****

# CS 305 Project One

**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** | **07.12.22** | **Jaime Rowland** |  |

## Client



## Instructions

Deliver this completed vulnerability assessment report, identifying your findings of security vulnerabilities and articulating recommendations for next steps to remedy the issues you have found.

Respond to the five steps outlined below and include your findings. Replace the bracketed text on all pages with your own words. If you choose to include images or supporting materials, be sure to insert them throughout.

## Developer

Jaime Rowland

## 1. Interpreting Client Needs

Determine your client’s needs and potential threats and attacks associated with their application and software security requirements. Consider the following regarding how companies protect against external threats based on the scenario information:

* What is the value of secure communications to the company?
* Are there any international transactions that the company produces?
* Are there governmental restrictions about secure communications to consider?
* What external threats might be present now and in the immediate future?
* What are the “modernization” requirements that must be considered, such as the role of open source libraries and evolving web application technologies?

Secure communication means the company is using a secure data communication method to receive and send between the client and the server. Using a secure communication method guarantees that there is breach in data. Since Artemis Financial creates financial plans for customers and handles the customer’s funds, AF has access to sensitive information that can be looked for by intruders on the outside. Secure communications should be a high priority in the company when it comes to the customer communication processes. Artemis Financial is a financial institution which means they must follow government regulations and restrictions when it comes to all transactions and communications. Any threats, current or potential, would be someone seeking of personal information of a client and/or financial information of both the client and the company. Attacks are a threat if the API isn’t secure enough. This can then cause information to leak. Two-factor authentication should be implemented to help stop any fake login attempts and all communication should be done through HTTPS because sensitive information will be going back and forth between company and customer.

## 2. Areas of Security

Referring to the Vulnerability Assessment Process Flow Diagram, identify which areas of security are applicable to Artemis Financial’s software application. Justify your reasoning for why each area is relevant to the software application.

Secure coding is important in order to create a structured code that will create a level of security to prevent any outside interference and also provide software to ensure company and client privilege. Code errors are relevant since errors should be handled in a secure manner by the company. APIs are significant since the web services uses RESTful API, which will need secure communication. Lastly, input validation is relevant because as the RESTful API takes in user input, the user input must be cleared and validated.

## 3. Manual Review

Continue working through the Vulnerability Assessment Process Flow Diagram. Identify all vulnerabilities in the code base by manually inspecting the code.

* Service does not use HTTPS, this is recommended when sharing sensitive information
* No authentication system in place for verification uses.
* Requests are not validated which in turn leaves the system vulnerable to outsiders
* Business names are sent as request parameters within the CRUD Controller class. This proves to be vulnerable by having leak able information to outsiders.

## 4. Static Testing

Run a dependency check on Artemis Financial’s software application to identify all security vulnerabilities in the code. Record the output from dependency check report. Include the following:

1. The names or vulnerability codes of the known vulnerabilities
2. A brief description and recommended solutions provided by the dependency check report
3. Attribution (if any) that documents how this vulnerability has been identified or documented previously

* bcprov-jdk15on-1.46.jar – several vulnerabilities on 1.46, update to latest version.
* 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-2018-5382
* Log4j-api-2.12.1.jar – one vulnerability, update to latest version.
* CVE-2020-9488
* Snakeyalm-1.25.jar – one vulnerability, update to latest version.
* CVE-2017-18640
* Jackson-databind-2.10.2.jar – one vulnerability, update to latest version.
* CVE-2020-25649
* Tomcat-embed-core-9.0.30.jar – several vulnerabilities, update to latest tomcat version.
* 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
* Hibernate-validator-6.0.18.Final.jar – one vulnerability, update to latest version.
* CVE-202-10693
* Spring-core-5.2.3.RELEASE.jar – one vulnerability, update to latest version.
* CVE-2020-5421

## 5. Mitigation Plan

After interpreting your results from the manual review and static testing, identify the steps to remedy the identified security vulnerabilities for Artemis Financial’s software application.

To address all future issue and any present ones, it is recommended that first we need to ensure company and client information is secure. To make sure the information is secure we need to switch to HTTPS for all communication which will prevent outsiders from prying. Second, we would move request parameters to headers. Third, we would remove any reference of business names within hard-coded database credentials. Fourth we want to implement and enable two-factor authentication systems which will help to protect user information. Lastly, we will update all the dependencies from the above dependency check.