****

# 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** | **January 24, 2021** | **Jacob Lee** | **First draft** |

## Client



## Developer

Jacob Lee

## 1. Interpreting Client Needs

The value of secure communications is that data will be protected in a system with secure processes. Without secure communications, data can leak outside the system and could be misused by unauthorized users to cause harm or steal property. Artemis Financial produces international transactions for any customers who are living in another country. The prompt did not specify where, Artemis Financial is located, but it will be important to have the capability to cater to international customers in order to grow business. There are no governmental restrictions about secure communications since Artemis does not have governmental contracts, but it will still be wise to ensure communications are secure to prevent data leaks. External threats that might be present now and, in the future, would be malicious hackers attempting to steal data and financial information about Artemis Financial’s clients. The recent trend is for programs to use more open source code, but it will be important that any open source code that is used does not introduce security vulnerabilities that can be exploited.

## 2. Areas of Security

* Input validation- Whenever taking input, it is important to validate the input and make sure it does not cause errors or malfunctions in the code. The Greeting Controller takes input and it should be validated.
* APIs- Secure API interactions are needed because Artemis Financial is using RESTful API.
* Cryptography – Encryption use will make the storage of sensitive data more secure. Financial data is very sensitive and vulnerable for exploitation, so the data should be encrypted.
* Client/Server – It will be important for data to be distributed securely and only to proper requests. Data should only be shared with users who have the authorization to view it. For example, customers should only be able to view their own data and not be able to access other clients’ private data.
* Code Error- Errors need to be handled in a way that doesn’t break the system, but just reports the errors in a secure manner.
* Encapsulation- the data structures need to be secure when storing and accessing data in the system.

## 3. Manual Review

Input Validation- GreetingController has a name input that should be validated to prevent unexpected behaviors.

APIs- myDateTime retrieves Data for date and time using an API. DocData also reads a mySQL query. These need to be secure.

Encryption- customer.java retrieves the customer account number and account balance. This data should be encrypted.

## 

## 4. Static Testing

**cpe:/:bouncycastle:legion-of-the-bouncy-castle-java-crytography-api:1.46**

- CVE-2018-1000613 - Use of Externally-Controlled Input to Select Classes or Code ('Unsafe Reflection') vulnerability in XMSS/XMSS^MT private key deserialization that can result in Deserializing an XMSS/XMSS^MT private key can result in the execution of unexpected code. This attack appear to be exploitable via A handcrafted private key can include references to unexpected classes which will be picked up from the class path for the executing application. This vulnerability appears to have been fixed in 1.60 and later.

**cpe:/:fasterxml:jackson-databind:2.10.2**

- CVE-2020-25649 - A flaw was found in FasterXML Jackson Databind, where it did not have entity expansion secured properly. This flaw allows vulnerability to XML external entity (XXE) attacks. The highest threat from this vulnerability is data integrity.

**cpe:/:apache:log4j:2.12.1**

-CVE-2020-9488 - 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 any log messages sent through that appender.

**cpe:/:snakeyaml\_project:snakeyaml:1.25**

- CVE-2017-18640 - The Alias feature in SnakeYAML 1.18 allows entity expansion during a load operation

**cpe:/:pivotal\_software:spring\_framework:5.2.3**

- CVE-2020-5421 - In Spring Framework versions 5.2.0 - 5.2.8, 5.1.0 - 5.1.17, 5.0.0 - 5.0.18, 4.3.0 - 4.3.28, and older unsupported versions, the protections against RFD attacks from CVE-2015-5211 may be bypassed depending on the browser used through the use of a jsessionid path parameter.

**cpe:/:apache:tomcat:9.0.30**

CVE-2020-17527 - While investigating bug 64830 it was discovered that Apache Tomcat 10.0.0-M1 to 10.0.0-M9, 9.0.0-M1 to 9.0.39 and 8.5.0 to 8.5.59 could re-use an HTTP request header value from the previous stream received on an HTTP/2 connection for the request associated with the subsequent stream. While this would most likely lead to an error and the closure of the HTTP/2 connection, it is possible that information could leak between requests.

## 5. Mitigation Plan

One of the most important steps to take for security would be to update all dependencies to the latest version. Another step that can be taken to increase security include input validation to make sure the name input to the Greeting does not cause unexpected behavior. Finally, the account data should be encrypted for additional security in case of data leak.