****

# 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** | **01/22/2023** | **LaNise Essick** |  |

## 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

LaNise Essick

## Interpreting Client Needs

Artemis Financial offers financial plans to its customers. Due to the high level of sensitivity associated with this financial information, all communications between the company and its clients must be secure. As part of its financial transactions and communications, Artemis Financial will also need to comply with government regulations, which will affect data retention policies and security requirements. If requests and responses are not structured securely, RESTful APIs such as Artemis Financials’ are vulnerable to data interception. The service must ensure that HTTPS is used for all communications and that confidential information is included in request and response headers or in request and response bodies. Additionally, a secure authentication scheme such as OAuth should be used.

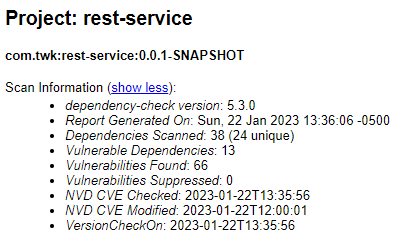
## Areas of Security

* Input validation – because the RESTful API will accept user input, this input will have to be validated.
* APIs – the web service includes a RESTful API which will need to communicate securely.
* Code Error: we will have to review all command inputs and any code used to access APIs to ensure there are no vulnerabilities.

## Manual Review

* Business names are sent as request parameters in the CRUDController class.
* Database connection parameters are hard-coded in DocData.
* Request parameters are not validated.
* Service does not use HTTPS.
* No authentication scheme present.

## Static Testing :



* bcprov-jdk15on-1.46.jar: bouncycastle:legion-of-the-bouncy-castle-java-crytography-api

CVE-2016-1000352

CVE-2016-1000346

CVE-2016-1000345

CVE-2016-1000344

CVE-2016-1000343

CVE-2016-1000342

CVE-2016-1000341

CVE-2016-1000339

CVE-2016-1000338

CVE-2018-5382

CVE-2017-13098

CVE-2013-1624

* spring-boot-2.2.4.RELEASE.jar

     CVE-2022-27772

* logback-core-1.2.3.jar

CVE-2021-42550

* log4j-api-2.12.1.jar

     CVE-2021-42550

* snakeyaml-1.25.jar

   CVE-2022-3064

CVE-2021-4235

* jackson-databind-2.10.2.jar

CVE-2022-42004

CVE-2022-42003

CVE-2020-36518

CVE-2020-25649

* tomcat-embed-core-9.0.30.jar

CVE-2022-42252

CVE-2021-43980

CVE-2022-34305

CVE-2022-29885

CVE-2021-41079

CVE-2021-33037

CVE-2021-30640

CVE-2021-25329

CVE-2021-25122

CVE-2021-24122

CVE-2020-17527

CVE-2020-13943

CVE-2020-13935

CVE-2020-13934

CVE-2020-8022

CVE-2020-11996

CVE-2020-9484

CVE-2020-1938

CVE-2020-1935

CVE-2019-17569

* hibernate-validator-6.0.18.Final.jar

CVE-2020-10693

For al vulnerabilities I recommend updating the packages to the latest version

## Mitigation Plan

* Switch to HTTPS protocol for all communications
* Move request parameters to headers or body rather than URI
* Remove hard-coded database connection credentials
* Implement secure authentication scheme
* Update dependencies as listed above