****

# 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** | **07/16/23** | **Scott Bosdash** |  |

## 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

Scott Bosdash

## Interpreting Client Needs

The company Artemis Financial is attempting to safeguard their business by keeping all their software current so they have no immediate threats to their security from outside events. There is a list of factors that can help them with that, including keeping their devices that are used for communication secure. This is important for any business, but particularly when a business is dealing with customers who entrust their financial documents and other personal files to that business, the files need to be protected from external threats. Unauthorized access should be shut down instantly to prevent any potential issues. Since Artemis will likely be dealing with countries outside of the US, they must be sure to adhere to any foreign and international laws that may apply to communications or transfers of funds to that end, privacy does not stop inside the states. There are also local laws that may restrict how the company handles its business, not just international, the company needs to be sure to follow those as well, laws vary by state and country of origin. Being hacked or getting viruses is a common problem, with this being the age of tech, more people are learning how to apply that technology in different, and sometimes sinister ways, Artemis needs to be sure they avoid scams such as that. Keeping the systems updated is just as important, using old or outdated technology is a potential risk factor, keeping the storage devices secure and updated is important. Using certain libraries could be another risk, ensuring that the libraries that are being used are current and secure should be a priority, the internet is ever changing, and the apps need to stay current and secure.

## Areas of Security

There are some areas where security is a concern that are of the utmost importance to the company, such as Authenticating the users, only a select few people should have access to the system, authorized individuals may access certain features that they need and no one else should be able to sign on. Confidential information will be recorded by Artemis, this should be kept private and secured as much as possible, with as little accessibility as possible, customers should be able to trust them with their personal information without regrets. Integrity is an important part of security, anyone who is not authorized should not be allowed to access or to change anything in the system. Lastly, there is the availability, when someone wants to use the app, they should be able to do so, assuming they have accessibility and are authorized to do so.

## Manual Review

After reading through the application, there were some potential threats in Artemis’ software, one of them is a SQL problem in one of the classes, someone could use this to add code which may pose a threat to the data. There is also a problem found in another class that could allow a hacker to run commands or scripts in the browser of a user of the app. The classes for logging in and out could be a tunnel for someone to take over a session if one were so inclined to do so. There is another threat due to an unsecured object, someone could gain access to parts of the system they are not authorized to view or access.

## Static Testing

By integrating the dependency check plugin for Maven on the Artemis software we identified some potential hazards along with a list of vulnerable dependencies scanned. They are mapped out with their IDs and what package they are in and listed severity levels of each. Using the dependency check also shows the user whether the dependency has been seen previously and how many times it has been seen. Some of the vulnerabilities mentioned are deserialization errors, cross site scripting, and SQL injection flaws, there are many more, but these are some examples, the rest may be seen in the html file.

## Mitigation Plan

The mitigation plan is lengthy but necessary. The serialization error can be remedied by using a secure open-source library that is able to deal with unregistered or untrustworthy certificates. To clean up the input for the application, Apache has a library that can handle such things and so does Google, either will work just fine. As far as sanitizing goes, OWASP and Apache have libraries to handle those situations, again, either one works well enough. To mitigate the SQL errors, the best way to handle that is to use a parameterized query if someone accesses the data, Spring has a useful tool to accomplish that task. With that said, there are some other options that could prove useful. One path would be more extensive training in secure coding for the staff, auditing the software frequently to keep everything current and up to date, and better encryption for the software.