# CS 305 Project One Template

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **02/05/2025** | **Matthew A Keaton** |  |

## Client



## Developer

Matthew A Keaton

**1. Interpreting Client Needs**

The client is a consulting company called Artemis Financial that develops financial plans for its customers including savings, retirement, investments, and insurance. The company is looking to add effective security improvements to protect sensitive data and promote client trust. Having secure communication is critical to the company since they will be handling sensitive financial information. Majority of the communications from Artemis Financial will contain sensitive or personal information which can lead to attackers trying to steal that information for identity theft to gain access to customer accounts. It is unsure if there are any internation transactions that the company produces. However, due to the plans for improved security it is safe to assume that Artemis Financial is looking to expand its organization internationally or already have international clients. If Artemis Financial conducts international transaction they need to be to be in compliance with any international laws or regulations. Governmental restrictions on secure communications to be considered is the Gramm-Leach-Bliley Act (GLBA) that requires financial institutions to protect customer data and ensure secure communications. Potential external threats present now and in the future are Man-in-the-Middle attacks, identity theft, injection attacks, insecure API’s that leak data, DDoS, and phishing attacks. These threats are vulnerabilities that can be exploited by external threats. Open-source libraries are used in software development and offer functions that help encrypt, authenticate, and process data. Using open-source libraries, Artemis Financial must conduct regular vulnerability scans from tools like OWASP dependency check to help prevent supply chain attacks. They must also use secure API designs and input validations to mitigate authentication and injection risks. Another modern requirement that must be considered is DevSecOps best practices where security is considered in every stage of the process and not just in one specific stage.

**2. Areas of Security**

Input Validation – Ensures that the data going into Artemis Financials’ web application is secure and safe. The web application requires input from the client and server side where input validations ensures that malicious data is not being processed. This will help mitigate SQL and XSS injection attacks.

API’s – Having secure API’s is relevant to Artemis Financial because their web application uses APIs for the communication between clients and the server. If these APIs are not secure this could lead to unauthorized access to sensitive data.

Cryptography – Is extremely relevant to Artemis Financial because encryption must be used to secure any financial transaction that are processed through their web application. Cryptography techniques such as digital signatures are used to ensure data integrity and that the data has not been changed.

Client/Server – Is relevant to Artemis Financial since it lays out the architecture used for the structure of communications from client and the server. Communication between the client and server must be secure since they primarily contain sensitive information. It is critical to understand the vulnerabilities in client/server interactions and build a secure architecture for Artemis Financials’ web-based application.

Code Quality – Is relevant to Artemis Financial because the web application will be processing sensitive financial information and require efficient software. Having high quality code will ensure that the program runs as expected while also promoting good user experience. It is important that the code has good quality to promote readability, maintainability, scalability, performance, cost effectiveness, and user trust.

**3. Manual Review**

* A vulnerability found in the DocData.java class file. The read\_document method in this class uses the username and password in the code as “root” that is a security vulnerability.
* A vulnerability found in CRUDController.java class file. The “/read” endpoint used can retrieve data without any authentication or restrictions. It makes the system vulnerable to retrieving malicious code as well.
* A vulnerability found in GreetingController.java class file. The parameter “name” used in the method Greeting is pulled from the URL and could be used to inject malicious code.
* A vulnerability found in customer.java class. The method ShowInfo() returns the customer account numbers.
* A vulnerability found in the customer.java class. The variable account\_balance should be made private and not accessible by other classes.
* A vulnerability found in the customer.java class. The class has no authentication checks and allows anyone to modify the balance.
* A vulnerability found in the myDateTime.java class. The variables declared for mySecond, myMinute, and myHour should be set to private. This ensures that other classes can’t access them and modify them. This could lead to data integrity risks.

**4. Static Testing**

A screenshot of a web page

Description automatically generated

Maven Dependency Check Report

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Vulnerability ID** | **Description** | **Solution** |
| Bcprov-jdk15on-1.46.jar | [CVE-2023-33202](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) | Bouncy Castle for Java before 1.73 contains a potential Denial of Service (DoS) issue within the Bouncy Castle org.bouncycastle.openssl.PEMParser class. | Upgrade to a version of 1.73 or later where the vulnerability has been patched. |
| Hibernate-validator-6.0.18.Final.jar | [CVE-2020-10693](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) | A flaw was found in Hibernate Validator version 6.1.2.Final. A bug in the message interpolation processor enables invalid EL expressions to be evaluated as if they were valid. Allowing attackers to bypass input sanitation. | Upgrade to later version. Use input validation for invalid EL expressions. |
| Jackson-databind-2.10.2.jar | [CVE-2023-35116](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) | jackson-databind through 2.15.2 allows attackers to cause a denial of service or other unspecified impact via a crafted object that uses cyclic dependencies | Upgrade to later or current version. |
| Log4j-api-2.12.1.jar | [CVE-2021-44832](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) | Apache Log4j2 versions 2.0-beta7 through 2.17.0 are vulnerable to a remote code execution (RCE) attack when a configuration uses a JDBC Appender with a JNDI LDAP data source URI when an attacker has control of the target LDAP server | Limit JNDI data sourcenames to the java protocol in Log4j2 versions 2.17, 2.12.4, and 2.3.2 |
| Logback-classic-1.2.3.jar | [CVE-2023-6378](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) | A serialization vulnerability in logback receiver component part of logback version 1.4.11 allows an attacker to mount a Denial-Of-Service attack by sending poisoned data | Upgrade to later version or input validation. |
| Logback-core-1.2.3.jar | [CVE-2023-6378](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) | A serialization vulnerability in logback receiver component part of logback version 1.4.11 allows an attacker to mount a Denial-Of-Service attack by sending poisoned data | Upgrade to later version or input validation. |
| Snakeyaml-1.25.jar | [CVE-2022-1471](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) | SnakeYaml's Constructor() class does not restrict types which can be instantiated during deserialization. Deserializing yaml content provided by an attacker can lead to remote code execution. | Use SnakeYaml’s SafeConstructor when parsing untrusted content to restrict deserialization and upgrade to version 2.0 or later. |
| Spring-boot2.2.4.RELEASE.jar | [CVE-2023-20883](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) | In Spring Boot versions 3.0.0 - 3.0.6, 2.7.0 - 2.7.11, 2.6.0 - 2.6.14, 2.5.0 - 2.5.14 and older unsupported versions, there is potential for a denial-of-service (DoS) attack if Spring MVC is used together with a reverse proxy cache. | Upgrade to 3.0.7 or later versions. |
| Spring-boot-starter-web-2.2.4.RELEASE.jar | [CVE-2023-20883](https://nvd.nist.gov/vuln/detail/CVE-2023-20883) | In Spring Boot versions 3.0.0 - 3.0.6, 2.7.0 - 2.7.11, 2.6.0 - 2.6.14, 2.5.0 - 2.5.14 and older unsupported versions, there is potential for a denial-of-service (DoS) attack if Spring MVC is used together with a reverse proxy cache. | Upgrade to 3.0.7 or later versions. |
| Spring-core-5.2.3.RELEASE.jar | [CVE-2023-20863](https://nvd.nist.gov/vuln/detail/CVE-2023-20863) | In spring framework versions prior to 5.2.24 release+ ,5.3.27+ and 6.0.8+ , it is possible for a user to provide a specially crafted SpEL expression that may cause a denial-of-service (DoS) condition | Upgrade to a later or current version. Input validation for SpEL expressions. |
| Spring-expression-5.2.3.RELEASE.jar | [CVE-2023-20863](https://nvd.nist.gov/vuln/detail/CVE-2023-20863) | In spring framework versions prior to 5.2.24 release+ ,5.3.27+ and 6.0.8+ , it is possible for a user to provide a specially crafted SpEL expression that may cause a denial-of-service (DoS) condition | Upgrade to a later or current version. Input validation for SpEL expressions. |
| Spring-web-5.2.3.RELEASE.jar | [CVE-2023-20863](https://nvd.nist.gov/vuln/detail/CVE-2023-20863) | In spring framework versions prior to 5.2.24 release+ ,5.3.27+ and 6.0.8+ , it is possible for a user to provide a specially crafted SpEL expression that may cause a denial-of-service (DoS) condition | Upgrade to a later or current version. Input validation for SpEL expressions. |
| Spring-webmvc-5.2.3.RELEASE.jar | [CVE-2023-20863](https://nvd.nist.gov/vuln/detail/CVE-2023-20863) | In spring framework versions prior to 5.2.24 release+ ,5.3.27+ and 6.0.8+ , it is possible for a user to provide a specially crafted SpEL expression that may cause a denial-of-service (DoS) condition | Upgrade to a later or current version. Input validation for SpEL expressions. |
| Tomcat-embed-core-9.0.30.jar | [CVE-2024-21733](https://nvd.nist.gov/vuln/detail/CVE-2024-21733) | Generation of Error Message Containing Sensitive Information vulnerability in Apache Tomcat.This issue affects Apache Tomcat: from 8.5.7 through 8.5.63, from 9.0.0-M11 through 9.0.43. | Users are recommended to upgrade to version 8.5.64 onwards or 9.0.44 onwards, which contain a fix for the issue |
| Tomcat-embed-websocket-9.0.30.jar | [CVE-2024-21733](https://nvd.nist.gov/vuln/detail/CVE-2024-21733) | Generation of Error Message Containing Sensitive Information vulnerability in Apache Tomcat.This issue affects Apache Tomcat: from 8.5.7 through 8.5.63, from 9.0.0-M11 through 9.0.43. | Users are recommended to upgrade to version 8.5.64 onwards or 9.0.44 onwards, which contain a fix for the issue |

**5. Mitigation Plan**

The results in the dependency check report show that most security concerns can be solved by upgrading all software used to its latest or current version. The code review highlighted several security vulnerabilities in the code base that need to be addressed. The code should be changed to implement security best practices and ensure the latest versions or being used. The action list below shows the steps needed to be taken to ensure all security vulnerabilities are addressed.

Action List:

Code Base

* Remove login credentials from source code
* Implement authentication and authorization access controls
* Input validation to sanitize user input
* Privatize the access to sensitive variables in the code

Libraries and Tools

* Upgrade all to their latest versions
* SnakeYaml upgrade to current version and use SnakeConstructor
* Limit JNDI sourcename lookups