****

# 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** | **20/03/2022** | **Nahom Mekonen** |  |

## 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

Nahom Mekonen

## 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?

As communication being vital. Secure communication is important to protect the client data, protect their intellectual property from any kind of threats.

* Are there any international transactions that the company produces?

Through the language, cultural barriers, currency exchange rates, foreign politics and policies there are many type of international transactions including specialized skills, international legal advices, Strategic counseling in risk assessment, development of business and licensing models, and protection of intellectual property and technology assets and much more.

* Are there governmental restrictions about secure communications to consider?

It depends on the government’s policies, but there are several restriction put by the government that should be considered for secure communications.

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

Malware, phishing, ransomware, Distributed Denial of Service (DDoS) attacks involve an attacker flooding a system, Network vulnerabilities, Data loss and End users.

* What are the “modernization” requirements that must be considered, such as the role of open source libraries and evolving web application technologies?

Perform an analysis of the range and diversity of resources needed for a successful outcome, conduct a comparative study on developing new capabilities, considering prior investments, recognizing there will be future change, not promulgating vendor-specific API’s, using micro-services instead of “big bang” approach and if it supports multiple API versions concurrently.

Artemis Financial is a company that devises financial plans for individuals ranging from savings, and retirements to investments and insurance. The value of secure communications is absolutely vital, considering the company is handling classified information from clients that may include SSN, tax information, etc. With this mind, there were no references that Artemis Financial was only based in the US, it would be safe to assume that this company would have to deal with international transactions. Theone major governmental restriction the company would have to consider about secure communications would be ensuring that there isn't exposure to information regarding trade secrets. Because of the severe need to protect all types of information, the main external threat is targeting any client's information, so much of the data will have to be heavily encrypted from anyone outside the company. Having up-to-date maintenance checks regarding bug fixes and weaken security threads would be a modernized requirements that Artemis Financial should consider.

## 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.

* Input Validation -- Artemis Financial requires input validation in order to validate owner of information. This would provide protection for users. The input validation would be written as strings.
* Code Quality, within code quality, you are able to take control of access of methods based on the user. For example, a user would only have access to their information, and not access to both another user's information or even access to the server, etc.
* APIs, The creation of an API would be necessary because it is running internally and externally. It would allow for which data accessed is acceptable.
* Code Error, implementation of error handling creates an understanding of what areas of the API would need to be fixed. This way Artemis Financial wouldn't have to worry about their user information being exposed or accessed.
* Cryptography, implementing cryptography within Artemis Financial would be crucial, because it would ensure that user information wouldn't be compromised from different areas of the world considering there would be different currencies being involved.

[Input Validation -- Artemis Financial requires input validation in order to validate owner

of information. This would provide protection for users. The input validation would be

written as strings.

•Code Quality, Within code quality, you are able to take control of access of methods

based on the user. For example, a user would only have access to their information, and

not access to both another user's information or even access to the server, etc.

•APIs, The creation of an API would be necessary because it is running internally and

externally. It would allow for which data accessed is acceptable.

•Code Error, implementation of error handling creates an understanding of what areas of

the API would need to be fixed. This way Artemis Financial wouldn't have to worry about

their user information being exposed or accessed.

•Cryptography, implementing cryptography within Artemis Financial would be crucial,

because it would ensure that user infromation wouldn't be compromised from different

areas of the world considering there would be different currencies being involved.

## 3. Manual Review

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

From reviewing the code, and applying the Vulnerability Assessment, I analyzed the both

POM.XML and the Greeting Controller. From the XML file, I wanted to verify if I could find an

Apache Validator. From the greeting Controller, I noticed there was a lack of input validation,

which would be something I would take note for the feature. Code quality was acceptable;

however, there was an issue with the error handling because it wasn't there. Moving on to the

API. From the API, I noticed that it lacked in a lot of things. There was a breach that could expose

the user input, because it wasn't written through a POST method. Lastly, I tried verifying signs of

cryptography; however, I was unable to find it.

From reviewing the code, and applying the Vulnerability Assessment, I analyzed the both POM.XML and the Greeting Controller. From the XML file, I wanted to verify if I could find an Apache Validator. From the greeting Controller, I noticed there was a lack of input validation, which would be something I would take note for the feature. Code quality was acceptable; however, there was an issue with the error handling because it wasn't there. Moving on to the API. From the API, I noticed that it lacked in a lot of things. There was a breach that could expose the user input, because it wasn't written through a POST method. Lastly, I tried verifying signs of cryptography; however, I was unable to find it.

## 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

intercepted by a man-in-the-

middle attack which could leak any

log messages sent through that

appender.

true to globally enable hostname

verification for SMTPS connections.

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:a

pache\_tomcat:9.0.30:\*:\*:\*

:\*:\*:\*:\*

Apache Tomcat 10.0.0-M1 to

10.0.6, 9.0.0.M1 to 9.0.46 and

8.5.0 to 8.5.66 did not correctly

parse the HTTP transfer-encoding

request header in some

circumstances leading to the

possibility to request smuggling

when used with a reverse proxy.

Specifically: - Tomcat incorrectly

ignored the transfer encoding

header if the client declared it

would only accept an HTTP/1.0

response; - Tomcat honoured the

identify encoding; and - Tomcat

did not ensure that, if present, the

chunked encoding was the final

encoding.

- Upgrade to Apache Tomcat 10.0.6

or later

|  |  |  |  |
| --- | --- | --- | --- |
| Dependency | Vulnerability | Description | Solution |
| log4j-api-2.12.1.jar | cpe:2.3:a:apache:log4j:2.12.1 | Improper validation of certificate with host mismatch in Apache Log4j SMTP appender. This could allow an SMTPS connection to be Intercepted by a man-in-the-middle attack which could leak anylog messages sent through that appender. | Upgrade to 2.13.2 which supports this feature. Previous versions can set the system property mail.smtp.ssl.checkserveridentity to true to globally enable hostname verification for SMTPS connections |
| 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:\*:\*:\*:\*:\*:\*:\* | Apache Tomcat 10.0.0-M1 to 10.0.6, 9.0.0.M1 to 9.0.46 and 8.5.0 to 8.5.66 did not correctly parse the HTTP transfer-encoding request header in some circumstances leading to the possibility to request smuggling when used with a reverse proxy. Specifically: - Tomcat incorrectly ignored the transfer encoding header if the client declared it would only accept an HTTP/1.0 response; - Tomcat honoured the identify encoding; and - Tomcat did not ensure that, if present, the chunked encoding was the final encoding | Upgrade to Apache Tomcat 10.0.6 or later |

## 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 Financials software application.

The quick remedy to the identified security vulnerabilities for Artemis Financial would be simply running the current versions of Snakeyaml, hibernator validator, Apache Tomcat, and bouncycastle.