****

# 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** | **1/27/2024** | **Andrew Piccirilli** | **Providing Security Information** |

## 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

Andrew Piccirilli

## Interpreting Client Needs

* **interpreting Client Needs**: Review the scenario to determine your client’s needs and potential threats and attacks associated with their application and software security requirements. Document your findings in your vulnerability assessment report. Consider the scenario information and the following questions regarding how companies protect against external threats:
  1. What is the value of secure communications to the company?
     1. Secure communications provides security for their consumers. If the clients information is protected they’ll feel better using this service.
  2. Does the company make any international transactions?
     1. The company is a web based company, so they do handle international transactions. The software is for entrepreneurs, businesses, and government agencies around the world.
  3. Are there governmental restrictions about secure communications to consider?
     1. Being that this is a web based company designed for government agencies etc. around the world there is a lot to consider as far as restrictions about secure communications. Not every country will play by the same rules, so the program will have to meet each and every regions guidelines for secure communications practices. This can be quite difficult to handle, and will require a lot of effort to make sure everyone’s needs/regulations are met.
  4. What external threats might be present now and in the immediate future?
     1. Being a web based program with a lot of sensitive information, it’s a prime target for hackers and possible cyber attacks. So viruses and data leaks are a real issues to be aware of and plan to protect against.
  5. What are the modernization requirements that you must consider? For example:
     1. The role of open-source libraries
        1. Open source libraries allow developers to review and share code with other developers that can use that to improve the software they are working on. By working together to supply open source libraries there is a trove of knowledge to help keep your program up to date and functioning the way it needs to, without having to do everything from scratch.
     2. Evolving web application technologies
        1. Web application technologies are constantly evolving. With the use of mobile devices, people are now using their smartphones to handle just about every aspect of their lives with very little use for computers, unless they are in an industry that requires more advanced processes. Serverless architecture, web security, mobilization, etc. there’s a lot to think about and as we progress further the more keeping up with the trends in web application technologies will benefit. By being outdated you will suffer security risks and accessibility issues to say the least.

## Areas of Security

API- API’s provide data protection through encryption, authentication, authorization, keys/tokens, protection against SQL injection, etc. The use of API in web applications such as this is essential, especially when part of the company slogan is “Security is everyone’s responsibility”.

Cryptography- Encrypts sensitive data to prevent unauthorized access, provides more secure use of API, protects from possible internal threats, provides password security, protects data over untrusted networks, etc. For a web application with sensitive information cryptography for data protection is a must.

Client/Server- Protects through the use of encryption, authentication, access control, secure communication, while allowing monitoring and incident response.

Secure Coding-Protects further through use of input validation, vulnerability mitigation, error handling, code reviews, testing, secure sessions management, etc.. Through secure coding the chances of any kind of threat succeeding are drastically reduced.

## Manual Review

* Doesn’t use HTTPS
* No authentication in place
* Requests are not validated
* Business names sent as request parameters in CrudController
* User name/password issues in the DocData.java file making passwords easy to guess/access

## Static Testing

* bcprov-jdk15on-1.46.jar
  1. Has multiple vulnerabilities which can be fixed by updating bouncy castle to the latest version
* log4j-api-2.12.1.jar
  1. CVE-2020-9488
     1. Fixed in Apache Log4j 2.12.3 and 2.13.1
* snakeyaml-1.25.jar
  1. multiple vulnerabilities fixed by updating to latest version
* jackson-databind-2.10.2.jar
  1. multiple vulnerabilities fixed by updating to latest version
* tomcat-embed-core-9.0.30.jar
  1. multiple vulnerabilities fixed by updating to latest version
* hibernate-validator-6.0.18.Final.jar
  1. CVE-2020-10693
     1. Update to latest version
* spring-boot-2.2.4.RELEASE.jar-fix all by updating to latest version
  1. CVE-2022-27772
  2. CVE-2023-20873
  3. CVE-2023-20883

## Mitigation Plan

First step to making sure the application is secure is making sure that all the dependencies are up to date, by using out of date dependencies/plugins even with proper coding practices you’re opening yourself up to vulnerabilities. After that’s fixed, going back through the code and fixing the issues with authentication, password creation, and making sure to implement secure coding practices will go a long way. Two factor authentication, cryptography, API’s and client/server will beef up the authentication and security of the program and should not be thought of as an option for a multinational web program. Stacking those security features together will ensure that hacking attempts/cyber attacks are near impossible.