****

# 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** |
| --- | --- | --- | --- |
| **1.1** | **7-20-2023** | **Tatiana Case** | **First Draft** |

## Client



## Instructions

Submit this completed vulnerability assessment report. Replace the bracketed text with the relevant information. In the report, identify your findings of security vulnerabilities and provide recommendations for the next steps to remedy the issues you have found.

* Respond to the five steps outlined below and include your findings.
* Respond using your own words. You may also choose to include images or supporting materials. If you include them, make certain to insert them in all the relevant locations in the document.
* Refer to the Project One Guidelines and Rubric for more detailed instructions about each section of the template.

## Developer

Tatiana Case

## Interpreting Client Needs

Artemis Financial provides customized financial services to their customers. Because of this, it can be assumed that the information they handle, and transfer may contain sensitive customer information. Sensitive information is almost always the target of malicious attacks. Artemis Financial can already perform transactions and communicate internationally. This will need to be considered when determining the potential threats and the steps that will need to be taken to protect against those threats. Artemis Financial must follow the established government policies where its headquarters is located and may need to adjust for international customers to follow their local ordinances as well. The company is asking for our assistance to protect their vulnerabilities and threats.

## Areas of Security

Having a secure environment for the customers is a requirement. The API will need to be secure so when individuals are accessing the system, the commands are not altered leading to a potential hack. Code quality and accuracy plays a large role in the security of the system. If the program is not up to date, then vulnerabilities may be exploited and a potentially malicious event to occur. Since the program is being used to interact with and provide a service to a patron, the server must be secured. Data that is being sent over the network must be encrypted and tested once they are received to aid in preventing any breach that may have occurred.

## Manual Review

There currently is no validation method in place within the coding for the program. This could lead to security threats and data breeches. When sending out requests within the system, these requests must be validated to prevent unwanted access. HTTPS should be used as a secure communication protocol.

## Static Testing

* Bcprov-jdk15on-1.46-jar
  + Update the version to the latest on available.
    - CVE-2013-1624
    - CVE-2016-6644
    - CVE-2015-7940
    - CVE-2016-100338
    - CVE-2016-100339
    - CVE-2016-100341
    - CVE-2016-100342
    - CVE-2016-100343
    - CVE-2016-100344
    - CVE-2016-100345
    - CVE-2016-100346
    - CVE-2016-100352
    - CVE-2017-13098
    - CVE-2018-1000613
    - CVE-2018-5382
* Log4j-api-2.12.1.jar
  + Update to the latest version available.
    - CVE-2020-9488
* Snakeyalm-1.25.jar
  + Update to the latest version available.
    - CVE-2017-08640
* Jackson-databind-2.10.2.jar
  + Update to the latest version available.
    - CVE-2020-25649
* Tomcat-embeded-core-9.0.30.jar
  + Update to the latest version available.
    - CVE-2019-17569
    - CVE-2020-11996
    - CVE-2020-13934
    - CVE-2020-13935
    - CVE-2020-17527
    - CVE-2020-1935
    - CVE-2020-1938
    - CVE-2020-8022
    - CVE-2021-24122
* Hibernate-validator-6.0.18.final.jar
  + Update to the latest version available.
    - CVE-2020-10693
* Spring-core-5.2.3release.jar
  + Update to the latest version available.
    - CVE-2020-5421

## Mitigation Plan

Files that were detected during the dependency test must be updated. Outdated software does not contain crucial security updates and eliminates flaws discovered in the older versions. Using the

RESTful APIs and using HTTPS as a communication protocol will protect data as it is moved about the program. Creating and implementing a better two-factor/validation method for users will also ensure that only authorized users are gaining access to certain accounts.