# CS 305 Project One Template

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **11/16/2024** | **Justin Schumann** |  |

## Client



## Instructions

Submit this completed vulnerability assessment report. Replace the bracketed text with the relevant information. In this report, identify your security vulnerability findings and recommend 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 include images or supporting materials. If you include them, make certain to insert them in 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

Justin Schumann

**1. Interpreting Client Needs**

Determine your client’s needs and potential threats and attacks associated with the company’s application and software security requirements. Consider the following questions regarding how companies protect against external threats based on the scenario information:

* What is the value of secure communications to the company?
* Are there any international transactions that the company produces?
* Are there governmental restrictions on secure communications to consider?
* What external threats might be present now and in the immediate future?
* What modernization requirements must be considered, such as the role of open-source libraries and evolving web application technologies?

Artemis Financial is a consulting firm that consults other companies involving finance, giving whole financial plans. These plans involve savings, retirement plans, investments and insurance. Since our client provides private information such as SSN, income, and tax, securing that information only between Artemis Financial and their clients is very important. Artemis Financial would have to deal with the typical government restrictions. These restrictions may vary if Artemis Financial operates internationally. In the U.S, all sensitive information must be confidential, meaning any sensitive information exchanged between parties must be kept only between these parties. Since the information involved could be valuable for harmful uses, it would be a prime target for hackers and data thieves, which is why it all has to be encrypted. Software involving security is constantly revolving, as too the tools and methods of hackers and cyberattacks. Every few months there should be an annual maintenance, checking for weak spots and bugs.

**2. Areas of Security**

Refer to the vulnerability assessment process flow diagram. Identify which areas of security apply to Artemis Financial’s software application. Justify your reasoning for why each area is relevant to the software application.

Input Validation: To protect users and information, all account owners must be validated. Allowing unauthorized access to information could cause damage to Artemis Financial and their clients.

Cryptography: all information will be encrypted, making it less accessible to unauthorized individuals.

APIs: there must be a method to separate acceptable information from un-acceptable information. Creating an API will allow this.

Code Error: when creating long complex codding, errors will be expected. And reviewed for correction is needed. Codding errors could allow a breach in security.

Code Quality: a user must only have access to their information, not the information of others, with Code Quality access could be based on the user.

**3. Manual Review**

Continue working through the vulnerability assessment process flow diagram. Identify all vulnerabilities in the code base by manually inspecting the code.

[Include your 7–10 findings here.]

When looking through the code, the first thing I notice that there is no use of Hypertext Transfer Protocol Secure or HTTPS. HTTPS is a secure version of HTTP, that is encrypted for protection during data transfer between a website and browser. What is oblivious is a total absent of an account owner authentication system. No where in the code dose it uses cryptography. From all these things stated, security breaches are a significant possible risk.

**4. Static Testing**

Run a dependency check on Artemis Financial’s software application to identify all security vulnerabilities in the code. Record the output from the dependency-check report. Include the following items:

* The names or vulnerability codes of the known vulnerabilities
* A brief description and recommended solutions provided by the dependency-check report
* Any attribution that documents how this vulnerability has been identified or documented previously
  + - 1. log4j-api-2.12.1.jar: Severity level: CRITICAL and Confidence: HIGHEST. Vulnerability numbered only to one. Have mismatch with host, solution will be to upgrade to the most recent update.
      2. bcprov-jdk15on-1.46.jar: Severity level: HIGH. Confidence: HIGHEST. Have about 19 vulnerabilities. Solution will be to upgrade to latest version.
      3. spring-boot-2.2.4.RELEASE.jar: Severity level: CRITICAL. Confidence: HIGHEST. Have three vulnerabilities. Solution will be to upgrade to latest version.
      4. snakeyaml-1.25.jar: severity level: CRITICAL. Confidence: HIGHEST. Have 10 vulnerabilities. Solution will be to upgrade to latest version.
      5. jackson-databind-2.10.2.jar: severity level: HIGH. Confidence: HIGHEST. Have 7 vulnerabilities. Solution will be to upgrade to latest version.
      6. tomcat-embed-core-9.0.30.jar: severity level: CRITICAL. Confidence: HIGHEST. Have a large number of vulnerabilities. Solution is to upgrade to latest version.
      7. spring-web-5.2.3.RELEASE.jar: Severity level: HIGH. Confidence: HIGHEST. Have 7 vulnerabilities. Solution is to upgrade to latest version.
      8. spring-expression-5.2.3.RELEASE.jar: Severity level: MEDIUM. Confidence: HIGHEST. Have 4 vulnerabilities. Solution is to upgrade to latest version.
      9. spring-webmvc-5.2.3.RELEASE.jar: severity level: HIGH. Confidence: HIGHEST. Have 2 vulnerabilities. Solution will be to upgrade to latest version.
      10. hibernate-validator-6.0.18.Final.jar: Severity level: MEDIUM. Confidence: HIGHEST. Have 2 vulnerabilities. Solution is to upgrade to latest version.
      11. spring-context-5.2.3.RELEASE.jar: Severity level: MEDIUM. Confidence: HIGHEST. Have 1 vulnerabilities. Solution is to upgrade to latest version.
      12. spring-beans-5.2.3.RELEASE.jar: Severity level: HIGH. Confidence: HIGHEST. Have 1 vulnerabilities. Solution is to upgrade to latest version.

**5. Mitigation Plan**

Interpret the results from the manual review and static testing report. Then identify the steps to mitigate the identified security vulnerabilities for Artemis Financial’s software application.

The most obvious solution is to see if there is a more updated version of the systems being used. Another thing is to add input and owner account authorization. Adding HTTPS and using cryptography would make the data much more secure.