****

# 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** | **7/12/2023** | **Elana Masboob** | **Client’s needs, areas of security, dependency checks, and mitigation plan added.** |

## 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

Elana Masboob

## Interpreting Client Needs

Artemis Financial specializes in providing financial plans to its clients. Due to the sensitive nature of this financial information, it is crucial to ensure secure communication between the company and its customers. Compliance with government regulations pertaining to financial transactions and communications imposes specific data retention policies and security requirements. RESTful APIs, including Artemis Financials, are susceptible to data interception if not securely designed. To mitigate these risks, the service must adhere to the following measures: utilizing HTTPS for all communications, transmitting confidential information through secure channels such as request and response headers or bodies, and implementing a robust authentication scheme like OAuth.

## Areas of Security

Based on the Vulnerability Assessment Process Diagram, Artemis Financials software application has the following areas of security:

* Validation of user input is required for the RESTful API to ensure data integrity.
* The web service incorporates a RESTful API that necessitates secure communication.
* Handling code errors caused by invalid user input must be done securely.
* Implementation of secure coding practices is essential.

## Manual Review

At a glance, these are the vulnerabilities present in the code:

- Request parameters in the CRUDController class contain business names.

- DocData class contains hard-coded database connection parameters.

- Lack of validation for request parameters.

- The service does not utilize HTTPS.

- Absence of an authentication scheme.

## Static Testing

Dependency check was run on Artemis Financial current software identifies security vulnerabilities. These vulnerabilities include, but are not limited to:

A screenshot of a computer

Description automatically generated

1. **bcprov-jdk15on-1.46.jar**
   * CVE-2013-1624
   * CVE-2015-6644
   * CVE-2015-7940
   * CVE-2016-1000338
   * CVE-2016-1000339
   * CVE-2016-1000341
   * CVE-2016-1000342
   * CVE-2016-1000343
   * CVE-2016-1000344
   * CVE-2016-1000345
   * CVE-2016-1000346
   * CVE-2016-1000352
   * CVE-2017-13098
   * CVE-2018-1000613
   * CVE-2015-5382

**Recommendation: update to at least 2.10.5.1**

1. **jackson-databind-2.10.2.jar**
   * CVE-2020-25649

**Recommendation: Update to at least 2.10.5.1**

1. **Apache Log4j API 2.12.1**
   * CVE-2020-9488

**Recommendation: Update to at least 2.13.2**

1. **SnakeYAML 1.25**
   * CVE-2017-18640

**Recommendation: Update to at least 1.26**

1. **Spring Core 5.2.3**
   * CVE-2020-5421

**Recommendation: Update to at least 5.2.8**

1. **Apache Tomcat 9.0.30**
   * CVE-2019-17569
   * CVE-2020-11996
   * CVE-2020-13934
   * CVE-2020-13935
   * CVE-2020-13943
   * CVE-2020-17527
   * CVE-2020-1935
   * CVE-2020-1938
   * CVE-2020-8022
   * CVE-2020-9484
   * CVE-2021-24122

**Recommendation: Update to at least 9.0.40**

## Mitigation Plan

After analyzing the results of the Dependency Report, an effective mitigation plan should include the following:

* + Transitioning to the HTTPS protocol for all types of communication.
  + Relocating request parameters from the URI to headers or the request body.
  + Eliminate embedded database connection credentials.
  + Implement a secure authentication mechanism.
  + Upgrade dependencies as specified above.