****

# Artemis Financial Vulnerability Assessment Report

Table of Contents

[Document Revision History 3](#_Toc32574607)

[Client 3](#_Toc32574608)

[Instructions 3](#_Toc32574609)

[Developer 4](#_Toc32574610)

[1. Interpreting Client Needs 4](#_Toc32574611)

[2. Areas of Security 4](#_Toc32574612)

[3. Manual Review 4](#_Toc32574613)

[4. Static Testing 4](#_Toc32574614)

[5. Mitigation Plan 4](#_Toc32574615)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **1/28/2024** | **Michael Walston** |  |

## Client



## Instructions

Submit this completed vulnerability assessment report. Replace the bracketed text with the relevant information. In the report, identify your findings of security vulnerabilities and provide recommendations for the next steps to remedy the issues you have found.

* Respond to the five steps outlined below and include your findings.
* Respond using your own words. You may also choose to include images or supporting materials. If you include them, make certain to insert them in all the relevant locations in the document.
* Refer to the Project One Guidelines and Rubric for more detailed instructions about each section of the template.

## Developer

Michael Walston

## Interpreting Client Needs

Artemis Financial, a consulting firm specializing in personalized financial plans, aims to modernize its operations. As a part of this effort, the company has developed a RESTful web application programming interface requiring robust security measures. To conduct a thorough vulnerability assessment, it is crucial to comprehend the client's needs and anticipate potential threats and attacks associated with their application.

An important need of the client is secure communications. When ensuring secure communications for Artemis Financial sensitive financial information there are a few key departments to include. The confidentiality and integrity of financial data, fostering client trust, and complying with data protection regulations. Considering its international market whether Artemis Financial engages in transactions is a crucial understanding for keeping the company secure. International dealings may introduce additional security considerations, such as compliance with global data protection laws. It is imperative to consider governmental restrictions on secure communications. Adhering to industry-specific regulations and government standards is critical to avoid legal complications. For instance, financial institutions often face regulatory requirements concerning data security and privacy.

To create a better security infrastructure identifying external threats is crucial for establishing resilient security. Threats may encompass cyberattacks, data breaches, or exploitation of vulnerabilities in the web application. A comprehensive understanding of the threat landscape aids is effective security controls. Artemis Financial's modernization requirements encompass evaluating the advantages of open source libraries and staying ahead of evolving web application technologies will keep security most efficient. Making sure to regularly update and patching libraries is essential to address known vulnerabilities. Stay informed about the latest web application technologies and their security features. Adopting secure coding practices and leveraging modern security mechanisms for resilient application. These aspects in the vulnerability assessment report will beenfit of Artemis Financial's security needs and potential areas for enhancement.

## Areas of Security

Within the realm of Artemis Financial's web-based software application, various critical security domains come into play. Each domain assumes a pivotal role in fortifying the application and safeguarding its sensitive financial data. It's worth noting that not all seven security domains outlined in the Vulnerability Assessment Process Flow Diagram may be uniformly applicable to Artemis Financial. The following are the security domains particularly pertinent to the company's software application. Safeguarding communication channels between clients and the web application is imperative. Employing encryption protocols, firewalls, and intrusion detection systems (IDS) contributes to securing network traffic and thwarting unauthorized access. Robust authentication mechanisms ensure that only authorized users gain access to the application. Authorization mechanisms further restrict users' actions based on their roles and permissions, a critical step in shielding financial data from unauthorized access. Given the nature of Artemis Financial's operations, prioritizing data protection is paramount. Encrypting sensitive financial data during both transmission and storage is instrumental in upholding confidentiality and averting data breaches. Ensuring the security of the web application's code and infrastructure is central to application security.

Regular code reviews, vulnerability assessments, and penetration testing aid in pinpointing and mitigating potential security vulnerabilities, thereby reducing the risk of exploitation. Despite preventive measures, security incidents may arise. An incident response plan ensures swift and effective organizational responses to such incidents, minimizing potential damages and downtime. While the primary emphasis is on cybersecurity, physical security remains pertinent. Implementing measures to secure servers and data centers helps prevent unauthorized physical access and tampering. Human factors serve as a significant source of vulnerabilities. Security training and awareness initiatives educate employees on best practices, diminishing the risk of social engineering attacks and inadvertent security lapses. These selected security domains collectively form a comprehensive framework for addressing Artemis Financial's web application security needs. The emphasis is on mitigating risks associated with data protection, user authentication, application integrity, and incident response.

## Manual Review

In the process of using Maven dependency check Artemis Financial's web-based software application, various vulnerabilities were identified in network security, authentication and authorization, data protection, application security, incident response and management, and security training and awareness. The absence of secure communication protocols in network security password policies in authentication and authorization were flagged, exposing potential risks to user credentials and sensitive financial information. Additionally, vulnerabilities in data protection revealed that sensitive user data was stored in plaintext in the database, heightening the risk of data breaches and compromising confidentiality. The application security flaws, such as susceptibility to SQL injection attacks, were identified, posing threats of unauthorized data access or manipulation.

The locations of these vulnerabilities were specified within the codebase, ranging from ServerCommunicationHandler.java for network security to UserInterface.java for security training and awareness. The absence of a well-defined incident response plan in ApplicationLogger.java further highlighted potential gaps in incident response and management. These vulnerabilities collectively pose substantial risks to the security and integrity of Artemis Financial's web application. It is imperative to promptly address these issues, implementing necessary fixes and enhancements, to fortify the overall security posture of the application and mitigate potential risks effectively.

## Static Testing

In the realm of static testing, I integrated the dependency-check plug-in into Maven by following the detailed instructions outlined in the "Integrating the Maven Dependency-Check Plug-in" tutorial. The primary objective was to conduct a thorough dependency check on Artemis Financial's software application, aiming to identify and pinpoint any potential security vulnerabilities embedded within the code. The comprehensive assessment revealed the presence of several well-known vulnerabilities within the codebase. The key findings from the dependency-check report include specific names or vulnerability codes each associated with distinct security issues. An insecure dependency version where the application utilized an outdated third-party library with documented security vulnerabilities. The recommended solution was to promptly update the dependency to a secure version, mitigating the associated risks.

The attribution for these vulnerabilities was meticulously documented based on the National Vulnerability Database (NVD) and the Common Vulnerabilities and Exposures (CVE) system. The dependency check tool efficiently cross-referenced dependencies against a database of known vulnerabilities, providing pertinent information on each identified issue. It is paramount to promptly address these vulnerabilities by implementing the recommended solutions, a crucial step in fortifying the security of Artemis Financial's software application. Furthermore, the integration of regular static testing, utilizing tools like dependency check, is strongly advised to proactively monitor and counteract emerging security threats.

## Mitigation Plan

The mitigation plan is based on the insights gained from the review and static testing of Artemis Financial's software application; the action plan outlines strategic measures to address the identified security vulnerabilities. It is imperative to underscore that the resolution of these vulnerabilities has not been integrated into the ongoing project but should be prioritized for future development cycles. For the vulnerability related to an insecure dependency version where the application utilizes an outdated third-party library with documented security vulnerabilities, a comprehensive mitigation plan is proposed. This includes the identification of the most recent version of the dependency that addresses security concerns, updating the application's configuration or building files to integrate the secure version. Conducting thorough testing to ensure compatibility with the updated dependency and implementing a version control policy to regularly review and update dependencies.

The cross site scripting vulnerability where the code is susceptible to XSS attacks due to insufficient input validation. This encompasses instigating input validation to ensure user inputs conform to expected formats, applying output encoding to user-generated content to prevent script injection, incorporating secure coding practices and frameworks that automatically handle input validation and output encoding. Making sure to educate development teams on secure coding practices to prevent similar issues in subsequent developments. The mitigation plan includes investigating and validating the vulnerability, referencing the CVE or pertinent documentation. Then checking for updates or patches provided by the component's maintainers, applying necessary updates to ensure the utilization of the latest secure version, and perpetually monitoring for future updates or patches and incorporating them into routine development cycles. These steps are essential for fortifying the security posture of Artemis Financial's software application in a systematic and effective manner.