Jay's Bank
SafeGuard Solution

Business Confidential

Date: June 1st, 2024
Project: DC-001
Version 1

Confidentiality Statement

This document is the exclusive property of Safe Guard Jay's Bank Application. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent of both Safe Guard and Jays Bank.

Safe Guard 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
Jay's Bank		
Jay's	Global Information Security Manager	Email : jay'sBank@gmail.com
SafeGuard		
Rehana Putri Salsabilla	Expert Security	Email: rehanaputri80@gmail.com NRP : 5027221015

Assessment Overview

From May 28th, 2024 to June 1st, 2024, Safe Guard 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.

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..
- 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. 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
Internal Penetration Test	• 167.172.75.216

Scope Exclusions

- All application functions.
- User account and authentication mechanisms.
- web interface and API.
- Database interactions and data handling processes.

Client Allowances

- You are permitted to search for and identify vulnerabilities in Jay's Bank applications.
- Focus on application vulnerabilities such as SQL injection, XSS, and authentication/authorization issues.
- If possible, the vulnerabilities found can be exploited to access other user accounts, but only to the application (not to the server).

Executive Summary

SafeGuard Solutions conducted a comprehensive penetration test on Jay's Bank application from May 25th, 2024 to June 1st, 2024. The primary objective of this assessment was to identify potential security vulnerabilities within the application, focusing on areas such as user account mechanisms, web and API interfaces, and data handling processes. This executive summary provides an overview of the key findings, including identified vulnerabilities, the success and failure of various exploitation attempts, and an evaluation of the application's security posture.

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) business days.

Testing Summary

The testing process for Jay's Bank application included comprehensive evaluations of Broken Access Control and Cross-Site Scripting (XSS) vulnerabilities. Utilizing Burp Suite, we focused on identifying areas where access control mechanisms could be bypassed, allowing unauthorized access to sensitive information or functionalities. This method revealed several endpoints that permitted access without proper authentication or authorization checks, enabling attackers to perform actions and access data that should be restricted. Additionally, manual testing and automated tools like OWASP ZAP were used to uncover XSS vulnerabilities. This involved identifying points where user input was not properly sanitized, leading to potential script injection attacks. Multiple input fields across different pages of the application were found to be susceptible to XSS attacks, potentially allowing attackers to execute malicious scripts in the context of a user's browser, leading to session hijacking or theft of user information.

The impact of these findings is substantial. The ability to bypass access controls can lead to unauthorized access to sensitive data and functionalities, potentially compromising user privacy and the overall security of the application. XSS vulnerabilities, on the other hand, can result in the execution of arbitrary scripts, compromising user sessions and potentially leading to data breaches. To mitigate these risks, it is recommended to implement comprehensive access control mechanisms, ensuring that every request is authenticated and authorized. Regular reviews and tests of access control policies are necessary to prevent unauthorized access. Additionally, proper sanitization and validation of all user inputs are essential to prevent XSS attacks. Implementing secure coding practices and deploying security measures such as Content Security Policy (CSP) can further protect against XSS vulnerabilities.

These findings underscore the urgent need for remediation efforts to strengthen the security posture of Jay's Bank application. By addressing these vulnerabilities and implementing the recommended security measures, the application will be better protected against potential exploitation and data breaches, ensuring the integrity and confidentiality of user data.

Tester Notes and Recommendations

Several vulnerabilities have been discovered in the Jay's Bank application that pose significant risks to its security and user data integrity. Notably, the application is susceptible to Broken Access Control, allowing unauthorized access to sensitive information or functionalities, and Cross-Site Scripting (XSS) vulnerabilities, enabling the injection of malicious scripts that could compromise user sessions and expose sensitive information.

The Broken Access Control vulnerability was identified through the use of Burp Suite, revealing that certain endpoints could be accessed without proper authorization. This flaw allows attackers to perform actions and access data that should be restricted to authorized users only. To mitigate this risk, it is crucial to implement proper access control mechanisms, ensuring that every request is appropriately authenticated and authorized before granting access.

The XSS vulnerabilities present a significant threat, as they create entry points for attackers to inject and execute malicious scripts on the client side. This can lead to data theft, session hijacking, and further exploitation of the application's vulnerabilities. Proper sanitization and validation of all user input are essential to prevent such attacks. Implementing secure coding practices and deploying security measures such as Content Security Policy (CSP) can further protect against XSS vulnerabilities.

These findings underscore the urgent need for remediation efforts to bolster the security posture of Jay's Bank application. By addressing these vulnerabilities and implementing the recommended security measures, the application will be better protected against potential exploitation and data breaches, ensuring the integrity and confidentiality of user data.

Key Strengths and Weaknesses

Key Strengths:

- **Secure Network Configuration**: Robust network layer security with no unnecessary open ports or exposed services.
- **Solid Server Configuration**: Secure server configurations reducing the risk of server-level vulnerabilities.

Key Weaknesses:

1. **Broken Access Control**: Endpoints allowed access without proper authentication or authorization, enabling unauthorized actions and access to restricted data.
2. **Cross-Site Scripting (XSS) Vulnerability**: Multiple input fields were susceptible to XSS attacks, allowing malicious script injection, leading to session hijacking and data theft.

Vulnerability Summary & Report Card

Internal Penetration Test Findings

0	2	0	0	0
Critical	High	Moderate	Low	Informational

Finding	Severity	Recommendation
Penetration Test		
Finding 1: Cross-Site Scripting (XSS) Vulnerability	High	To mitigate the XSS vulnerability, it is crucial to properly sanitize and validate all user inputs, preventing the injection of malicious scripts. Implement secure coding practices such as output encoding and input validation to safeguard against XSS attacks. Additionally, consider deploying a Content Security Policy (CSP) and a Web Application Firewall (WAF) to provide an extra layer of defense against XSS and other web-based attacks. These measures will help protect user sessions and prevent data theft, ensuring the overall security of the application.
Finding 2: Broken Authentication and Access Control Vulnerability	High	To address the broken access control vulnerability, it is essential to implement comprehensive access control mechanisms. Ensure that every request is properly authenticated and authorized, preventing unauthorized access to sensitive data and functionalities. Regularly review and test access control policies to identify and rectify any potential weaknesses. This will help maintain the integrity and confidentiality of user data and protect against unauthorized actions.

Technical Findings

Exploitation Proof of Concept

Finding 1 : Cross-Site Scripting (XSS) Vulnerability

• **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 :**	- Manual testing
• **References:**	• OWASP Top 10 2021 - A03:2021 – Injection

Evidence :

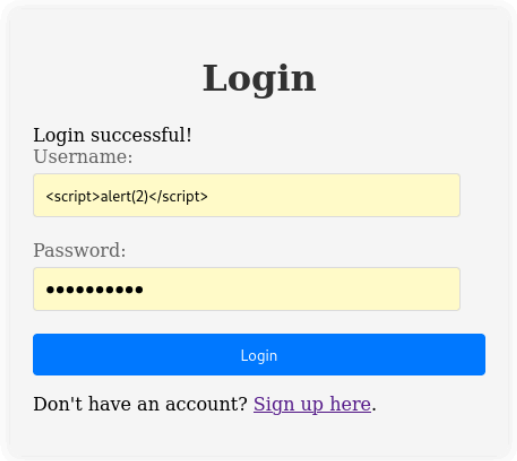

Figure 1: Login with username in the form of a script

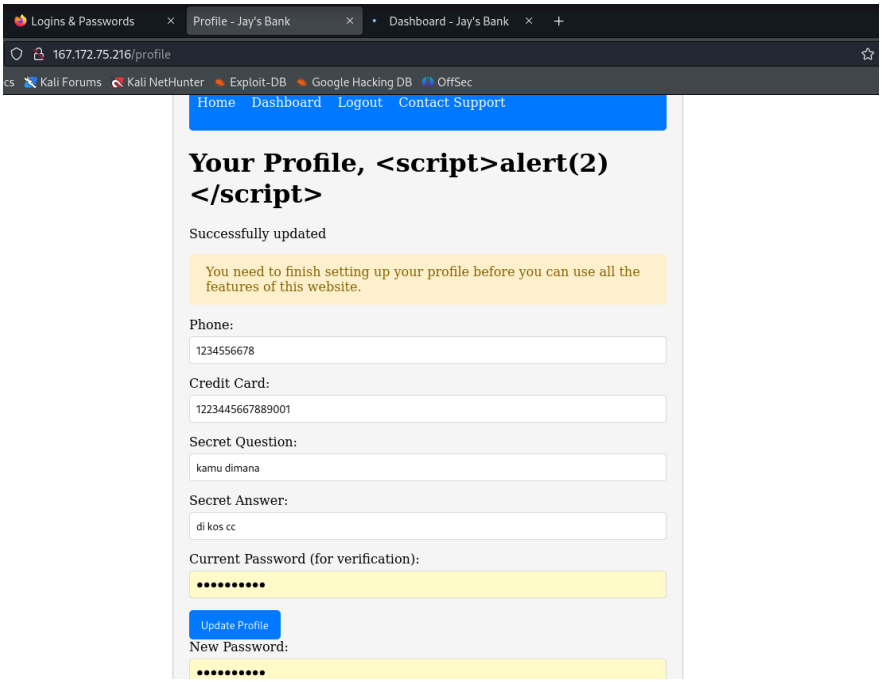

Figure 2: Update profile

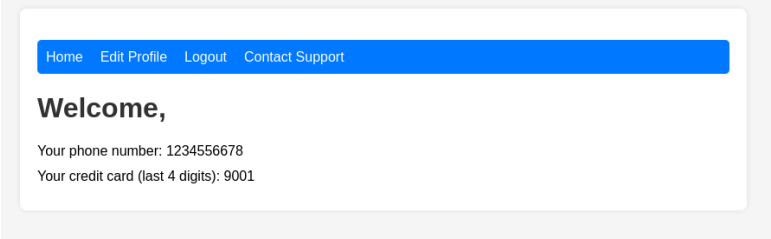

Figure 3: Display of phone number and credit card

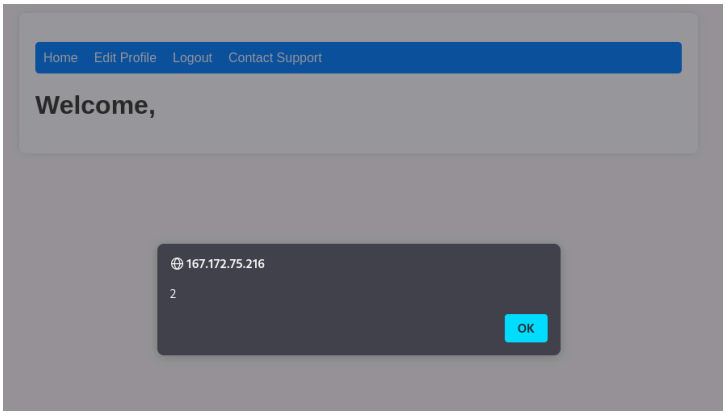

Figure 4: Pop up of alert from script

Remediation :

To remediate the Cross-Site Scripting (XSS) vulnerability discovered in the Jay's Bank application, several key steps must be taken. Firstly, robust input validation and sanitization procedures should be implemented across all user-controllable areas of the application. This ensures that any user input is thoroughly validated and sanitized to remove any potentially malicious scripts or HTML tags that could be executed in the user's browser. Additionally, output encoding techniques should be applied to all output data rendered in the application's responses. By encoding user-supplied data before rendering it in HTML pages, the risk of it being interpreted as executable code by the browser is mitigated. Furthermore, deploying a Content Security Policy (CSP) is essential. A CSP helps mitigate the impact of XSS attacks by specifying trusted sources of content and enforcing strict guidelines for loading external resources, thereby preventing the execution of injected scripts. It is also crucial to provide comprehensive security training to developers and stakeholders involved in the development process. This ensures they are aware of common XSS attack vectors and equipped with best practices for secure coding to prevent future occurrences of XSS vulnerabilities. Lastly, regular security assessments, including code reviews and dynamic testing, should be conducted to identify and remediate any remaining XSS vulnerabilities systematically. By implementing these remediation measures, Jay's Bank can effectively mitigate the risk of XSS vulnerabilities and enhance the overall security of the application, safeguarding against potential exploitation and data breaches.

Finding 2: Broken Authentication and Access Control Vulnerability

• **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:**	- 167.172.75.216
• **Tools :**	- Burp suite
• **References:**	• OWASP Top 10 2021 - A07:2021 – Identification and Authentication Failures

Evidence :

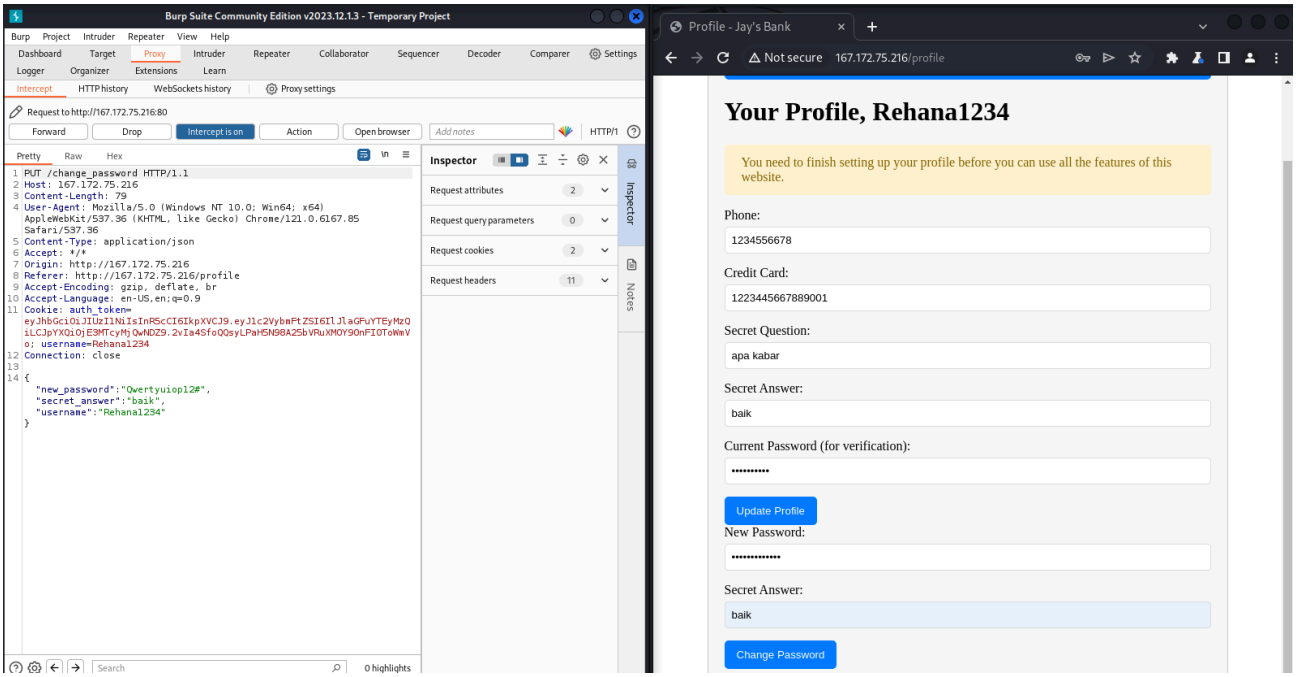

Figure 5: Update password of username Rehana1234

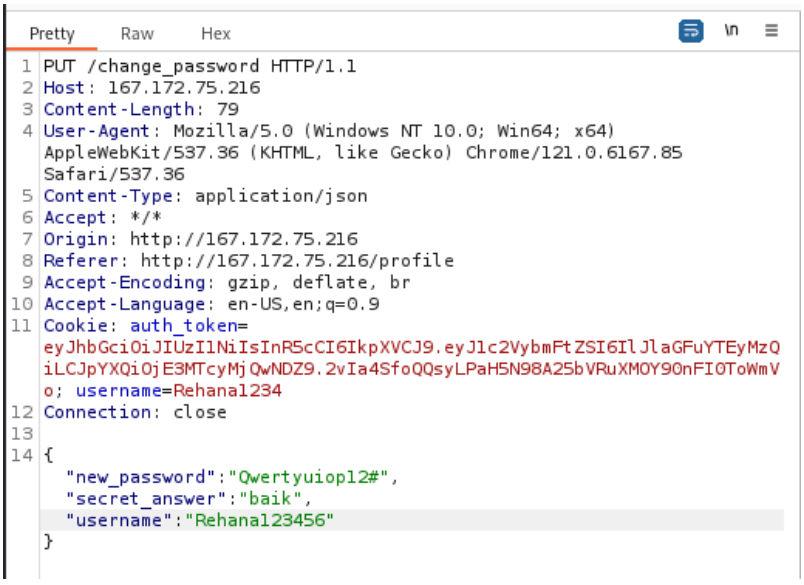

Figure 6: update username to Rehana1234556 with new password in username Rehana1234

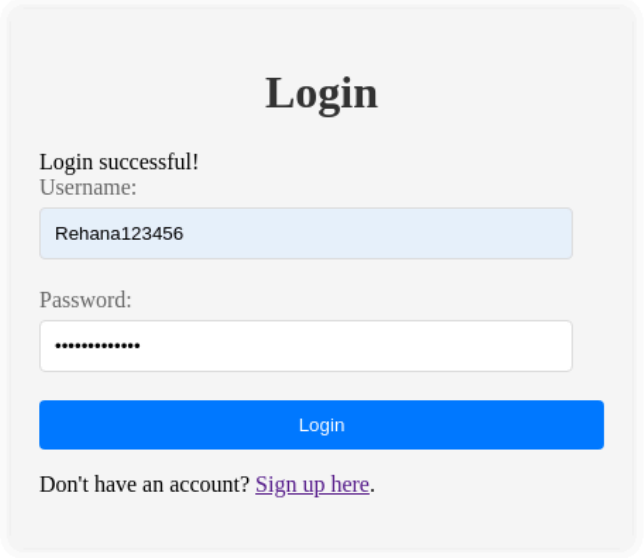

Figure 7: Log in with username Rehana123456 with password as username Rehana1234

Remediation :

To remediate the Broken Authentication and Access Control Vulnerability identified in the Jay's Bank application, several key actions need to be taken to strengthen the authentication mechanisms and access control policies. Firstly, it is imperative to implement robust authentication measures, including multifactor authentication (MFA) or strong password requirements, to prevent unauthorized access to user accounts. Additionally, implementing session management practices, such as session timeouts and secure cookie attributes, can help mitigate the risk of session hijacking. Moreover, access control mechanisms must be thoroughly reviewed and updated to ensure that every request is properly authenticated and authorized before granting access to sensitive functionalities or data. Role-based access control (RBAC) can be employed to assign specific privileges to users based on their roles within the application. Regularly reviewing and testing access control policies is essential to identify and address any potential weaknesses or misconfigurations that could be exploited by attackers. Finally, conducting security awareness training for users and administrators can help raise awareness about the importance of strong authentication practices and the risks associated with unauthorized access, contributing to a more secure application environment.

Additional Scans and Reports

In addition to internal penetration testing, further scans and reports can offer valuable insights into Jay's Bank application security. External vulnerability scans can identify potential entry points for attackers, like exposed ports or misconfigured services. Specialized assessments, such as web or mobile application penetration testing, provide targeted recommendations for improvement.

Last Page