****

# 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** | **09/16/23** | **Mark Carrillo** |  |

## 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

Mark Carrillo

## Interpreting Client Needs

Artemis Financial is a company that deals with financial plans for customers in regard to their savings, retirements investments and insurance. All of these plans involve people’s money and livelihoods, so we have to understand our clients’ needs in order to establish a safe and secure product for their customers. The value of secure communications to the company is enormous because there is a significant amount of sensitive information (SSN’s,DOBs..) being handled as well as money. Upon first inspection there’s no real mention of whether the company does or doesn’t deal with international transactions so for arguments sake we will be judging it based on if it does. Most international governments require that we ensure that communications and information are kept confidential/private. Some countries also require that we disclose what information is being kept where as to ensure that info is not being sold to other parties. The main threat that we would be facing as a financial product would be other parties targeting client’s information via hacks/data breaches. We will need to make sure that data is heavily encrypted and only stored in safe locations as well as ensuring any employees that handle or have access to this information is trained against phishing/social engineering plots. Some modernization requirements should be scheduled bug fixes/firmware updates as well as checking for vulnerabilities on a regular basis.

## Areas of Security

A yellow and blue sign with black text

Description automatically generatedA yellow and blue square with black text

Description automatically generatedA diagram of a computer code

Description automatically generatedA yellow and blue sign with black text

Description automatically generated

For Artemis financial as shown above all I personally believe these 5 areas of security apply to this company’s product for numerous reasons. Input validation because we need to ensure that we validate the users information is correct with the expected user. API’s ensure that we have control over which information is being accessed from other sources such as users, admins, and network administrators. Code error and code quality go hand in hand because we need to make sure that our code is both strong and correct. What Is meant by this is that we ensure no vulnerabilities are found in our code that would allow wrong parties access to the system and therefore we need to make sure our code quality is up to professional standards. Code errors would make sure that we have no areas of code where there is weak or spaghetti code causing our system to either run slow or be vulnerable.

## Manual Review

When taking the code into my hands and visually inspecting it, the first red flag I happened to notice the lack of some security features built into the code. The focus on cryptography is definitely lacking in some of the files I’ve come to read. Some of the code was acceptable but there was some weird input validation or lack thereof. The main lack of input validation I noticed immediately was in greeting controller because once the user inputs some info its kind of up in the air.

## Static Testing

A screenshot of a computer

Description automatically generated

Upon reading most of the documentation on these errors what I’ve come to see is that most vulnerabilities just require that we update most of our system to the latest firmware. For example for the tomcat error we need to upgrade it to a later version. Same thing with bounty castle,hibernate validator and snakeyaml errors

## Mitigation Plan

The very simple and effective plan I would have set forth to artemis at this point would be to ensure that we update to the current versions of all the software’s previously mentioned. These softwares include bountycastle, apache tomcat, hibernator validator and snakeyaml. Once these fixes are in we can circle back and focus on more input validation.