****

# 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** | **March 17, 2023** | **Hutch Ellis** |  |

## 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

Hutch Ellis

## Interpreting Client Needs

Artemis Financial is a consulting company that develops individualized financial plans for their customers. The company wants to modernize their operations by using the most effective and up to date security software. The company has a RESTful web application interface (API) that needs to be protected from external threats.

* The value of secure communications is crucial for Artemis Financial because they must safeguard their clients’ financial information in order to maintain customer trust.
* Artemis Financial conducts international transactions which make it vulnerable to data breaches and cyberattacks.
* Artemis Financial must comply with government restrictions and regulations to ensure secure communication and data protection.
* There are many different external attacks the company could be susceptible to, such as data breaches, phishing, SQL injection, or DDoS attacks.
* When it comes to the modernization requirements, we should consider what changes need to be made to implement the up-to-date security software and how well it pairs with the API.

## Areas of Security

Based on the needs of Artemis Financial, some security measures to consider would be:

* Authentication and Authorization – This would ensure that only authorized users have access.
* Input Validation and Output Encoding – This would help to prevent common vulnerabilities such as cross-site scripting (CSS) or SQL injections.
* Data Protection – Encrypting sensitive data will help to protect against potential data breaches. Data should be encrypted while and rest and in transit.
* Session Management – User sessions should be securely managed to prevent session hijacking.
* Error Handling and Logging – Error messages should be provided to users without exposing any sensitive information.

## Manual Review

During the manual review, I should focus on the areas of security I addressed in the previous step to address any potential vulnerabilities.

* Authentication and Authorization – Potential vulnerabilities could be weak password policies, missing authorization checks, and improper access control.
* Input Validation and Output Encoding – Potential vulnerabilities could be unsanitized user inputs and missing output encoding.
* Data Protection – Potential vulnerabilities could be lack of encryption during data transmission and lacking secure storage of sensitive data.
* Session Management – Potential vulnerabilities could be insecure session tokens and unsuitable handling of expired sessions.
* Error Handling and Logging – Potential vulnerabilities could be leaked information through error messages or insufficient security event loggings.

## Static Testing

During static testing, I am integrating the dependency check plug-in into Maven and running a dependency check on the Artemis Financial software.

Text, letter

Description automatically generated

After reviewing the report I can see there are a total of 111 vulnerabilities found within 14 vulnerability dependencies. I have provided a table of the vulnerabilities along with the number of vulnerabilities of each.

|  |  |
| --- | --- |
| Vulnerability Name | Number of Vulnerabilities |
| bcprov-jdk15on-1.46.jar | 17 |
| hibernate-validator-6.0.18.Final.jar | 1 |
| jackson-databind-2.10.2.jar | 4 |
| log4j-api-2.12.1.jar | 1 |
| logback-core-1.2.3.jar | 1 |
| snakeyaml-1.25.jar | 8 |
| spring-boot-2.2.4.RELEASE.jar | 1 |
| spring-boot-starter-web-2.2.4.RELEASE.jar | 1 |
| spring-core-5.2.3.RELEASE.jar | 9 |
| spring-expression-5.2.3.RELEASE.jar | 10 |
| spring-web-5.2.3.RELEASE.jar | 10 |
| spring-webmvc-5.2.3.RELEASE.jar | 9 |
| tomcat-embed-core-9.0.30.jar | 19 |
| tomcat-embed-websocket-9.0.30.jar | 20 |

## Mitigation Plan

The mitigation plan is used to address results from the manual review and static testing report.

Based on the results of the manual review here are some things that we should consider:

* Authentication and Authorization – Create stronger password policies, ensure authorization checks are in place, and have proper access control mechanisms.
* Input Validation and Output Encoding – All user inputs should be sanitized and apply output encoding when displaying data from an untrusted source.
* Data Protection – Sensitive information should be stored securely using encryption and remain encrypted when transferred.
* Session Management – Secure session tokens should be used, and session expiration should be handled properly to prevent unauthorized access.
* Error Handling and Logging – Error messages could be implemented into their own error pages in order to prevent potential data leaks and to ensure sufficient logging of security events.

We should also consider the numerous vulnerabilities that were present during the static testing.