****

# 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** | **[09/17/2023]** | **Thomas Cimino** |  |

## 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

Thomas Cimino

## Interpreting Client Needs

Artemis Financial is a financial consulting firm that manages finance place for individuals. Secure communication is essential here, as they handle sensitive information. SSNs, tax information, banking information, etc. are all handled by Artemis Financial. It’s possible that Artemis is doing international transactions, not just within the US. If this is the case, we must follow government regulations when it comes to those transactions, possibly even protected by a third party.

## Areas of Security

* Code Errors – Implementation of error handling will help us better see what areas in the code are weak and need to be fixed.
* Cryptography – Ensures that all information is properly encrypted to ensure its protected.
* Api – Controls what information is assessable to who, and what data is accepted.
* Code Quality – Will control who can access what information through the API.
* Input Validation - Used to validate account owner information.

## Manual Review

* HTTPS is not used, which is crucial in sharing sensitive data.
* Business names are sent as request parameters, this leaves the system vulnerable.
* No signs of cryptography used.
* No authentication system in place to verify account owners.

## Static Testing

|  |  |  |  |
| --- | --- | --- | --- |
| Dependency | Vulnerability | Desc. | Possible Solution |
| Bcprov-jdk15on-1.46.jar | cpe:2.3:a:bouncycastle:bouncy-castle-crypto-package:1.46:\*:\*:\*:\*:\*:\*:\*  cpe:2.3:a:bouncycastle:bouncy\_castle\_crypto\_package:1.46:\*:\*:\*:\*:\*:\*:\* | A lot of vulnerabilities | Update to 1.60 |
| Log4j-api-2.12.2.jar | cpe:2.3:a:apache:log4j:2.12.1:\*:\*:\*:\*:\*:\*:\* | Improper validation of certificate with host mismatch in Apache Log4j SMTP appender. | Update to 2.13.2 |
| Snakeyaml-1.25.jar | cpe:2.3:a:snakeyaml\_project:snakeyaml:1.25:\*:\*:\*:\*:\*:\*:\* |  | Update SnakeYAML engine |
| Tomcat-embed-core-9.0.30.jar | cpe:2.3:a:apache:tomcat:9.0.30:\*:\*:\*:\*:\*:\*:\*  cpe:2.3:a:apache\_software\_foundation:tomcat:9.0.30:\*:\*:\*:\*:\*:\*:\*  cpe:2.3:a:apache\_tomcat:apache\_tomcat:9.0.30:\*:\*:\*:\*:\*:\*:\* |  | Update to 10.0.6 or later |
| Jackson-databind-2.10.2.jar | cpe:2.3:a:fasterxml:jackson-modules-java8:2.10.2:\*:\*:\*:\*:\*:\*:\* | Vulnerability to XXE attacks. | Update to current version |
| Spring-aop-5.2.3.RELEASE.jar | cpe:2.3:a:pivotal\_software:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*  cpe:2.3:a:springsource:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*  cpe:2.3:a:vmware:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*  cpe:2.3:a:vmware:springsource\_spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\* | Protections against RFD attacks may be bypassed. | Upgrade to current version |
| Hibernate-validator-6.0.18.final.jar | cpe:2.3:a:redhat:hibernate\_validator:6.0.18:\*:\*:\*:\*:\*:\*:\* | A bug in the message interpolation enables invalid EL expressions. | Upgrade to 6.0.20 |

## Mitigation Plan

The steps to remedy the vulnerability findings would be to update all systems to their current versions and implementing HTTPs to secure online data. Some form of authentication system should also be put into place.