****

# Practices for Secure Software Report

Table of Contents

[Document Revision History 3](#_Toc102040754)

[Client 3](#_Toc102040755)

[Instructions 3](#_Toc102040756)

[Developer 4](#_Toc102040757)

[1. Algorithm Cipher 4](#_Toc102040758)

[2. Certificate Generation 4](#_Toc102040759)

[3. Deploy Cipher 4](#_Toc102040760)

[4. Secure Communications 4](#_Toc102040761)

[5. Secondary Testing 4](#_Toc102040762)

[6. Functional Testing 4](#_Toc102040763)

[7. Summary 4](#_Toc102040764)

[8. Industry Standard Best Practices 4](#_Toc102040765)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **2/19/2023** | **Jason Ditullio** | **Filled in required sections of document** |

## Client



## Instructions

Submit this completed practices for secure software report. Replace the bracketed text with the relevant information. You must document your process for writing secure communications and refactoring code that complies with software security testing protocols.

* Respond to the 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 Two Guidelines and Rubric for more detailed instructions about each section of the template.

## Developer

Jason Ditullio

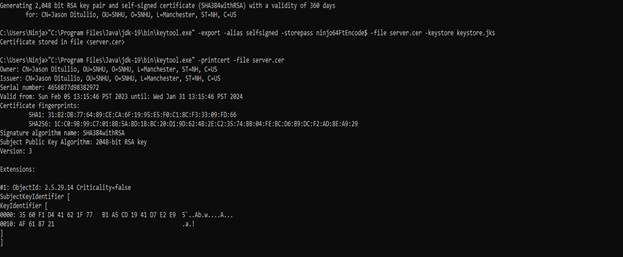
## Algorithm Cipher

After reviewing the ciphers available for Artemis Financial, I’ve recommended that they use AES for their cipher. The reasons for this are that the AES cipher is the standard used by the government for protecting and securing private or government files. It is open for public or private use, but if the organization that wants to use it is a nongovernmental organization then there are extra rules and limitations put in place by the U.S. export control. Another reason why AES would be useful here is that it uses symmetric keys rather than non-symmetric keys. The difference between symmetric and non-symmetric keys is that symmetric keys use the same exact secret key for encrypting and decrypting a file, while non-symmetric keys use two different keys that are linked to a file, one for encrypting and one for decrypting. Since it would use symmetric keys, it would lessen any confusion or chances of keys being mixed up for their information files. AES offers different bit levels for its encryption as well depending on what people need. These bit levels are 128, 192, and 256. Each level represents a different number of rounds of encryption to the data. For Artemis Financial, I’d recommend they use the 256-bit level as it offers the highest level of protection, making it harder for hackers to crack the encryption. Overtime, encryption algorithms have had to evolve and grow to defend against outside threats from hackers. Through the use of increasing bit-levels to newly implemented standards over the years, Encryption algorithms have been very useful in fending off attacks on sensitive data. While attacks and ways hackers can attack companies also grows and changes overtime, encryption algorithms will always be updating to protect and defend against them.

## Certificate Generation

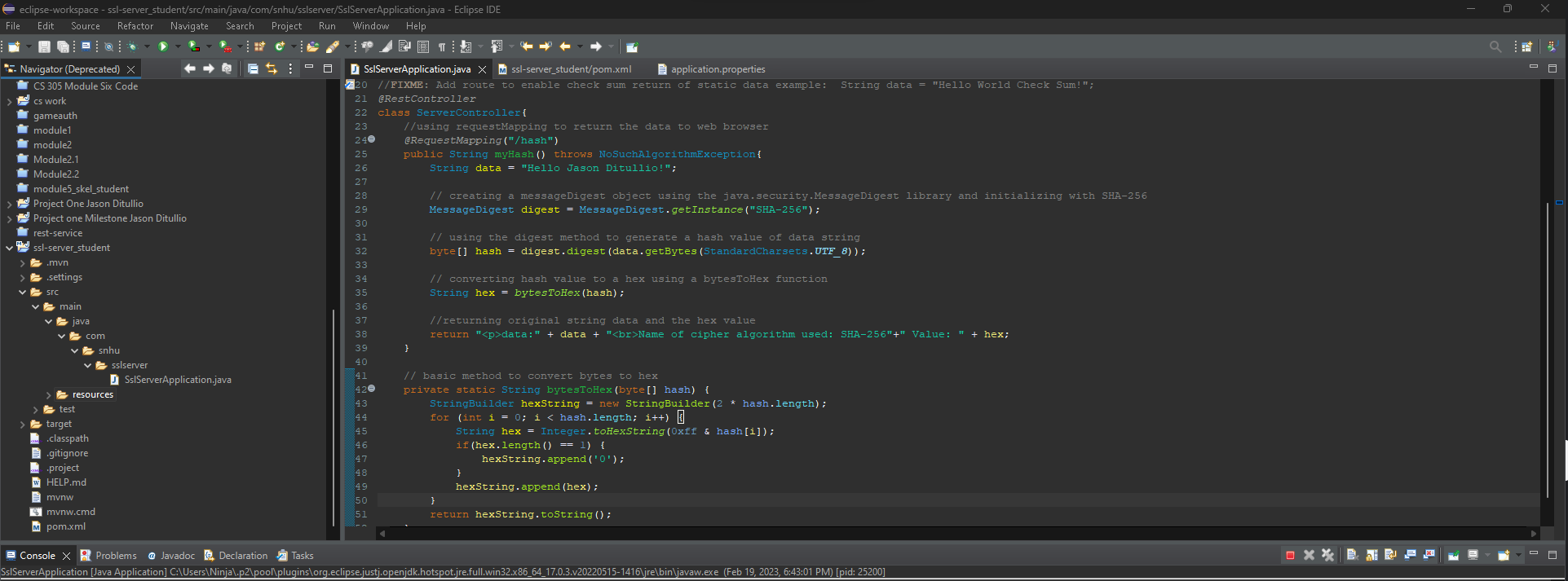
Insert a screenshot below of the CER file.

Text

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.



## Secure Communications

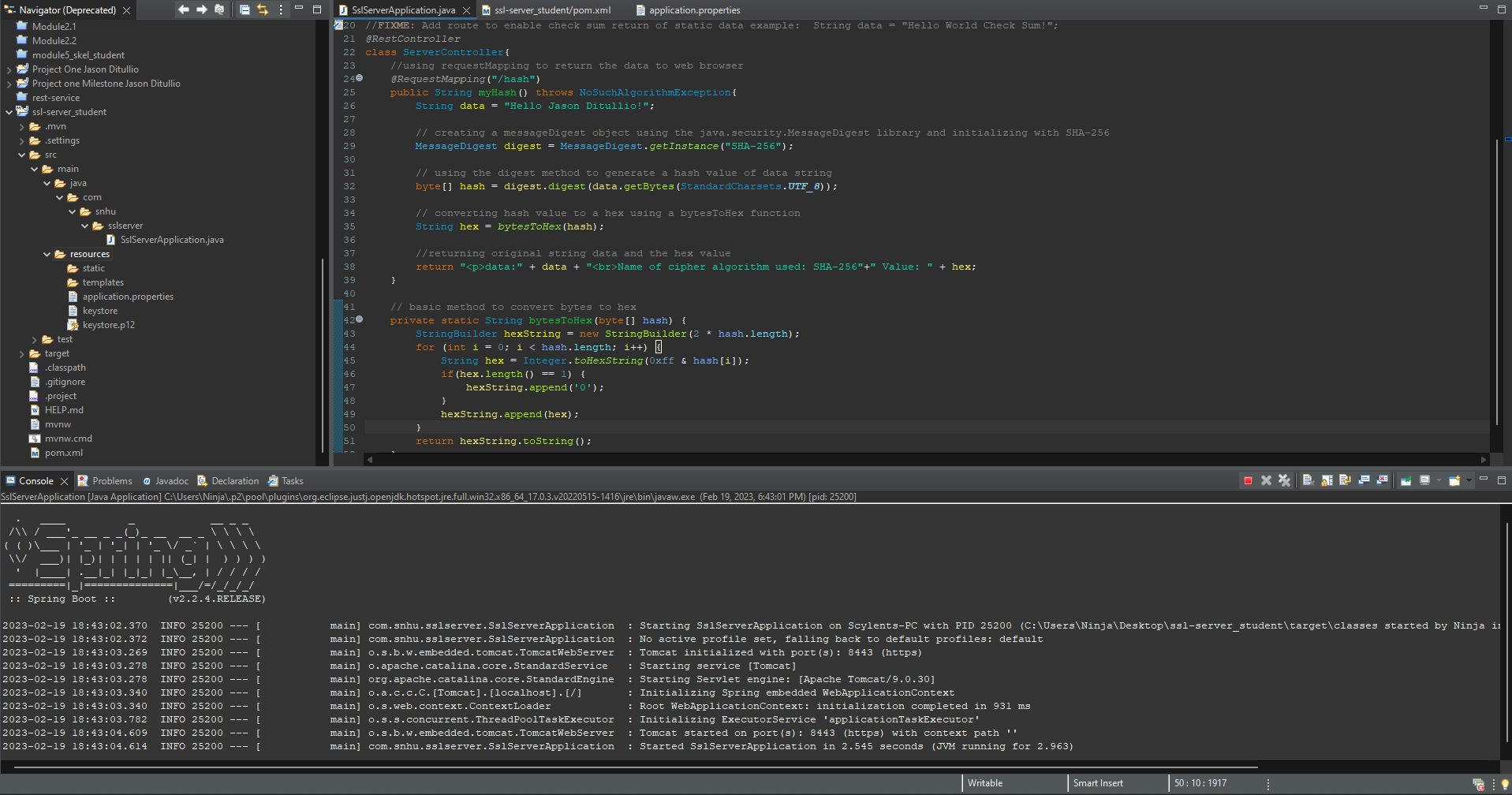
Insert a screenshot below of the web browser that shows a secure webpage.

Graphical user interface, text, application

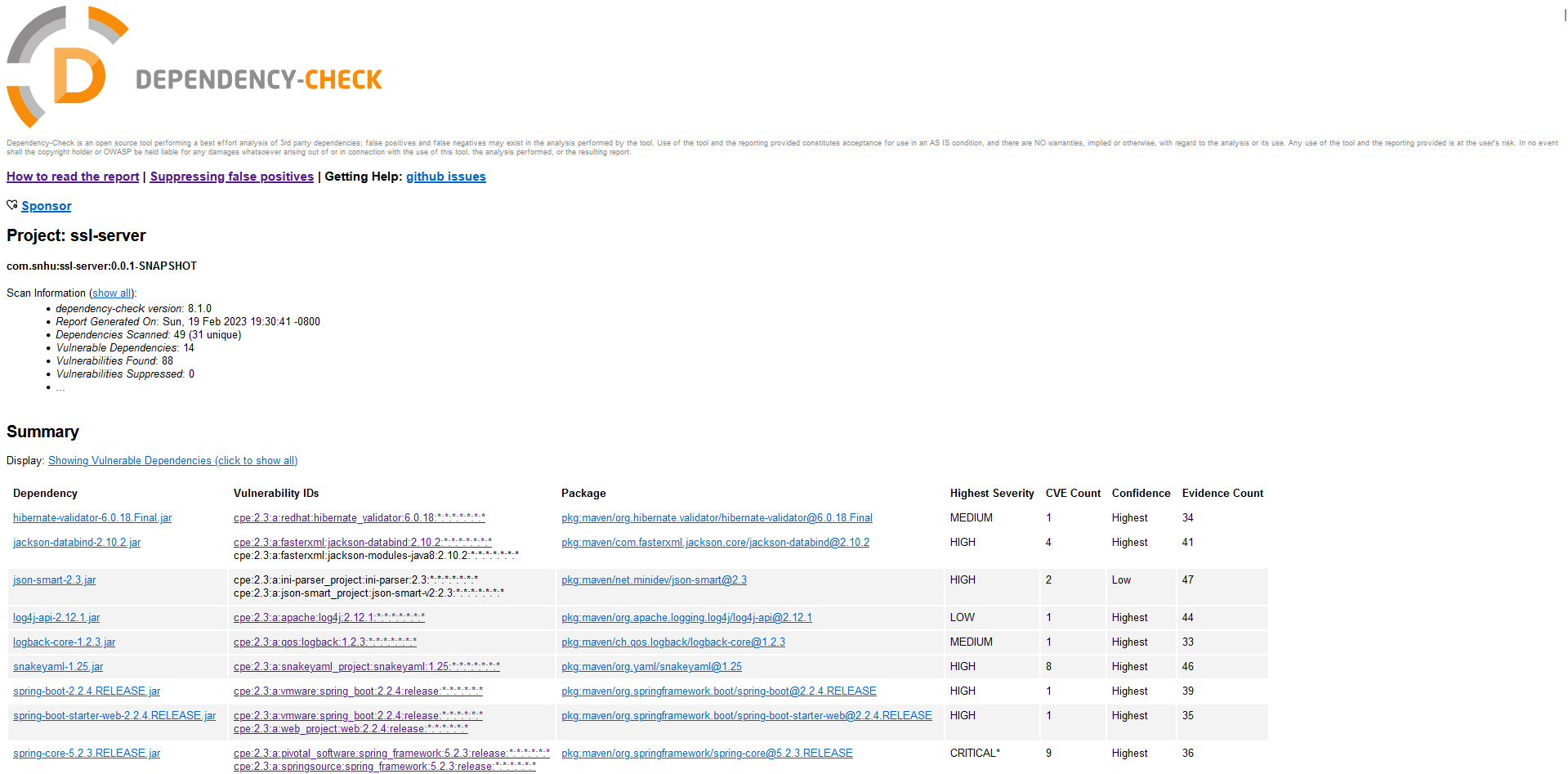
Description automatically generated

## Secondary Testing

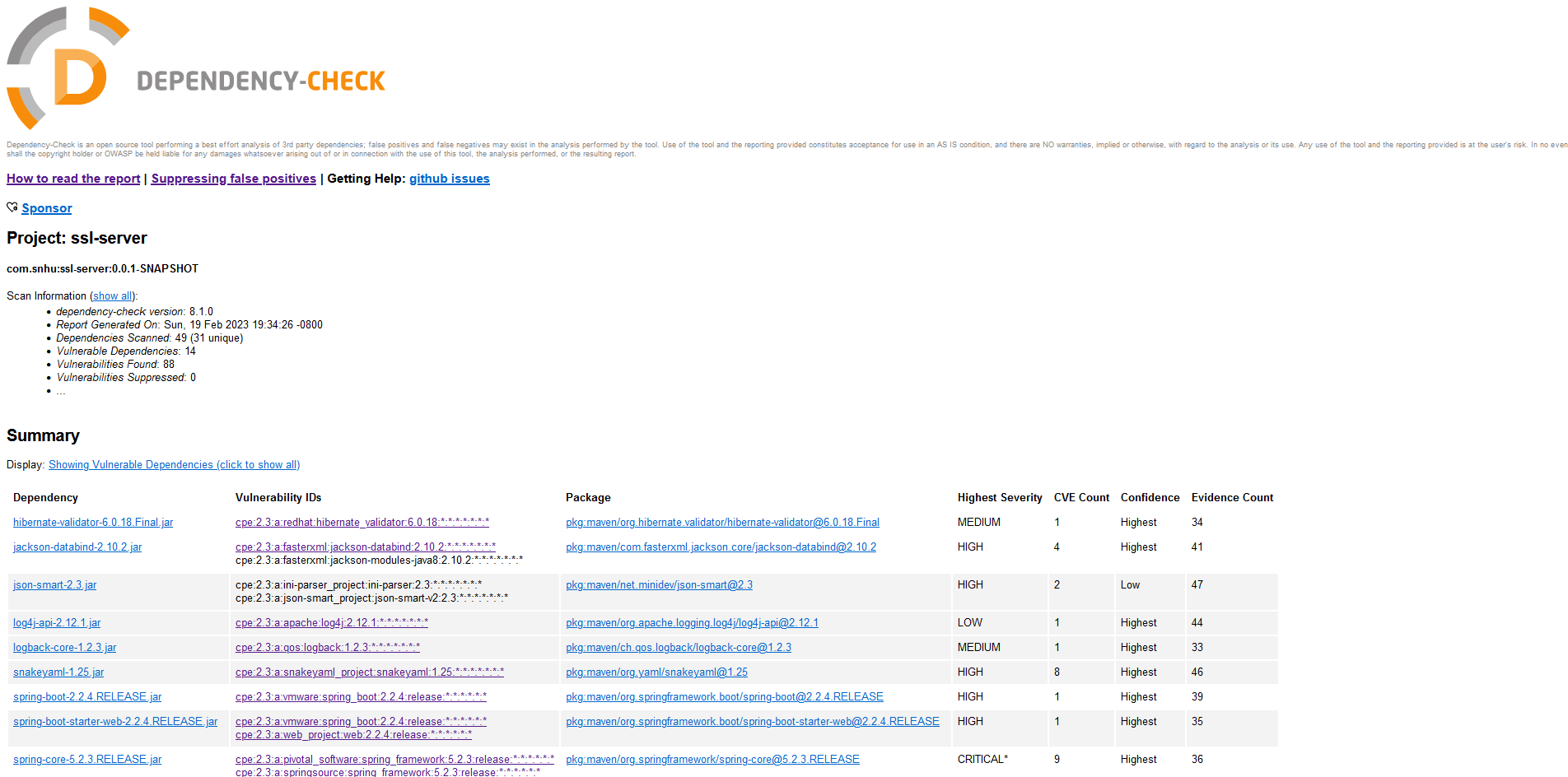
Insert screenshots below of the refactored code executed without errors and the dependency-check report.



**Dependency check before adding refactored code**

****

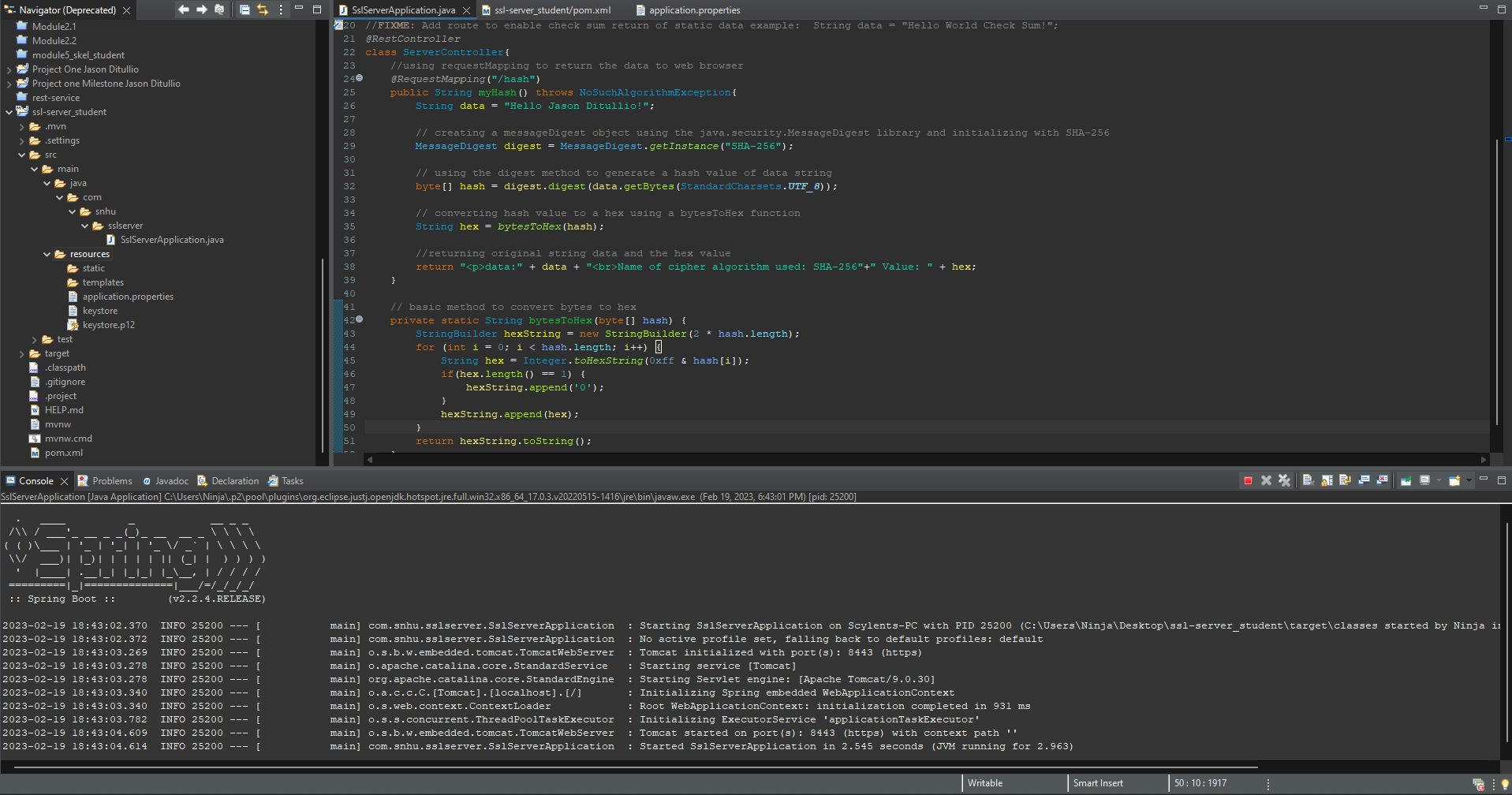
**Dependency check after adding refactored code**

****

After adding the refactored code there are no new vulnerabilities added as shown by the two screenshots above.

## Functional Testing

Insert a screenshot below of the refactored code executed without errors.



## Summary

While working on the code for this application, there were certain aspects of the Vulnerability Assessment Process that I addressed through the changes. By applying industry standard best practices to the application, I addressed Code Quality by making sure that the code worked properly and was free of any errors that could cause the application to not work. Two other aspects that I addressed were cryptography and client/server interactions. By adding a checksum and encryption to the application, I made sure that data using a hash algorithm would be encrypted and secured. This also shows that data between the clients and the server is being encrypted and secured. The different layers of security I added were a self-signed certificate that was added to the application and a cryptographic hash algorithm to demonstrate the effectiveness of encrypting information in Artemis Financials’ application. By adding these two layers of security, Artemis Financial can protect their sensitive financial documents from hackers and ensure that any data that is being accessed or transferred between users is encrypted and secured.

## Industry Standard Best Practices

When working on securing the code, I made sure to follow industry standard practices to make sure that I did not compromise the applications current security with my changes. I did this by first making sure that my code did not include any errors or improper syntax that would cause the application to fail to run. I also made sure that If I added any new dependencies or imports to the program that they would work with the application and did not add any new vulnerabilities. To do this I checked the dependency report before and after I refactored the code and made sure there were no new vulnerabilities. It is important to apply industry standard best practices for secure coding because it makes sure that there will be no errors or added vulnerabilities when Artemis Financial tries to run their application. It also makes sure that the added secure code is easy to read and is not confusing to others.