****

# 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** | **11/14/21** | **Talha Shafiq** |  |

## 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

Talha Shafiq

## 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?

Artemis Financial deals with funds and makes a financial plan for clients, so they conceal data that is very sensitive that could be desired by intruders. Because of this, the worth of safe communication to the business must be high and a prime concern. Considering that Artemis Financial is a financial consulting company, they need to obey the rules and limitations of the government when dealing with transactions to avoid any security threats between the customer and business. Asking for personal information is considered a security risk and if the API is not fully protected, it may lead to attacks and also result in information leakage. Multi-factor authentication must be executed in order to avoid wrong log in attempts. Furthermore, any communication needs to be done through HTTPS since sensitive data is exchanged back and forth utilizing response and request headers.

## 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.

An area of security that is relevant to the application is secure coding. It assists in developing structured code as well as ensuring safety of any intruders. Code error is also applicable because errors like several log in attempts need to be managed by the company in a secured way. APIs are applicable because Artemis Financial utilizes a RESTful web API. Finally, input validation is applicable since as the RESTful API receives user input, the input needs to be cleared and verified in a secured way.

## 3. Manual Review

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

* HTTPS is not utilized, which is suggested when sharing sensitive data.
* Authentication system not set up for validation purposes.
* Requests aren’t verified which can make the system unprotected
* In the CRUDController class, business names are being sent as request parameters. This can also make the system unprotected and result in information leakage.

## 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.

The first step is to make sure that the user/client and company’s data is protected by utilizing HTTPS for all communication in order to avoid any intruders. The second step is to shift request parameters to headers. The third step is to get rid of any mention of business names in embedded credentials. The fourth step is enabling and using multi-factor authentication or other common authentication systems in order to secure the identity and data of the business and customer. Lastly, updating each dependency from the dependency check report above.