****

# 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** | **November 8, 2022** | **Mason Utt** |  |

## 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

Mason Utt

## Interpreting Client Needs

1. What is the value of secure communications to the company?

As a banking institution Artemis Financial should place high value on secure communication. This is because usernames, passwords, transactions, and all other sensitive information needs to be secure.

1. Does the company make any international transactions?

The information provided does not indicate if Artemis Financial does or does not have international transactions. As it is a large bank, I would assume that they do have international transactions for safe practices.

1. Are there governmental restrictions about secure communications to consider?

In general, most government restrictions state that financial transactions and communications need some sort of security requirements and data retention policies.

1. What external threats might be present now and in the immediate future?

Artemis Financial may be vulnerable to external threats such as data interception, ransomware, DDoS, and malware.

1. What are the modernization requirements that you must consider?

Open-source libraries need to be used but with consideration due to known vulnerabilities. HTTPS should be used as it ensures a secure connection with the website. Always staying up to date with maintenance and security practices helps prevent against new common exploits.

## Areas of Security

The areas of security to apply to Artemis Financials’ web application according to the Vulnerability Assessment Process Flow Diagram would be Input Validation, APIs, Cryptography, and Code Error.

* Input Validation: The web application needs to validate user information, so that only the correct information will be passed and checked.
* APIs: It is important to have secure interactions. The API should make sure only secure communications exist. It should not allow for interactions to happen to none authorized sites.
* Cryptography: Passwords, usernames, transactions, and some currencies would need to be encrypted. Otherwise, sensitive user information could be lost which would delegitimize Artemis Financial.
* Code Error: Error handling should be done so it is easier to know what corrections need to be implemented. Helps prevent user information from being accessed.
* Code Quality: It is important to follow industry standards for coding so maintenance and upgrades can happen.

## Manual Review

There is no input validation in GreetingController.java this fails the input validation area of security. In DocData.java the database is hard coded instead of using variable names this fails code quality. Business name is passed as a RequestParam without input validation. In Pom.xml, the code does not use HTTPS.

## Static Testing

There exist vulnerabilities for BouncyCastle, FasterXML, RedHat, Apache Log4j, Logback- Core, SnakeYAML, Spring Boot, Spring Core, Spring Web, and Apache Tomcat. Bouncy Castle has 17 vulnerabilities: CVE-2016-1000338, CVE-2016-1000342, CVE-2016-1000343, CVE-2016-1000344, CVE-2016-1000352, CVE-2016-1000341, CVE-2016-1000345, CVE-2017-13098, CVE-2020-15522, CVE-2020-0187, CVE-2016-1000339, CVE-2020-26939, CVE-2015-7940, CVE-2018-5382, CVE-2013-1624, CVE-2016-1000346, CVE-2015-6644. I recommend upgrading to BouncyCastle version 1.61 at least, but best to upgrade to the latest version. RedHat has one vulnerability CVE-2020-10693, I suggest upgrading to the latest version. FasterXML has 4 vulnerabilities: CVE-2020-25649, CVE-2020-36518, CVE-2022-42003, CVE-2022-42004. I suggest upgrading to at least 2.14.0 or newer. Apache Log4j has 1 vulnerability, CVE-2020-9488, I suggest upgrading to at least 2.13.1 or newer. Logback-Core has 1 vulnerability, CVE-2021-42550, I suggest upgrading to version 1.2.7 or newer. SnakeYAML has 6 vulnerabilities: CVE-2017-18640, CVE-2022-25857, CVE-2022-38749, CVE-2022-38751, CVE-2022-38752, CVE-2022-38750. I suggest upgrading to at least 1.32 or newer. Spring Boot has 1 vulnerability, CVE-2022-27772, I suggest upgrading to version v2.2.11.RELEASE or newer. Spring Core has 9 vulnerabilities: CVE-2022-22965, CVE-2021-22118, CVE-2020-5421, CVE-2022-22950, CVE-2022-22971, CVE-2022-22968, CVE-2022-22970, CVE-2021-22060, and CVE-2021-22096. I suggest upgrading to version 5.3.20 or newer. Spring Web has 10 vulnerabilities: CVE-2016-1000027, CVE-2022-22965, CVE-2021-22118, CVE-2020-5421, CVE-2022-22950, CVE-2022-22971, CVE-2022-22968, CVE-2022-22970, CVE-2021-22060, and CVE-2021-22096. I would suggest upgrading to version 5.3.20 or newer. Apache Tomcat has 20 vulnerabilities: CVE-2020-1938, [CVE-2020-8022](http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-8022), [CVE-2020-11996](http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2020-11996), CVE-2020-13934, CVE-2020-13935, CVE-2020-17527, CVE-2021-25122, CVE-2021-41079, CVE-2022-29885, CVE-2022-42252, CVE-2020-9484, CVE-2021-25329, CVE-2021-30640, CVE-2022-34305, CVE-2021-24122, CVE-2021-33037, CVE-2019-17569, CVE-2020-1935, CVE-2020-13943, and CVE-2021-43980. I suggest upgrading to version 10.1.0 or newer.

## Mitigation Plan

I would suggest that Artemis Financial first upgrades all the dependency vulnerabilities. I would also suggest that https functionality is added. I would suggest that standard coding practices would be implemented such as not hard coding values. I would also suggest adding in input validation.