****

# 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** | **11/18/2022** | **Robert Sanford** | **N/A** |

## 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

Robert Sanford

## Interpreting Client Needs

Secure communications are extremely valuable to a company such as Artemis Financial for many reasons. Because they develop “individualized financial plans” for their customers, there is likely a large amount of sensitive information within their systems. This may include banking and investment information, personal details like address, SSN, and DOB’s. Having possession of this kind of information can make the company a huge target for hackers and cyber criminals looking to exploit that information. Therefore, security is not only valuable, but also critical to the company’s success and survival. There are certain regulations that all financial institutions must follow regarding keeping sensitive data safe under the Gramm-Leach-Bliley Act. This act ensures that organizations, such as Artemis Financial, “must ensure the security and confidentiality of customer information”. They must also explain how they keep that data safe as well. Phishing, malware, ransomware, and SQL injection attacks are all examples of potential external threats that must be taken seriously both now and in the future. Modernization requirements include updating the platform that the system is running on and updating the code and design from open-source libraries. Keeping everything up to date is very important.

## Areas of Security

* Input Validation – This area is relevant because when the RESTful API accepts user input, it must be verified and cleared in a secure way.
* Secure Coding – This area is relevant because secure code is needed because it will help with having a high level of security and prevent penetration outside of the system.
* Code Errors – This area is relevant since errors, such as multiple login attempts, need to be addressed to ensure that someone is not trying to hack into the system.
* APIs – This area is also required because this web-based software uses a RESTful API which, being a financial institution, requires a high level of security.

## Manual Review

* This service does not use HTTPS which is highly recommended for security when sharing sensitive information.
* Requests do not appear to be validated, which may leave the system vulnerable to outsiders.
* There does not appear to be an authentication system for verifications.
* Names are sent as request parameters in CRUDController.java, this can be a vulnerability as there is possibility of leaking information to unauthorized people.

## Static Testing

* bcprov-jdk15on-1.46.jar – there are several vulnerabilities on 1.46, Updating to the latest version is ideal.
  + CVE-2016-1000338
  + CVE-2016-1000339
  + CVE-2016-1000342
  + CVE-2016-1000343
  + CVE-2016-1000344
  + CVE-2016-1000345
  + CVE-2016-1000346
  + CVE-2016-1000352
  + CVE-2017-13098
  + CVE-2015-7940
  + CVE-2018-5382
* Hibernate-validator-6.0.18.Final.jar – one vulnerability, update to the latest version.
  + CVE-2020-10693
* Jackson-databind-2.10.2.jar – several vulnerabilities, update to latest versions.
  + CVE-2020-25649
  + CVE-2020-36518
  + CVE-2022-42003
  + CVE-2022-42004
* Log4j-api-2.12.1.jar – one vulnerability, update to latest version
  + CVE-2020-9488
* Logback-core-1.2.3.jar One vulnerability, update to latest version
  + CVE2021042550
* Snakeyalm-1.25.jar – several vulnerabilities, update to latest version.
  + CVE-2017-18640
  + CVE-2022-38751
  + CVE-2022-38752
  + CVE-2022-38750
* Spring-core-5.2.3.RELEASE.jar several vulnerabilities, update to latest version
  + CVE-2022-22965
  + CVE-2021-22118
  + CVE-2020-5421
  + CVE-2022-22971
  + CVE-2022-22968
* Tomcat-embed-core-9.0.30.jar – several vulnerabilities, update to latest tomcat version
  + CVE-2020-1938
  + CVE-2020-11996
  + CVE-2020-13934
  + CVE-2020-13935
  + CVE-2020-17527
  + CVE-2021-25122
  + CVE-2021-41079
  + CVE-2022-29885
  + CVE-2022-42252
  + CVE-2020-9484

## Mitigation Plan

The first step that we should take in order address the current and future security issues is to ensure customer information is secure by implementing HTTPS for all communications. Second, we should also implement two-factor authentication for all users logging into the system, as this will help prevent any unwanted person(s) from entering the system. Third, we should update all dependencies listed in the dependency check above. Fourth, we should train all staff members on how to spot phishing e-mails and set up an e-mail address they can forward any suspicious e-mails to. Lastly, we should also consider moving request parameters to headers and removing references to business names within the database.

References:

Privacy Act Issues under Gramm-Leach-Bliley

https://www.fdic.gov/consumers/consumer/alerts/glba.html