# CS 305 Project One

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **07.21.2024** | **John Miller** |  |

## Client



## Developer

John Miller

**1. Interpreting Client Needs**

* **What is the value of secure communications to the company?**
  + Artemis Financial offers financial planning to a wide variety of clients. They provide services for investments, retirement, savings, and insurance. Thus, Artemis Financial collects and stores a vast amount of sensitive data that would be extremely harmful if compromised. Clients must be confident that their information is secure and can trust Artemis Financial with their financial futures.
* **Are there any international transactions that the company produces?**
  + The software is for entrepreneurs, businesses, and government agencies worldwide. Therefore, international transactions will often be considered.
* **Are there governmental restrictions on secure communications to consider?**
  + As a worldwide financial institution, Artemis Financial will have many localized and international guidelines to consider. One example is the Patriot Act, which requires secure communication with the Financial Criminal Enforcement Network (FinCEN). FinCEN is a federal organization that monitors international transactions. Therefore, Artemis Financial is required to adhere to its guidelines.
* **What external threats might be present now and in the immediate future?**
  + Denial of Service (DoS): Clients and users must always be able to access their information, and a DoS attack could be detrimental to the business.
  + Man In the Middle (MITM): Artemis Financial must be able to communicate securely worldwide and must be vigilant against communication interception.
  + Injection Attacks: This involves integrating input into commands, where malicious input may result in unintended functionality or data distribution.
  + Spoofing – A spoofed request may result in a data leak or even an escalation of privileges by a malicious user.
* **What modernization requirements must be considered, such as the role of open-source libraries and evolving web application technologies?**
  + Dependencies should regularly be scanned using the latest tools, such as the OWASP dependency-checker.
  + Prefer using libraries with active communities to ensure security updates and support.
  + Implement best security practices for interacting with RESTful API’s.
  + Implement HTTPS for secure communications.

**2. Areas of Security**

* **Input Validation**
  + This aids in the prevention of injection attacks, which could be disastrous.
* **APIs**
  + Artemis Financial must securely interact with its RESTful API.
* **Cryptography**
  + Communications must be securely encrypted and transmitted as they will contain sensitive data.
* **Code Error**
  + Errors must be appropriately handled to prevent a DoS or an unintended escalation of privileges.
* **Encapsulation**
  + Encapsulation helps ensure that sensitive data and functionality are kept secure. An adequately encapsulated system can limit damage if/when a service is broken.
* **Client / Server**
  + Secure client/server communication is essential for confidentiality and integrity.
* **Code Quality**
  + The program must be developed using secure coding practices and patterns.

**3. Manual Review**

* **CRUDController.java**
  + Contains a vulnerability related to the mapping variable. The pattern uses request information to direct the request. A malicious user could alter the cookie to access sensitive information or functionality.
  + Business name parameter is never properly sanitized or parameterized before being used in a command.
* **GreetingController.java**
  + Contains a vulnerability related to the mapping variable. The pattern uses request information to direct the request. A malicious user could alter the cookie to access sensitive information or functionality.
  + The name parameter is never properly sanitized or parameterized before being used in a command.
* **customer.java**
  + Inconsistent naming.
  + Lack of encapsulation. accountBalance should be private, with appropriate getters and setters to access data.
  + Class functions should use proper access control methods to ensure appropriate use.
* **DocData.java**
  + Key and value parameters are never sanitized.
  + Implement parameterization
* **CRUD.java**
  + Needs an appropriate authenticator and authorizer.
  + The parameters should be properly sanitized before being committed to memory.
  + Parameterization of the variables used to access/modify the database is necessary.
* **myDateTime.java**
  + Inconsistent naming.
  + The date variables should be private with appropriate getters/setters.

**4. Static Testing**

**Dependency:** [bcprov-jdk15on-1.46.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l1_991c96a4e31e6c19e2b9136c8955bd423f2dc4c7)

**Vulnerability ID:** [cpe:2.3:a:bouncycastle:bouncy\_castle\_for\_java: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%3Abouncy_castle_for_java&cpe_version=cpe%3A%2F%3Abouncycastle%3Abouncy_castle_for_java%3A1.46)

**Description:** The listed vulnerabilities in the Bouncy Castle cryptography library primarily involve improper validation, cryptographic issues, and resource exhaustion. These include improper validation of certificates, leading to potential host mismatches, and improper verification of cryptographic signatures, allowing injection of extra elements.

**Potential Solution:** All security issues are addressed in more up-to-date versions.

**Dependency:** [hibernate-validator-6.0.18.Final.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l3_7fd00bcd87e14b6ba66279282ef15efa30dd2492)

**Vulnerability ID:** [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)

**Description:** A flaw in Hibernate Validator 6.1.2.Final allows invalid expressions to be mistakenly treated as valid. This means attackers can bypass the safety measures developers use to handle user input in error messages, potentially leading to security issues.

**Potential Solution:** This error has been fixed in more recent solutions.

**Dependency:** [jackson-databind-2.10.2.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l5_0528de95f198afafbcfb0c09d2e43b6e0ea663ec)

**Vulnerability ID:** [cpe:2.3:a:fasterxml:jackson-databind:2.10.2:\*:\*:\*:\*:\*:\*:\*](https://nvd.nist.gov/vuln/search/results?form_type=Advanced&results_type=overview&search_type=all&cpe_vendor=cpe%3A%2F%3Afasterxml&cpe_product=cpe%3A%2F%3Afasterxml%3Ajackson-databind&cpe_version=cpe%3A%2F%3Afasterxml%3Ajackson-databind%3A2.10.2)  
 cpe:2.3:a:fasterxml:jackson-modules-java8:2.10.2:\*:\*:\*:\*:\*:\*:\*

**Description:** Multiple versions of Jackson-data-bind are vulnerable to various forms of denial of service, typically through excessive resource usage or deep nesting of objects/arrays

**Potential Solution:** Update to version 2.15.3 or later.

**Dependency:** [log4j-api-2.12.1.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l10_a55e6d987f50a515c9260b0451b4fa217dc539cb)

**Vulnerability ID:** [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)

**Description:** Vulnerable to man-in-the-middle attacks that could lead to private log information being leaked.

**Potential Solution:** Update to version 2.17.0, 2.12.3, 2.3.1 or most recent.

**Dependency:** [logback-core-1.2.3.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l12_864344400c3d4d92dfeb0a305dc87d953677c03c)

**Vulnerability ID**: [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)

**Description:** A serialization error that could cause a DoS and an error that may allow malicious configurations.

**Potential Solution:** Update to version 1.2.8 or later.

**Dependency:** [snakeyaml-1.25.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l14_8b6e01ef661d8378ae6dd7b511a7f2a33fae1421)

**Vulnerability ID:** [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)

**Description:** Various stackoverflow vulnerabilities that may lead to DoS attacks.

**Potential Solution:** Sanitize user input before parsing. Update to version 2.0 or later.

**Dependency:** [spring-boot-2.2.4.RELEASE.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l15_225a4fd31156c254e3bb92adb42ee8c6de812714)

**Vulnerability ID:** [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)

**Description:** Vulnerable to DoS, security bypass, and temporary directory hijacking.

**Potential Solution:** Update to supported versions.

**Dependency:** [spring-boot-starter-web-2.2.4.RELEASE.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l16_ec75d01d212b5229c16d872fb127744c0ed46ed8)

**Vulnerability ID:** [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)

**Description:** There are many vulnerabilities associated with this dependency. Risks include remote code execution, privilege escalation, DoS, and injection attacks.

**Potential Solution:** Update to supported versions.

**Dependency:** [spring-core-5.2.3.RELEASE.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l17_3734223040040e8c3fecd5faa3ae8a1ed6da146b)

**Vulnerability ID:** [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\_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)

**Description:** As with the other spring dependencies, many vulnerabilities have been identified, including remote code execution, DoS, privilege escalation, and injection attacks.

**Potential Solution:** Update to supported versions.

**Dependency:** [spring-web-5.2.3.RELEASE.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l18_dd386a02e40b915ab400a3bf9f586d2dc4c0852c)

**Vulnerability ID:** [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\_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)

**Description:** As with the other spring dependencies, many vulnerabilities have been identified, including remote code execution, DoS, privilege escalation, and injection attacks.

**Potential Solution:** Update to supported versions

**Dependency:** [spring-webmvc-5.2.3.RELEASE.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l19_745a62502023d2496b565b7fe102bb1ee229d6b7)

**Vulnerability ID:** [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\_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)

**Description:** As with the other spring dependencies, many vulnerabilities have been identified, including remote code execution, DoS, privilege escalation, and injection attacks.

**Potential Solution:** Update to supported versions

**Dependency:** [tomcat-embed-core-9.0.30.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l20_ad32909314fe2ba02cec036434c0addd19bcc580)

**Vulnerability ID**: [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)

**Description:** Vulnerabilities include improper privilege management, DoS, information disclosure, request smuggling, and remote code execution.

**Potential Solution:** All security issues are addressed in more up-to-date versions.

**Dependency:** [tomcat-embed-websocket-9.0.30.jar](file:///Users/johnmiller/Desktop/rest-service/target/dependency-check-report.html#l22_33157f6bc5bfd03380ebb5ac476db0600a04168d)

**Vulnerability ID:** [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)

**Description:** Vulnerabilities include improper privilege management, DoS, information disclosure, authentication errors, and request smuggling.

**Potential Solution:** All security issues are addressed in more up-to-date versions.

**5. Mitigation Plan**

* **Up-to-date Libraries**
  + Continually update dependencies when possible. Be wary of using older unsupported versions of libraries. Newer versions work to patch known security vulnerabilities.
* **Keep Trusted and Untrusted Data Separate**
  + Sanitize all user input before committing it to memory.
* **Implement a secure authenticator.**
  + Users shall have to enter a username and key.
  + Log all authentication failures for intrusion detection.
* **Use session management**
  + Implement a singleton pattern to store user information on the server side for access controls.
* **Implement access controls**
  + Role-based access controls or a more contextual approach should be implemented to limit access.
  + Use the rule of ‘least privilege,’ where users are assigned the least privilege necessary to complete duties.
  + Perform consistent authorization checks across all requests.
* **Encapsulate**
  + Keep data members private with appropriate getters/setters. This will keep the code more secure when a class is inevitably broken.
* **Verify each request**
  + Request verification should be handled server-side, and authentication should be tied to a session or contextual so that a changed URL or cookie does not escalate privileges.
  + Treat internal requests as untrusted data as well.
* **Validate Input**
  + Set appropriate limitations on input where relevant, such as only alpha-numeric characters or no trailing/leading hyphens (-).
  + A blacklist of known threatening input in conjunction with a whitelist of acceptable input is recommended.
* **Parameterize Variables**
  + One of the best methods against injection attacks is to parameterize variables when using them in database commands or storage. The database will create a path for the command and execute it with the user input, ensuring the command only follows the created path.