****

# 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** | **11/13/2021** | **Bijal Chauhan** | **Vulnerability Assessment** |

## 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

Bijal Chauhan

## 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?

All the company resources are very important. When it comes to communication, all the resources are exposed, so it is very important to secure all resources to avoid any attacks. Securing communication will help company to secure data between employees and client. The main goal for Artemis Financial is to secure the organization from any exterior attacks.

* Are there any international transactions that the company produces?

As per the description, the company offers many different financial plans and deals with multinational transactions. Since many customers access their finances or access their accounts from out of the country, the transactions will include internet usage.

* Are there governmental restrictions about secure communications to consider?

From the Artemis Financial’s point of view, I don’t think there is any governmental restrictions to secure communications. The law provides 100% consent to encrypt the communication as long as it does not violate any of the client’s rights. The main concern in these scenarios is to protect the client’s data and finances. Hence, Artemis Financial is free to encrypt communication.

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

External Threats that might be present now include:

* Authentication threat where hackers might exploit the vulnerabilities and gain access to the system because the company’s core application is web based and client needs to log in the system to access their portals.
* Communication threat where the group of people such as man in the middle attacks the communication between organization and client. Through these attacks, they can expose the company’s sensitive information.
* Denial of service threat where the intruders throw lot of traffic to webservers and overload it which eventually crashes the application.
* What are the “modernization” requirements that must be considered, such as the role of open-source libraries and evolving web application technologies?

The incremental method is used to update the system without affecting any of its operations. This method is applied just by replacing the old code with the new modernized code using the open-source code technology. Open-source code increases simpler changes to the code with the help of several coders with a variety of expertise, who can carry down their proficiency in securing the application.

## 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 has a web-based application where various clients create their own account to access their financial transactions. Therefore, input validation must be the priority here to ensure that clients filled the data to meet the company’s policy.
* API’s: A web-based application utilizes API’s system, and it is effective system giving fast results compared to traditional applications because traditional applications were heavily loaded from server-side. In this modern era where everyone is mobile users, companies should come up with the mobile based application that can be easier to access the services whenever and wherever needed. With the luxury, we should not forget the security concerns. Almost all operations done on client’s device that will expose the application with vulnerabilities that hackers can take advantage of because the device security is under client side rather than company. It is important to assess the vulnerabilities presented by APIs because through APIs can execute the sensitive communication as well as denial of any service attacks. This will help in determining the proper mitigation plan in early phase.
* Cryptography: The assessment will help to evaluate the application performance based on secure communication. The test is done to know any vulnerabilities in the system and to implement it to ensure secure communication without leaving any gaps for attacks.
* Code Quality: The code used for the system needs to be tested for overall security and operational errors because it is important to have secure code to run the application successfully in terms of security concerns. Developers do their best while developing the code, but occasionally forgets that implementing simple security features will be exposed to hackers to hack the application. Therefore, to prevent this hacking, it is recommended to assess all the executables that have been determined. All these will lock the doors for hackers to hack the system.
* Encapsulation: Data must be encapsulated properly to avoid any leaks from the system. Failed encapsulation will lead to system attacks, which will give access to hackers to access the application’s sensitive data.

## 3. Manual Review

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

* Data Exposure: This vulnerability is found in Artemis Financial’s DocData class in which the method is used to define the database location name, user, and password. The code given uses a core username and password that anyone can guess easily. Such programming method can easily attract outsiders to access the application using simple password techniques. If this vulnerability lefts untreated, there will be high chances of attacks on the applicated. The code example is attached below:

Graphical user interface, text, application

Description automatically generated

* Insecure Direct Object References: This vulnerability was found in the CRUDController class where an SQL injection attack can easily take advantage of. The code given references directly to the database and all the sensitive data can be easily exposed. The code example is as follows:

Graphical user interface, text, application, email

Description automatically generated

## 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

Dependency Check Report:

Graphical user interface, text, application

Description automatically generated

Vulnerabilities as per dependency check report:

|  |  |  |
| --- | --- | --- |
| Vulnerability Code | Description | Mitigation plan |
| CVE-2015-6644 | This vulnerability is found in Bouncy Castle that could enable a local malicious application to gain access to user’s confidential information. | To prevent security concerns due to this vulnerability, we need to update the application and the operating system on the regular basis. |
| CVE-2020-13935 | This vulnerability related to the denial-of-service threat. The payload length in a WebSocket frame have validation issue that triggers an infinite loop for Dos attacks in Apache Tomcat. | To prevent this vulnerability, we need to ensure about proper validation. |
| CVE-2013-1624 | This vulnerability shows the cryptographic issues where TLS implemented in the Java library does not considerate the timing side-channel attacks. This allows remote attacker to utilize the plaintext-recovery attacks with the help of statistical analysis of timing data. | To prevent this vulnerability, one should use proper cryptographic library and its helping functions. Also, appropriate timing checks should be performed to avoid any timing data attacks. |
| CVE-2020-9488 | This vulnerability relates to improper validation of certificate in Apache Log4j. Therefore, there is high chances man in middle attacks. | Upgrade the system to 2.13.2 that supports this feature and give benefits in regard to security. |

## 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.

* Upgrade Login Information: The application should be upgraded with the strong password method. There should be mandatory things set to the password like specific length of character with alphabets, number, and a special character. The default username and password should be changed for better secure the data.
* Use Proper Validation: There will be many different cryptographic certificates that should be validated into the code. The code will be cleaned to allow appropriate validation of the digital certificate in the application.
* Code Quality Review: The code should be reviewed to ensure secure coding method. The proper programming skills will help enhance the quality of the code and prevent any error that led to application attacks.
* Update an Apache Server: upgrading the server to the newer version helps preventing threats so it is important to apply this upgrade for security purpose.