****

# 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** | **9/11/2023** | **Nicholas Kreuziger** | **Conducted Client Assessment, Manual Code Review, Static Testing and Mitigation recommendations.** |

## 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

Nicholas Kreuziger

## Interpreting Client Needs

Artemis Financial is a consulting company that develops individualized financial plans (savings, retirement, investments, insurance). The brevity of customers and financial plans they service creates a client relationship that demands trust. Communications security failures, no matter how big or small, would challenge the trust of their customers and greatly harm that trust.

Artemis Financial utilizes a RESTful web Application Programming interface (API). They want to ensure the API is operating securely and in accordance with Information Security Standards. We are being contracted to identify and evaluate risks to customer information in addition to providing a plan to mitigate those risks (The Fed - Interagency Guidelines Establishing Information Security Standards, n.d.). As a financial institution they will continue to be the target of external threats, educating their employees to use their systems securely and keeping their use of libraries and web-applications updated is key.

## Areas of Security

There are a few areas of security that should be considered from the first level of the VAPF (Vulnerability Assessment Process Flow).

1. **Security Area:** Input Validation : Secure Input and Representations
   1. **Justification:** Artemis Financial is consulting with a wide variety of customers about financial plans. Malicious actors can use any point of input to breach the system and steal client information.
2. **Security Area:** APIs : Secure API Interactions
   1. **Justification:** The application is likely to interact with other services through APIs. Secure interactions are essential , especially when passing any data obtained from a command input to another service or API.
3. **Security Area:** Code Error : Secure Error Handling
   1. **Justification:** Ensuring that any errors in user input or logical processing are handled in a manner that doesn’t result in a breach is important. Any processing of external data or processes is a risk.
4. **Security Area:** Code Quality : Secure Coding Practices / Patterns
   1. **Justification:** Since we are using a framework, ensuring educated use of that framework and its known vulnerabilities is important. Aligning code with best practices will be key.

## Manual Review

* File: DocData.java
  + Vulnerability
    - The class DocData retrieves documents from a database and accepts inputs from the user , must ensure read method doesn’t expose itself to SQL injection.
    - Database credentials are in the code, instead consider storing environment variables or using external configuration files.
    - Database connection is not closed, this is an unnecessary memory leak risk.
  + Screenshot:

A screen shot of a computer program

Description automatically generated

* File: customer.java
  + Vulnerability
    - The account\_balance variable should be private to avoid unauthorized access or modification.
    - Input validation will have to be incorporated to avoid negative deposits.
    - Exception handling is also not incorporated to handle runtime errors.
    - A setter method would allow controlling how the variables are altered.
    - We should ask the client if the showInfo method is required to show the account number, less information exposure is better.
  + Screenshot: A screen shot of a computer program

    Description automatically generated
* File: CRUDController.java
  + Vulnerability
    - There is no input validation of the business name.
    - Error handling improvements to give errors clear guidance without exposing internal details would be beneficial.
  + Screenshot: A computer screen shot of a program

    Description automatically generated
* File: CRUD.java
  + Vulnerability
    - Two attributes seem to be storing the same content. Consider removing one.
    - Determine if the data exposure with getcontent and getcontent2 is necessary, or pare down how much data is exposed.
  + Screenshot:

A screen shot of a computer program

Description automatically generated

* File: Greeting.java
  + Vulnerability
    - Input Validation is not implemented, implement to avoid malicious input.
  + Screenshot:

A screen shot of a computer program

Description automatically generated

* File: myDateTime.java
  + Vulnerability
    - Input validation on setMyDateTime needs to be implemented to avoid invalid date/time settings.
  + Screenshot: A screen shot of a computer program

    Description automatically generated

## Static Testing



| **Dependency** | **Vulnerability IDs** | **Description** |
| --- | --- | --- |
| tomcat-embed-websocket-9.0.30.jar | [cpe:2.3:a:apache:tomcat:9.0.30:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aapache&cpe_product=cpe%3A%2F%3Aapache%3Atomcat&cpe_version=cpe%3A%2F%3Aapache%3Atomcat%3A9.0.30) [cpe:2.3:a:apache\_tomcat:apache\_tomcat:9.0.30:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aapache_tomcat&cpe_product=cpe%3A%2F%3Aapache_tomcat%3Aapache_tomcat&cpe_version=cpe%3A%2F%3Aapache_tomcat%3Aapache_tomcat%3A9.0.30) | Incoming connection security allows for the possibility of remote code execution or returning files. |
| tomcat-embed-core-9.0.30.jar | [cpe:2.3:a:apache:tomcat:9.0.30:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aapache&cpe_product=cpe%3A%2F%3Aapache%3Atomcat&cpe_version=cpe%3A%2F%3Aapache%3Atomcat%3A9.0.30) [cpe:2.3:a:apache\_tomcat:apache\_tomcat:9.0.30:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aapache_tomcat&cpe_product=cpe%3A%2F%3Aapache_tomcat%3Aapache_tomcat&cpe_version=cpe%3A%2F%3Aapache_tomcat%3Aapache_tomcat%3A9.0.30) | Incoming connection security allows for the possibility of remote code execution or returning files. |
| spring-webmvc-5.2.3.RELEASE.jar | [cpe:2.3:a:pivotal\_software:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Apivotal_software&cpe_product=cpe%3A%2F%3Apivotal_software%3Aspring_framework&cpe_version=cpe%3A%2F%3Apivotal_software%3Aspring_framework%3A5.2.3) [cpe:2.3:a:springsource:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aspringsource&cpe_product=cpe%3A%2F%3Aspringsource%3Aspring_framework&cpe_version=cpe%3A%2F%3Aspringsource%3Aspring_framework%3A5.2.3) [cpe:2.3:a:vmware:spring:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring&cpe_version=cpe%3A%2F%3Avmware%3Aspring%3A5.2.3) [cpe:2.3:a:vmware:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring_framework&cpe_version=cpe%3A%2F%3Avmware%3Aspring_framework%3A5.2.3) [cpe:2.3:a:web\_project:web:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aweb_project&cpe_product=cpe%3A%2F%3Aweb_project%3Aweb&cpe_version=cpe%3A%2F%3Aweb_project%3Aweb%3A5.2.3) | Priviledge Escalation is possible for a verified user within the application by recreating temporary storage directory. Denial of Service Attacks are possible as well. |
| spring-web-5.2.3.RELEASE.jar | [cpe:2.3:a:pivotal\_software:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Apivotal_software&cpe_product=cpe%3A%2F%3Apivotal_software%3Aspring_framework&cpe_version=cpe%3A%2F%3Apivotal_software%3Aspring_framework%3A5.2.3) [cpe:2.3:a:springsource:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aspringsource&cpe_product=cpe%3A%2F%3Aspringsource%3Aspring_framework&cpe_version=cpe%3A%2F%3Aspringsource%3Aspring_framework%3A5.2.3) [cpe:2.3:a:vmware:spring:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring&cpe_version=cpe%3A%2F%3Avmware%3Aspring%3A5.2.3) [cpe:2.3:a:vmware:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring_framework&cpe_version=cpe%3A%2F%3Avmware%3Aspring_framework%3A5.2.3) [cpe:2.3:a:web\_project:web:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aweb_project&cpe_product=cpe%3A%2F%3Aweb_project%3Aweb&cpe_version=cpe%3A%2F%3Aweb_project%3Aweb%3A5.2.3) | Priviledge Escalation is possible for a verified user within the application by recreating temporary storage directory. Denial of Service Attacks are possible as well. |
| spring-core-5.2.3.RELEASE.jar | [cpe:2.3:a:pivotal\_software:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Apivotal_software&cpe_product=cpe%3A%2F%3Apivotal_software%3Aspring_framework&cpe_version=cpe%3A%2F%3Apivotal_software%3Aspring_framework%3A5.2.3) [cpe:2.3:a:springsource:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aspringsource&cpe_product=cpe%3A%2F%3Aspringsource%3Aspring_framework&cpe_version=cpe%3A%2F%3Aspringsource%3Aspring_framework%3A5.2.3) [cpe:2.3:a:vmware:spring:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring&cpe_version=cpe%3A%2F%3Avmware%3Aspring%3A5.2.3) [cpe:2.3:a:vmware:spring\_framework:5.2.3:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring_framework&cpe_version=cpe%3A%2F%3Avmware%3Aspring_framework%3A5.2.3) | Priviledge Escalation is possible for a verified user within the application by recreating temporary storage directory. Denial of Service Attacks are possible as well. |
| spring-boot-starter-web-2.2.4.RELEASE.jar | [cpe:2.3:a:vmware:spring:2.2.4:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring&cpe_version=cpe%3A%2F%3Avmware%3Aspring%3A2.2.4) [cpe:2.3:a:vmware:spring\_boot:2.2.4:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring_boot&cpe_version=cpe%3A%2F%3Avmware%3Aspring_boot%3A2.2.4) [cpe:2.3:a:web\_project:web:2.2.4:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aweb_project&cpe_product=cpe%3A%2F%3Aweb_project%3Aweb&cpe_version=cpe%3A%2F%3Aweb_project%3Aweb%3A2.2.4) | Priviledge Escalation is possible for a verified user within the application by recreating temporary storage directory. Denial of Service Attacks are possible as well. |
| spring-boot-2.2.4.RELEASE.jar | [cpe:2.3:a:vmware:spring:2.2.4:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring&cpe_version=cpe%3A%2F%3Avmware%3Aspring%3A2.2.4) [cpe:2.3:a:vmware:spring\_boot:2.2.4:release:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Avmware&cpe_product=cpe%3A%2F%3Avmware%3Aspring_boot&cpe_version=cpe%3A%2F%3Avmware%3Aspring_boot%3A2.2.4) | Priviledge Escalation is possible for a verified user within the application by recreating temporary storage directory. Denial of Service Attacks are possible as well. |
| snakeyaml-1.25.jar | [cpe:2.3:a:snakeyaml\_project:snakeyaml:1.25:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Asnakeyaml_project&cpe_product=cpe%3A%2F%3Asnakeyaml_project%3Asnakeyaml&cpe_version=cpe%3A%2F%3Asnakeyaml_project%3Asnakeyaml%3A1.25) | According to GitHub appears to be a false positive. |
| jackson-databind-2.10.2.jar | [cpe:2.3:a:snakeyaml\_project:snakeyaml:1.25:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Asnakeyaml_project&cpe_product=cpe%3A%2F%3Asnakeyaml_project%3Asnakeyaml&cpe_version=cpe%3A%2F%3Asnakeyaml_project%3Asnakeyaml%3A1.25) | Denial of Service can occur with large depth of nested objects. |
| bcprov-jdk15on-1.46.jar | cpe:2.3:a:bouncycastle:bouncy-castle-crypto-package:1.46:\*:\*:\*:\*:\*:\*:\* cpe:2.3:a:bouncycastle:bouncy\_castle\_crypto\_package:1.46:\*:\*:\*:\*:\*:\*:\* [cpe:2.3:a:bouncycastle:legion-of-the-bouncy-castle-java-crytography-api:1.46:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Abouncycastle&cpe_product=cpe%3A%2F%3Abouncycastle%3Alegion-of-the-bouncy-castle-java-crytography-api&cpe_version=cpe%3A%2F%3Abouncycastle%3Alegion-of-the-bouncy-castle-java-crytography-api%3A1.46) cpe:2.3:a:bouncycastle:the\_bouncy\_castle\_crypto\_package\_for\_java:1.46:\*:\*:\*:\*:\*:\*:\* | Improper verification of cryptographic systems and cryptographic issues. |
| logback-core-1.2.3.jar | [cpe:2.3:a:qos:logback:1.2.3:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aqos&cpe_product=cpe%3A%2F%3Aqos%3Alogback&cpe_version=cpe%3A%2F%3Aqos%3Alogback%3A1.2.3) | An attacker with the required privileges to edit configurations files could craft a malicious configuration allowing to execute arbitrary code loaded from LDAP servers. |
| hibernate-validator-6.0.18.Final.jar | [cpe:2.3:a:redhat:hibernate\_validator:6.0.18:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aredhat&cpe_product=cpe%3A%2F%3Aredhat%3Ahibernate_validator&cpe_version=cpe%3A%2F%3Aredhat%3Ahibernate_validator%3A6.0.18) | Improper input validations due to invalid EL expressions expressed as valid. |
| log4j-api-2.12.1.jar | [cpe:2.3:a:apache:log4j:2.12.1:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Aapache&cpe_product=cpe%3A%2F%3Aapache%3Alog4j&cpe_version=cpe%3A%2F%3Aapache%3Alog4j%3A2.12.1) | Log messages can be leaked by a man-in-the-middle attack. |

## Mitigation Plan

| **Dependency** | **Mitigation Plan** |
| --- | --- |
| tomcat-embed-websocket-9.0.30.jar  tomcat-embed-core-9.0.30.jar | * Upgrade: Upgrade to the latest version of Apache Tomcat which has patches for known vulnerabilities. * Firewall: Ensure proper firewall rules are in place to block unauthorized access. * Monitoring: Set up monitoring tools to detect unusual patterns which could indicate an attack. |
| spring-webmvc-5.2.3.RELEASE.jar  spring-web-5.2.3.RELEASE.jar  spring-core-5.2.3.RELEASE.jar | * Upgrade: Upgrade to a version of Spring Framework that has addressed the known vulnerabilities. * Access Control: Implement stricter access control to prevent privilege escalation. * Input Validation: Enhance input validation to prevent users from recreating temporary storage directories. |
| spring-boot-starter-web-2.2.4.RELEASE.jar  spring-boot-2.2.4.RELEASE.jar | * Upgrade: Upgrade to a later version where the known vulnerabilities have been addressed. * Monitoring: Set up systems to monitor and prevent Denial of Service (DoS) attacks. |
| snakeyaml-1.25.jar | * Upgrade: Upgrade to a later version where the known vulnerabilities have been addressed. * Monitoring: Set up systems to monitor and prevent Denial of Service (DoS) attacks. |
| jackson-databind-2.10.2.jar | * Upgrade: Upgrade to a later version where the known vulnerabilities have been addressed. * Monitoring: Set up systems to monitor and prevent Denial of Service (DoS) attacks. |
| bcprov-jdk15on-1.46.jar | * Upgrade: Upgrade to a later version where the known vulnerabilities have been addressed. * Monitoring: Set up systems to monitor and prevent Denial of Service (DoS) attacks. |
| logback-core-1.2.3.jar | * Upgrade: Upgrade to a later version where the known vulnerabilities have been addressed. * Monitoring: Set up systems to monitor and prevent Denial of Service (DoS) attacks. |
| hibernate-validator-6.0.18.Final.jar | * Upgrade: Upgrade to a later version where the known vulnerabilities have been addressed. * Monitoring: Set up systems to monitor and prevent Denial of Service (DoS) attacks. |
| log4j-api-2.12.1.jar | * Upgrade: Upgrade to a later version where the known vulnerabilities have been addressed. * Monitoring: Set up systems to monitor and prevent Denial of Service (DoS) attacks. |

General measures that should also be taken are:

**Testing**: Regularly perform static and dynamic testing to identify and address new vulnerabilities.

**Training**: Provide training to developers on secure coding practices to prevent future vulnerabilities.

**Incident Response Plan**: Develop an incident response plan to address any security breaches promptly.

Citation

*The Fed - Interagency Guidelines Establishing Information Security Standards*. (n.d.). Board of Governors of the Federal Reserve System. Retrieved September 12, 2023, from https://www.federalreserve.gov/supervisionreg/interagencyguidelines.htm#:~:text=Under%20the%20Security%20Guidelines%2C%20each%20financial%20institution%20must%3A

‌

In-text citation: (*The Fed - Interagency Guidelines Establishing Information Security Standards*, n.d.)