# CS 305 Project One Template

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **11/17/2024** | **Alexander Ray** | **Vulnerabilities show that system needs update.** |

## 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

[Insert your name here.]

**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?

Since Artemis Financial is as its namesake implies, a company that deals with clients’ important financial information it is immediately evident why they would need a good security setup. Secure communications will be crucial to the company’s ability to coordinate and work with their clients as there are guaranteed to be malicious entities that wish to access the vital information being communicated through the system. Our company name Global Rain suggests we are equipped to deal with security on an international scale and since Artemis financial never specified its operation being limited to only within the US it is safe to assume that their company reach is international as well, creating an entirely new layer of complexity. We will need to account for the wider range of influence opening our program up to more possible threats. There are governmental restrictions on secure communication in the US that vary from state to state, but all revolve around the required protection of sensitive information, with some states even requiring that the information be destroyed within a specified time period after being received. All this means for us is that we have to keep doing what we were already planning to implement a strong security system that will protect the clients’ sensitive information. The financial consulting industry is often targeted by individuals and entities who wish to gain access to the customers’ important personal information such as social security numbers, bank statements, emails etc. Since so much important information is exchanged and recorded by these companies, they must be weary of the possibility of a breach in security. There will need to be protocols in place to protect and encrypt the company API connection such as an http.

**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** is the ability of the system to assess the inputs of the users and discern between what is acceptable and what is possibly malicious based on criteria decided by the developers. This will be crucial on a website with as much sensitive information on it as Artemis Financial’s website will.

Code quality is the first line of defense where the developers tighten the actual code and can build security measures into the code itself adding an extra layer of protection. For our client the financial consultants

**The APIs** are what allow the customers to connect on a unified server with Artemis and with an added protocol in in place this connection can be encrypted and protected. The protocol’s protection of our API will be very important as the users will be inputting their personal information into the website and that information should be encrypted and protected at all times.

**The client/server** aspect goes hand in hand with the APIs and their protocol encrypted systems, as any means of communication or coordination between the clients and the server must be monitored and protected to ensure that none of that information becomes available outside of the website.

**Code error** measures prevent the code from revealing any sensitive information when an error occurs while a user is on the website. Knowing exactly what the issue is could tip users onto the inner workings of the site and how to manipulate it, so it is important to keep error messages somewhat cryptic and generic.

**Cryptography** involves encryption of the messages and an added layer of protection against anyone without authorization to see the data in its original intended form. As many layers of encryption as we can have will ensure that the Artemis Financial website is safe for our customers.

Encapsulation is done within the code itself and involves abstracting objects and information in the program that the developers do not wish to be seen by those who are not authorized to do so.

**3. Manual Review**

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

The program does not use a proper protocol for the APIs like previously discussed which leaves user input onto the site visible and unprotected. There needs to be a protocol in place like the ever-popular http in order to prevent any possible future security breaches. In the same vein of mentioning encryption there are not enough measures put in place to encrypt website data. There was distinct lack of cryptography methods used throughout the program to provide extra protection from data breach. The input validation needs to be updated as there are not enough measures taken in this area to protect the program from those who may want to input harmful malware onto the site. There also exists a weakness in the code error returns as there is no active encryption of the messages to avoid revealing vital information to users.

**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

**bcprov-jdk15on-1.46.jar**

CVE-2013-1624, CVE-2015-6644, CVE-2015-7940, CVE-2016-1000338, CVE-2016-1000339, CVE-2016-1000341, CVE-2016-1000342, CVE-2016-1000343, CVE-2016-1000344, CVE-2016-1000345, CVE-2016-1000346, CVE-2016-1000352, CVE-2017-13098, CVE-2018-5382, CVE-2020-0187, CVE-2020-26939, CVE-2023-33201, CVE-2024-29857, CVE-2024-30171, CVE-2024-34447

CVE-2022-27772, CVE-2023-20873, CVE-2023-208833

***Synopsis: R****elated data left vulnerable to being breached by outside source. S*

***Solution:*** *Update all sections to latest version in system.*

**spring-boot-2.2.4.RELEASE.jar**

CVE-2022-27772, CVE-2023-20873, CVE-2023-20883

***Synopsis****: Related security measures may be bypassed due to obsolete model used.*

***Solution:*** *Update all sections to latest version in system.*

**log4j-api-2.12.1.jar**

CVE-2020-9488, CVE-2021-44228, CVE-2021-44832, CVE-2021-45046, CVE-2021-45105

***Synopsis****: An improper validation in this specific Api could lead to a breach in messages communicated within the system.*

***Solution:*** *Update all sections to latest version in system.*

**snakeyaml-1.25.jar**

CVE-2017-18640, CVE-2021-4235, CVE-2022-1471, CVE-2022-25857, CVE-2022-3064, CVE-2022-38749, CVE-2022-38750, CVE-2022-38751, CVE-2022-38752, CVE-2022-41854

***Synopsis****: System not secured during program expansion period.*

***Solution:*** *Update all sections to latest version in system.*

**jackson-databind-2.10.2.jar**

CVE-2020-25649, CVE-2020-36518, CVE-2021-46877, CVE-2022-42003, CVE-2022-42004, CVE-2023-35116

***Synopsis****: Improper security measures leave the system vulnerable, allowing for breach and weakened data integrity.*

***Solution:*** *Update all sections to latest version in system.*

**tomcat-embed-core-9.0.30.jar**

CVE-2019-17569, CVE-2020-11996, CVE-2020-13934, CVE-2020-13935, CVE-2020-13943, CVE-2020-17527, CVE-2020-1935, CVE-2020-1938, CVE-2020-8022, CVE-2020-9484, CVE-2021-24122, CVE-2021-25122, CVE-2021-25329, CVE-2021-30640, CVE-2021-33037, CVE-2021-41079, CVE-2021-43980, CVE-2022-29885, CVE-2022-34305, CVE-2022-42252, CVE-2023-28708, CVE-2023-41080, CVE-2023-42795, CVE-2023-44487, CVE-2023-45648, CVE-2023-46589, CVE-2024-2173

***Synopsis****: Tomcat failed to do second individualized thorough testing after the initial bulk testing.*

***Solution:*** *Update all sections to latest version in system.*

**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.

Most of the vulnerabilities found open the possibility of some form of a data breach so we need stronger API protocol protection as well as better cryptography and code quality so that there are not as many openings for malicious entities to steal data. Since as we established earlier Artemis financial is a financial consultant company and they are in possession of much of their clients’ personal information, the security measures in place should not leave as many openings for breaches in the data and this can be accomplished through having a tighter code with better encryption methods on every level. Updating all sections within the system to the most recent available source will put us in a much more secure spot than we are in now.