****

# 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** | **[Date]** | **[Your Name]** |  |

## 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

Scot C. Batton

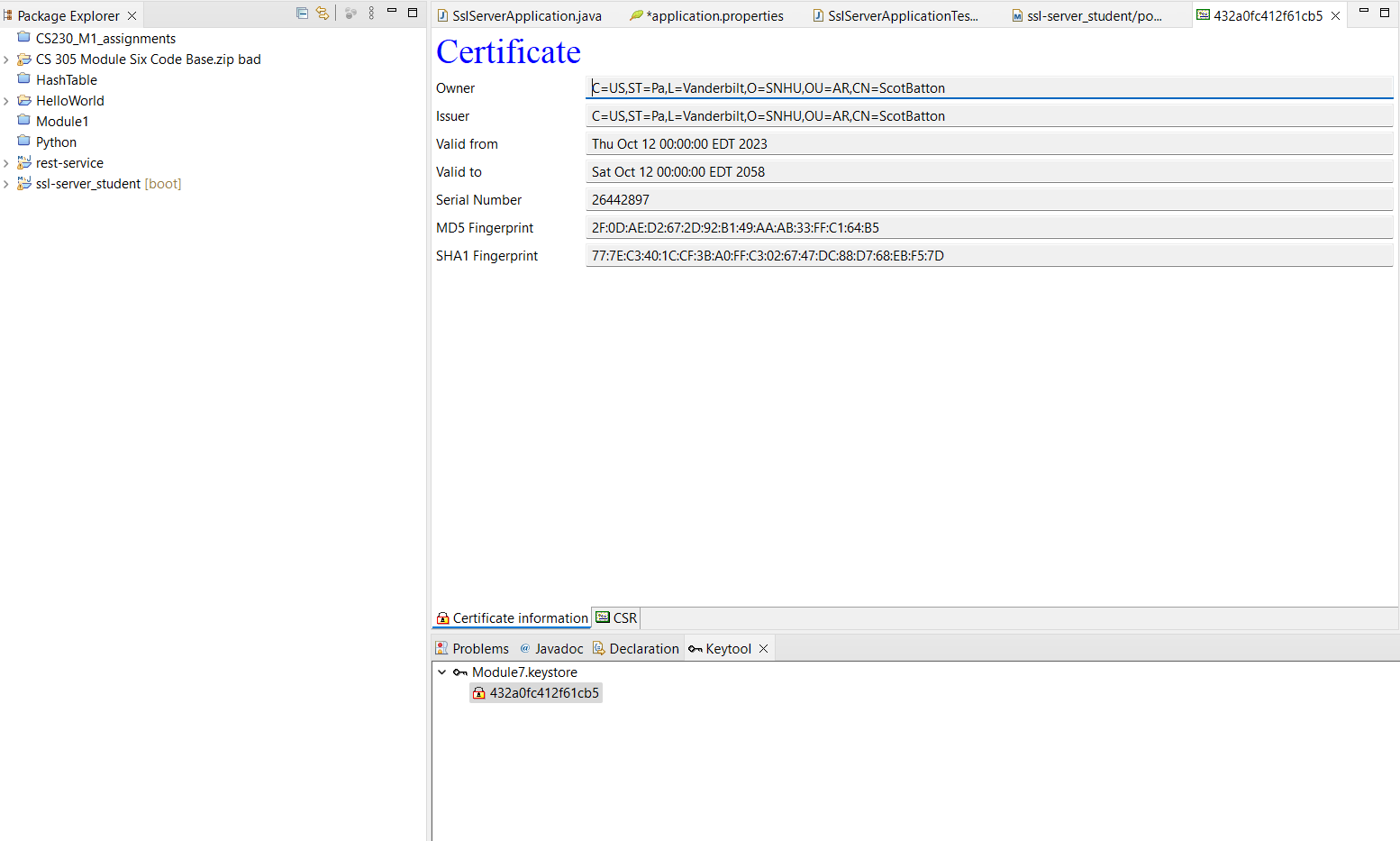
## Algorithm Cipher

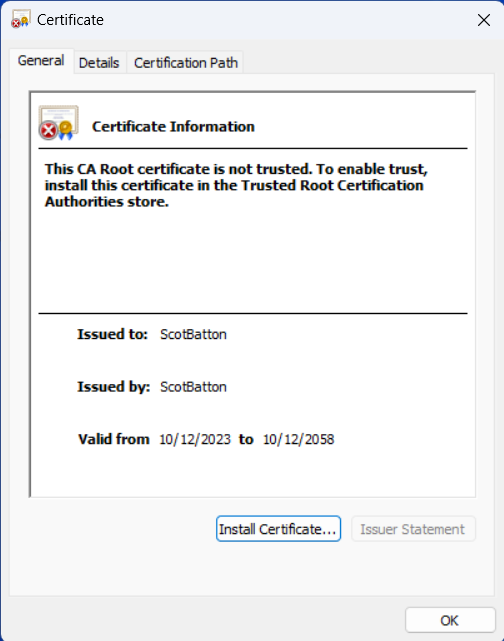
Artemis Financial is in the middle of a software examination to address vulnerabilities within the security of their software. As has been seen throughout this examination process, there are vulnerabilities within every system, and it is our job to mitigate them and provide the most secure and flexible means of security as possible. For this system I would recommend using SHA-256 encryption algorithm cypher with 256-bit keys. SHA-256 provides high level bit encryption and is one of the most widely used hash algorithms. SHA-256 is a part of the SHA-2 family of algorithms and was developed by the NSA and NIST in 2001 to fight back against brute force attacks (Jena, 2023). SHA-256 has 256-bits and 32 bytes. The bits and the hash functions are random. The length of the encryption is determined by the bit size.

The use of random numbers within SHA-256 makes it much harder for hackers to gain access to private and personal systems and allows for the system to remain unpredictable and ever changing. Within these systems there are symmetric and non-symmetric keys. Symmetric keys use the same key for encryption and decryption, while non-symmetric keys use different keys for each (Bernstein & Cobb, 2021). Encryption has been traced back deep into our history, ever since people had things that needed to remain secret. Civilizations used symbols and keys to encrypt and decrypt messages. As the centuries progressed and war became ever present, computing and encryption became more sophisticated and more lifesaving. Today’s encryption, in a world built between networks, is now more important than ever and the presence of hackers is a threat that is always looming and must be always taken very seriously.

## Certificate Generation

Insert a screenshot below of the CER file.

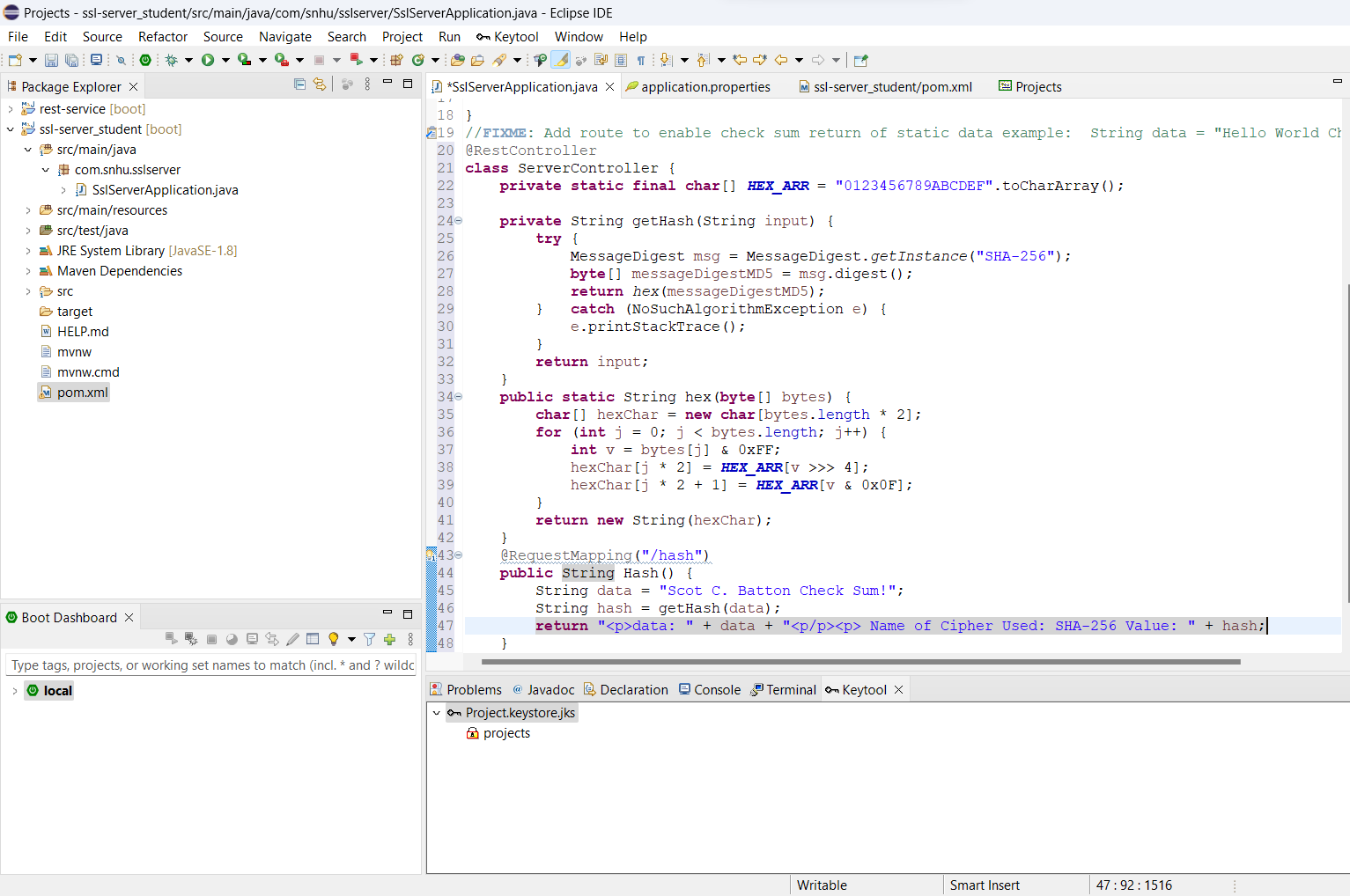




## Deploy Cipher

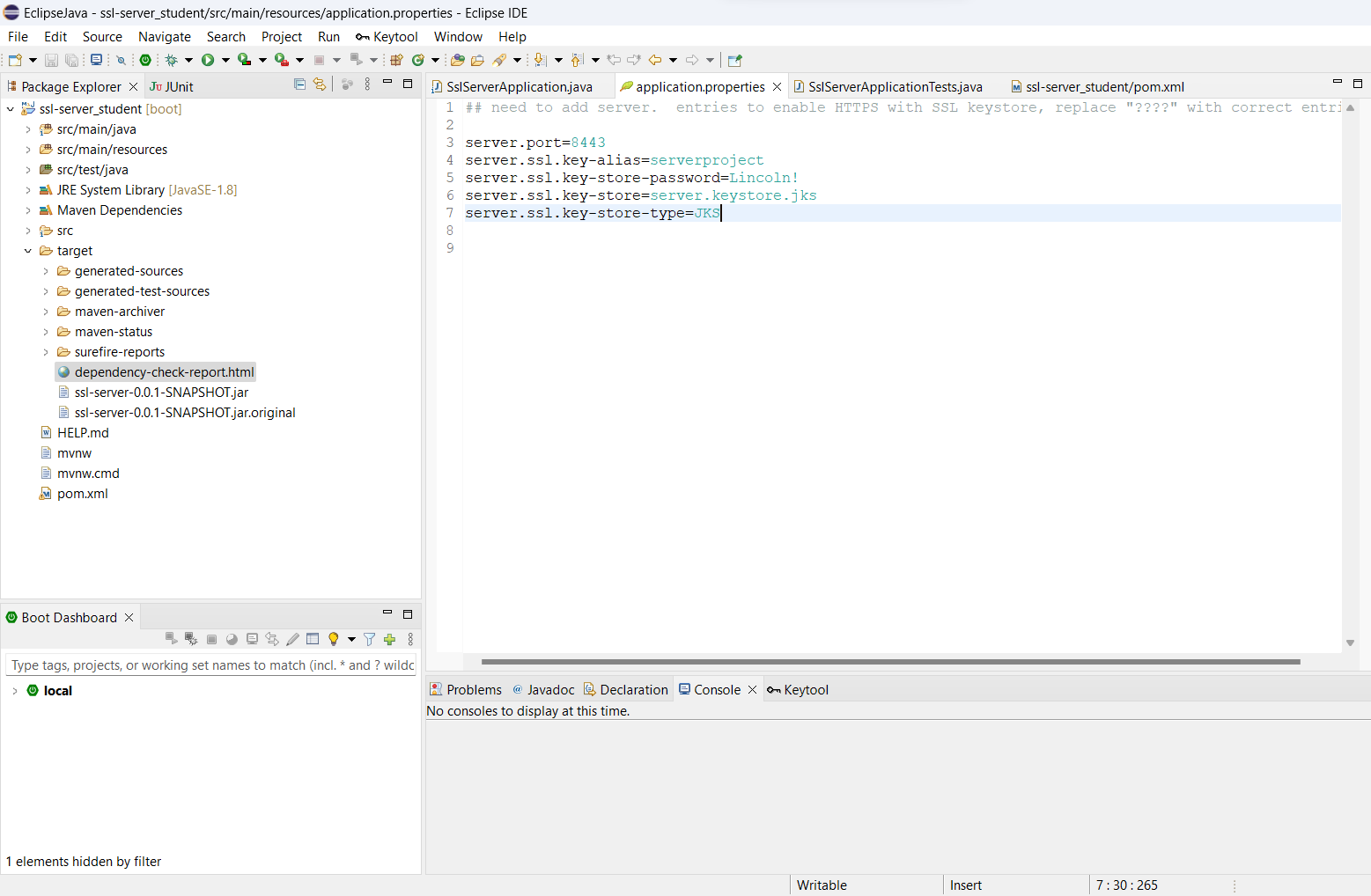
Insert a screenshot below of the checksum verification.

As with my secure communications below, I could not get the local host webpage to load and to show me any type of information. I have provided the code that I used below and do not have any further screenshots to show since I was unsuccessful in getting one.



## Secure Communications

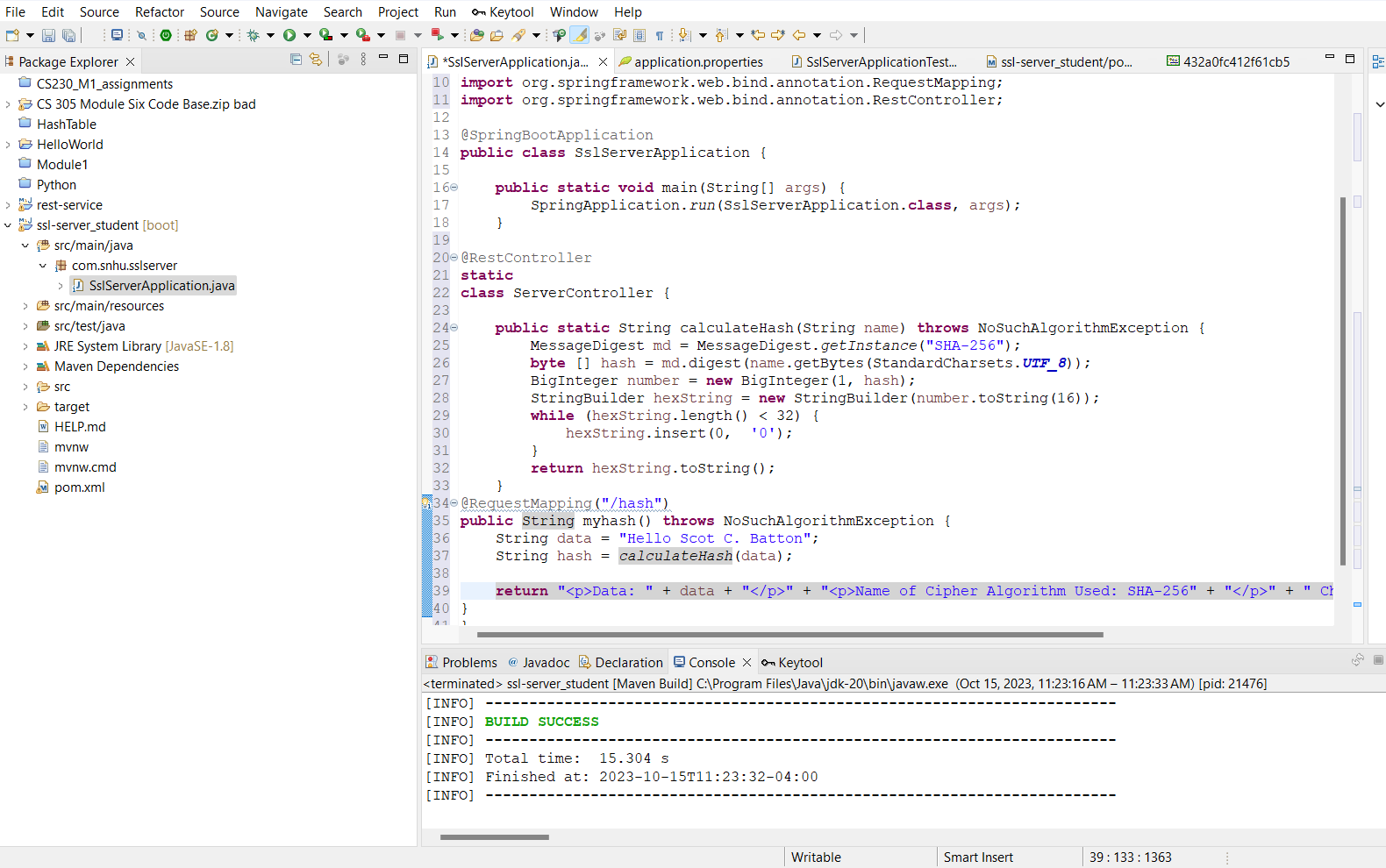
Insert a screenshot below of the web browser that shows a secure webpage.

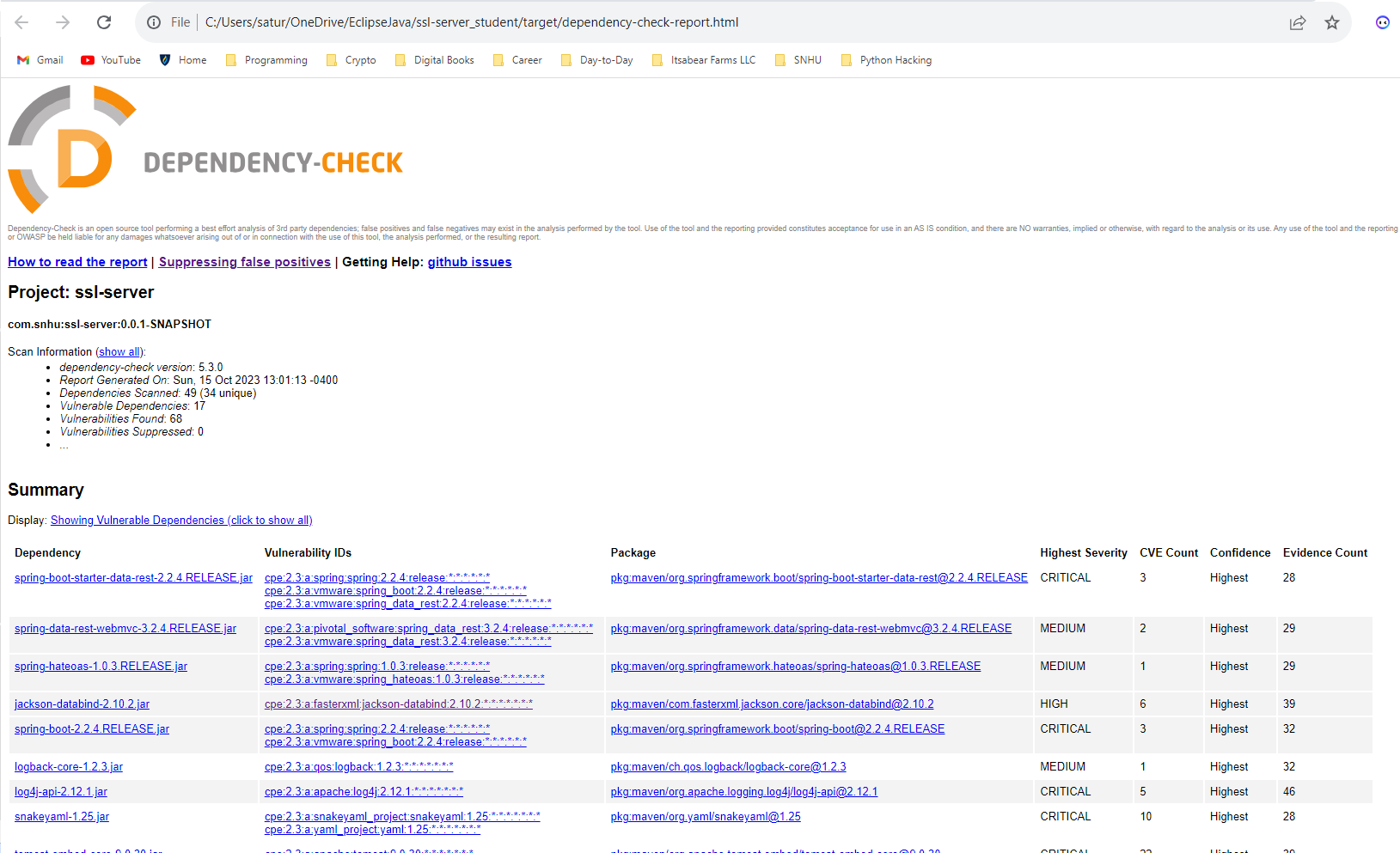


I refactored the code within the applications.properties file but I could not get the secure webpage to load. I tried changing settings within the computer and tried several versions of my code and could not reach a positive conclusion. Above is the code that I used in the applications.properties file so show the work that I was able to complete.

## Secondary Testing

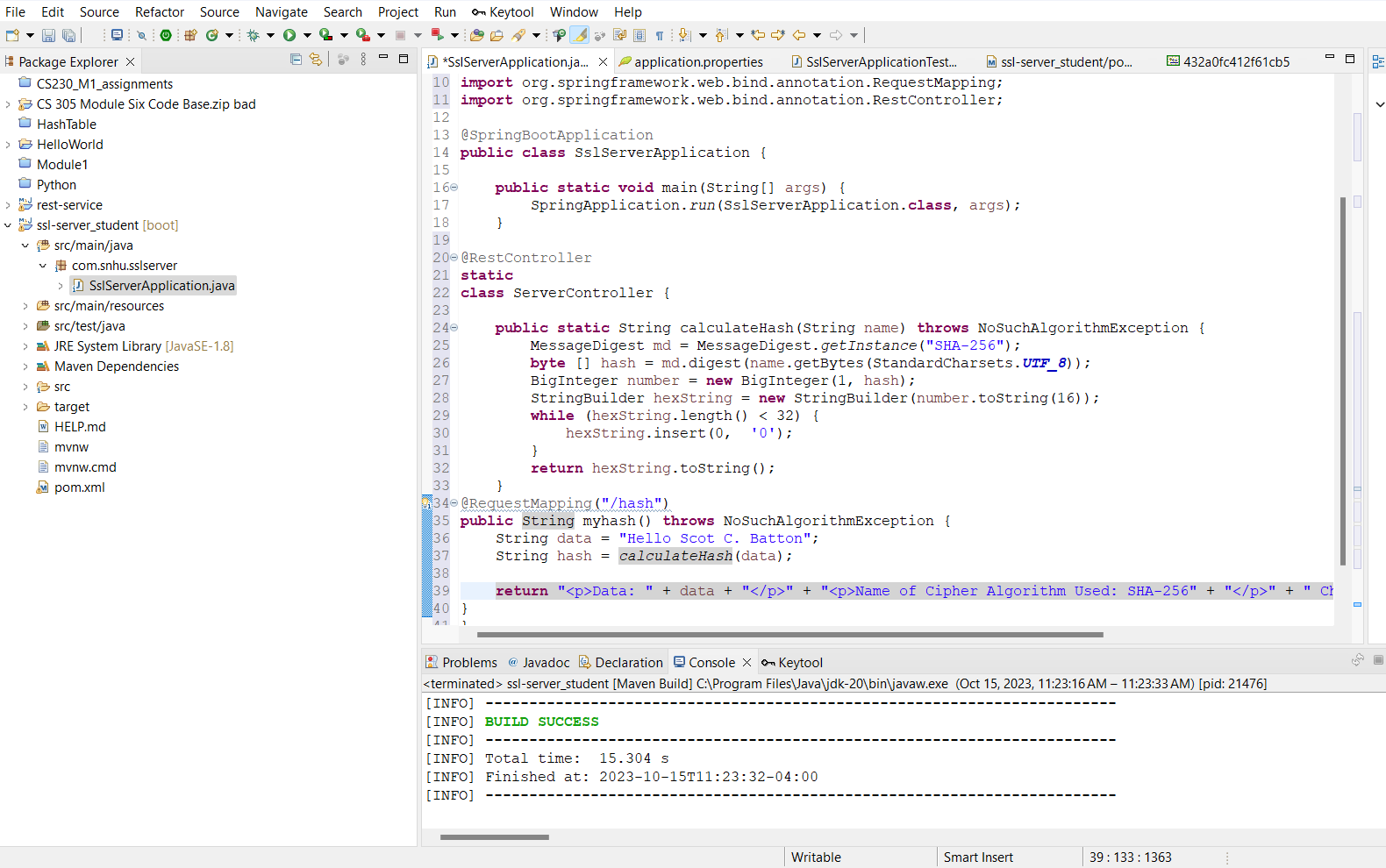
Insert screenshots below of the refactored code executed without errors and the dependency-check report.





## Functional Testing

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



## Summary

For this specific project, I had to refactor the code that was provided within the code base to provide secure software for Artemis Financial. I added a RestController to the file to be able to work as a secure controller for the API. The code that was added was very minimal and provided only what was needed to secure the information for the company and make a secure connection to the RESTful endpoint. I also provided @RequestMapping to make sure that the program will run properly when the hash is added. A checksum of the data was then performed, and the protocol was changed from http to https for security. A self-signed certificate was created and then exported. A dependency check was then run on the codebase and vulnerabilities and fixes were presented in the form of a dependency report. Within the code and throughout the company, vulnerabilities needed to be addressed. Input validation, secure authentication, and secure data storage all needed to be addressed to allow for financial information to remain private and for only certain personnel access to said information.

## Industry Standard Best Practices

The industry standard must be followed to the best of my ability when it comes to providing recommendations and solutions for Artemis Financial. I was able to use a strong and secure encryption when it comes to SHA-256 and its 256 bit encryption. The use of a certificate to secure my https connection provided another layer of security and is a practice widely used within the field. I finally was able to use many different levels of security practices to limit and stop vulnerabilities, such as input validation, secure authentication, and secure data storage. The proper use of industry best practices allow for continuity and security within the client patient relationship and also within the company itself.

*References*

Wikimedia Foundation. (2023, August 14). *Secure hash algorithms*. Wikipedia. https://en.wikipedia.org/wiki/Secure\_Hash\_Algorithms

Bernstein, C., & Cobb, M. (2021, September 24). *What is the Advanced Encryption Standard (AES)? definition from searchsecurity*. Security. https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard