtNKNSG0jnkF0muQ2u4p1b32Bt-QwCY; username=ascomycota

{
  "username":"anakmanusia",
  "password":"Anakayamciapl:v"
}
```

Figure 1: change username with 2nd user and connection to true



```
GET /dashboard HTTP/1.1
Host: 167.172.75.216
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: close
Referer: http://167.172.75.216/login
Cookie: auth_token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFuYWttYW51c2lhIiwiaWF0IjoxNzE3
MjIxMzY4fQ.57VQ-E8rQduYrKu4_oKEkZve5OGoARPGF5cfSYVryu8; username=anakmanusia
Upgrade-Insecure-Requests: 1


```

Figure 2: Succesful login

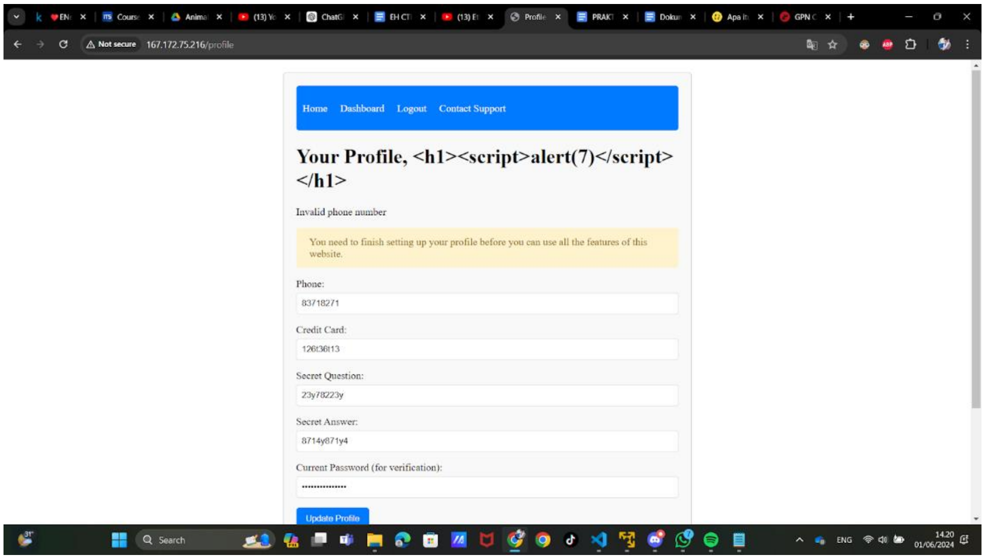
Remediation

- Implement a robust authentication mechanism that requires both a username and a strong password for user accounts.
- Enforce password complexity rules and regular password changes to enhance security.
- Implement multi-factor authentication (MFA) for an additional layer of security.
- Conduct regular security audits and penetration testing to identify and address any potential authentication and access control vulnerabilities.
- Implement access control mechanisms, such as role-based access control (RBAC), to ensure that users can only access resources and data they are authorized to access.
- Regularly review and update access controls to ensure they align with the principle of least privilege.
- Implement secure session management and logout functionality to prevent unauthorized access to active sessions.

Finding 2: Cross-Site Scripting (XSS) Vulnerability (High)

Description:	The application is susceptible to Cross-Site Scripting (XSS) attacks, which allow the injection of malicious scripts that can be executed in the user's browser when the affected page is loaded. This vulnerability can be triggered by injecting scripts into input fields or other user-controllable areas.
Risk:	Likelihood: High - XSS vulnerabilities are common in web applications and can be easily exploited if user input is not properly sanitized. Impact: Very High - Successful exploitation can lead to various attacks, including stealing user session tokens, hijacking user sessions, and potentially compromising the entire application or system, resulting in data breaches and financial losses.
System:	Web Application
Tools Used:	Manual testing, web browser, script injection
References:	OWASP Top 10 2021 - A03:2021 – Injection

Evidence



Your Profile, <h1><script>alert(7)</script></h1>

Invalid phone number

You need to finish setting up your profile before you can use all the features of this website.

Phone: 83718271

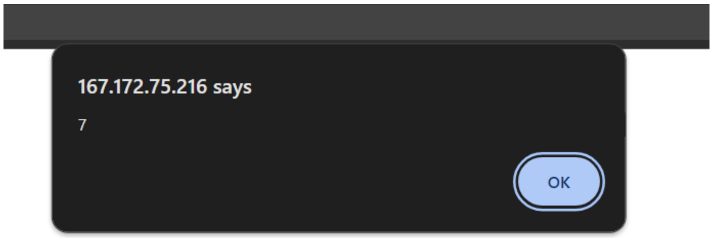
Credit Card: 126t36t13

Secret Question: 23y78223y

Secret Answer: 8714y871y4

Current Password (for verification): •••••••••••••••

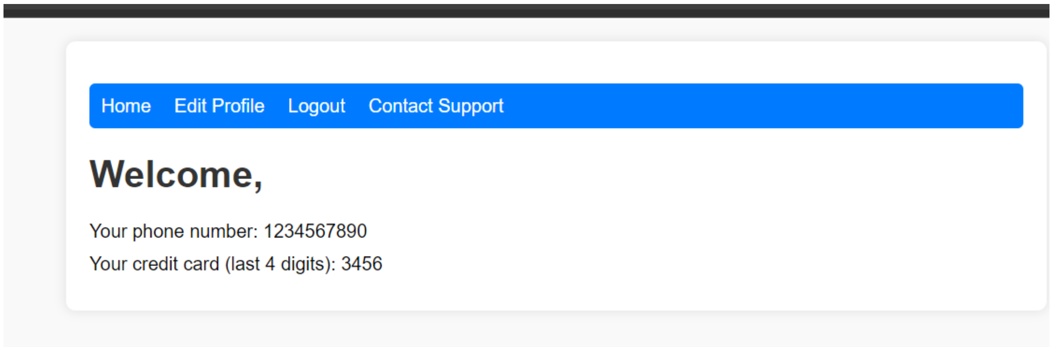
Figure 3: regist and login with script



167.172.75.216 says

7

Figure 4: Login again and appear popup



Welcome,

Your phone number: 1234567890

Your credit card (last 4 digits): 3456

Figure 5: dasboard after scripting

Remediation

- Implement input validation and sanitization for all user-supplied data to prevent the injection of malicious scripts.
- Use context-sensitive output encoding for all user-supplied data before rendering it in the browser.
- Implement a Content Security Policy (CSP) to restrict the execution of untrusted scripts and limit the impact of XSS vulnerabilities.
- Regularly update and patch the web application and its dependencies to address known XSS vulnerabilities.
- Implement a Web Application Firewall (WAF) to provide an additional layer of defense against XSS and other web-based attacks.
- Conduct regular security testing, including static
- code analysis and dynamic application security testing (DAST), to identify and address XSS vulnerabilities.
- Implement secure coding practices and provide security awareness training to developers to prevent the introduction of XSS vulnerabilities in the codebase.

Additional Scans and Reports

No additional scans or reports were included in this assessment. This report only covers the primary vulnerability findings identified during the web application security assessment.

Last Page