****

# Practices for Secure Software Report

**Table of Contents**

[**Document Revision History 3**](#_heading=h.30j0zll)

[**Client 3**](#_heading=h.1fob9te)

[**Instructions 3**](#_heading=h.3znysh7)

[**Developer 4**](#_heading=h.2et92p0)

[**1. Algorithm Cipher 4**](#_heading=h.tyjcwt)

[**2. Certificate Generation 4**](#_heading=h.3dy6vkm)

[**3. Deploy Cipher 4**](#_heading=h.1t3h5sf)

[**4. Secure Communications 4**](#_heading=h.4d34og8)

[**5. Secondary Testing 4**](#_heading=h.2s8eyo1)

[**6. Functional Testing 4**](#_heading=h.17dp8vu)

[**7. Summary 4**](#_heading=h.3rdcrjn)

[**8. Industry Standard Best Practices 4**](#_heading=h.26in1rg)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **2/25/2024** | **Joseph Dengler** |  |

## Client



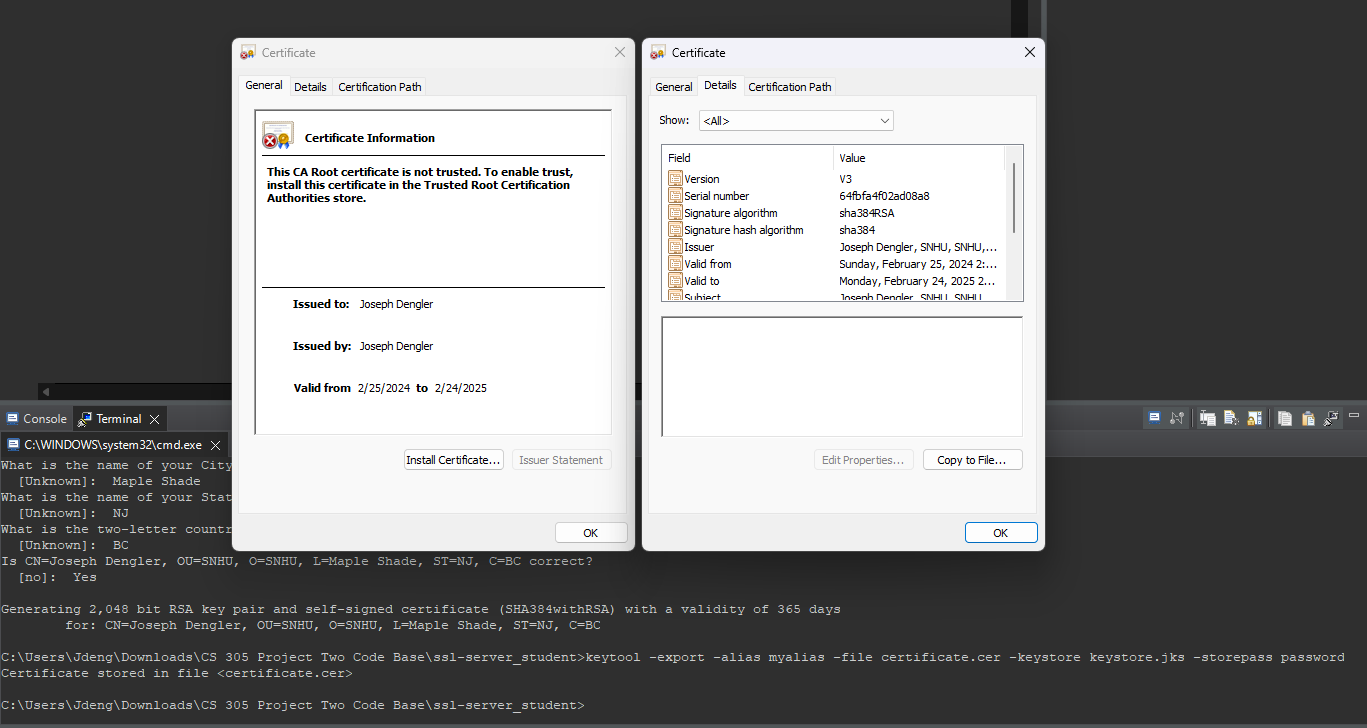
## Developer

Joseph Dengler

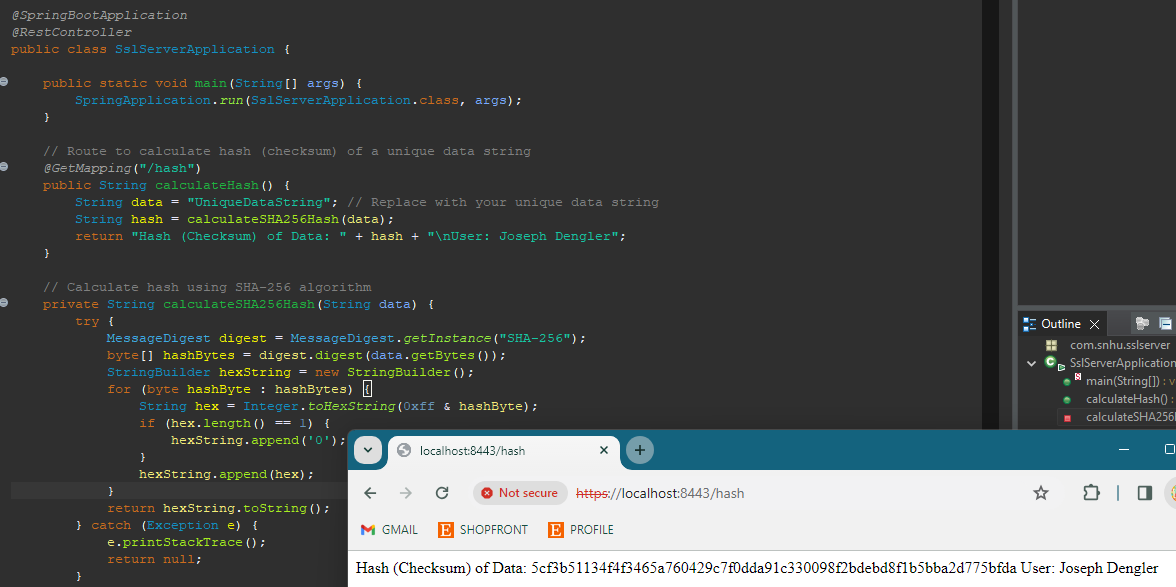
## Algorithm Cipher

For Artemis Financial's web application, it's crucial to choose an encryption algorithm cipher that provides robust security while being computationally efficient. Considering the security vulnerabilities and the nature of the project, SHA-256 is recommended as an appropriate encryption algorithm cipher. SHA-256 is a cryptographic hash function that generates a 256-bit hash value from input data, providing strong resistance against collision attacks. It does not involve random numbers and operates deterministically on the input data, making it suitable for data integrity verification. While it falls under symmetric cryptography and does not use keys for encryption, its historical track record and widespread adoption in various cryptographic applications underscore its reliability and security. Overall, SHA-256 offers robust protection against data tampering and unauthorized modifications, making it a suitable choice for ensuring the integrity of data in the project.

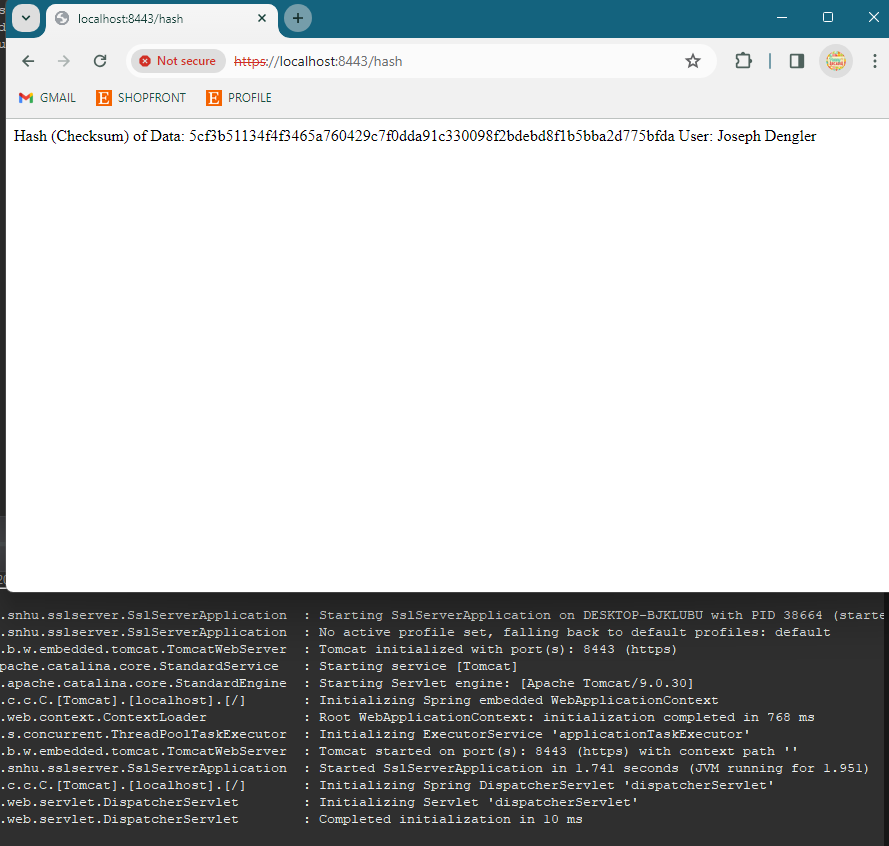
## Certificate Generation



## Deploy Cipher

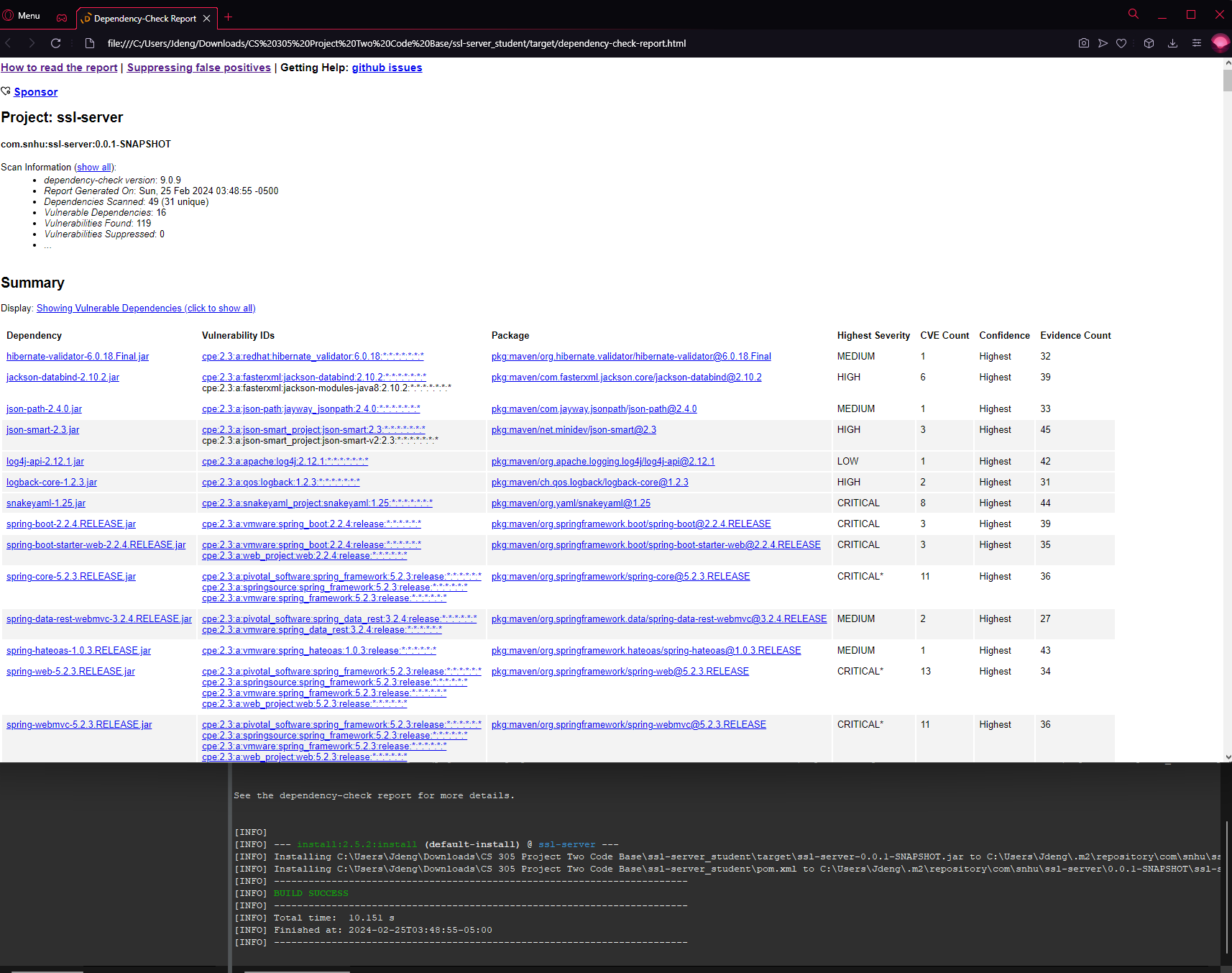


## Secure Communications

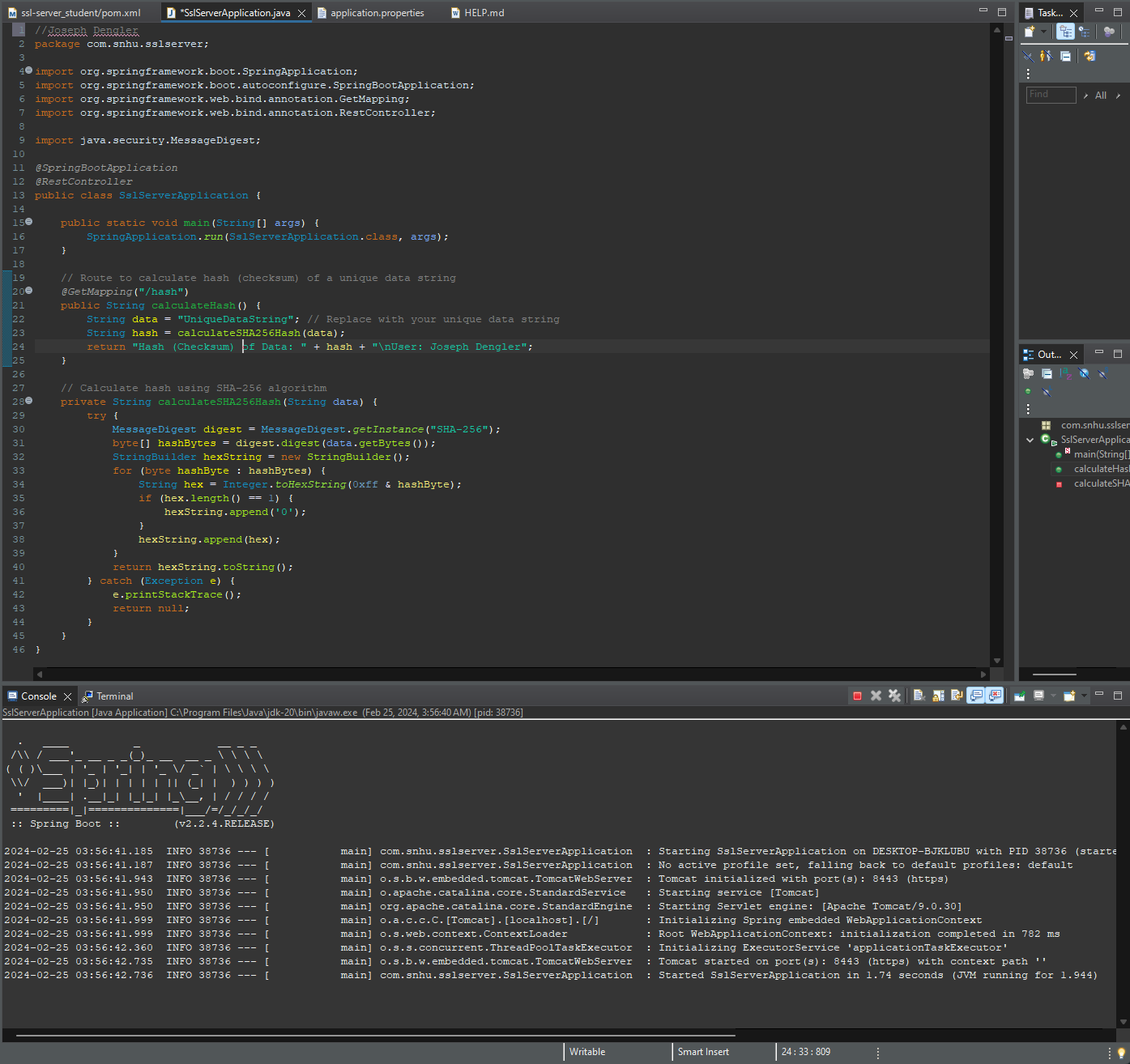


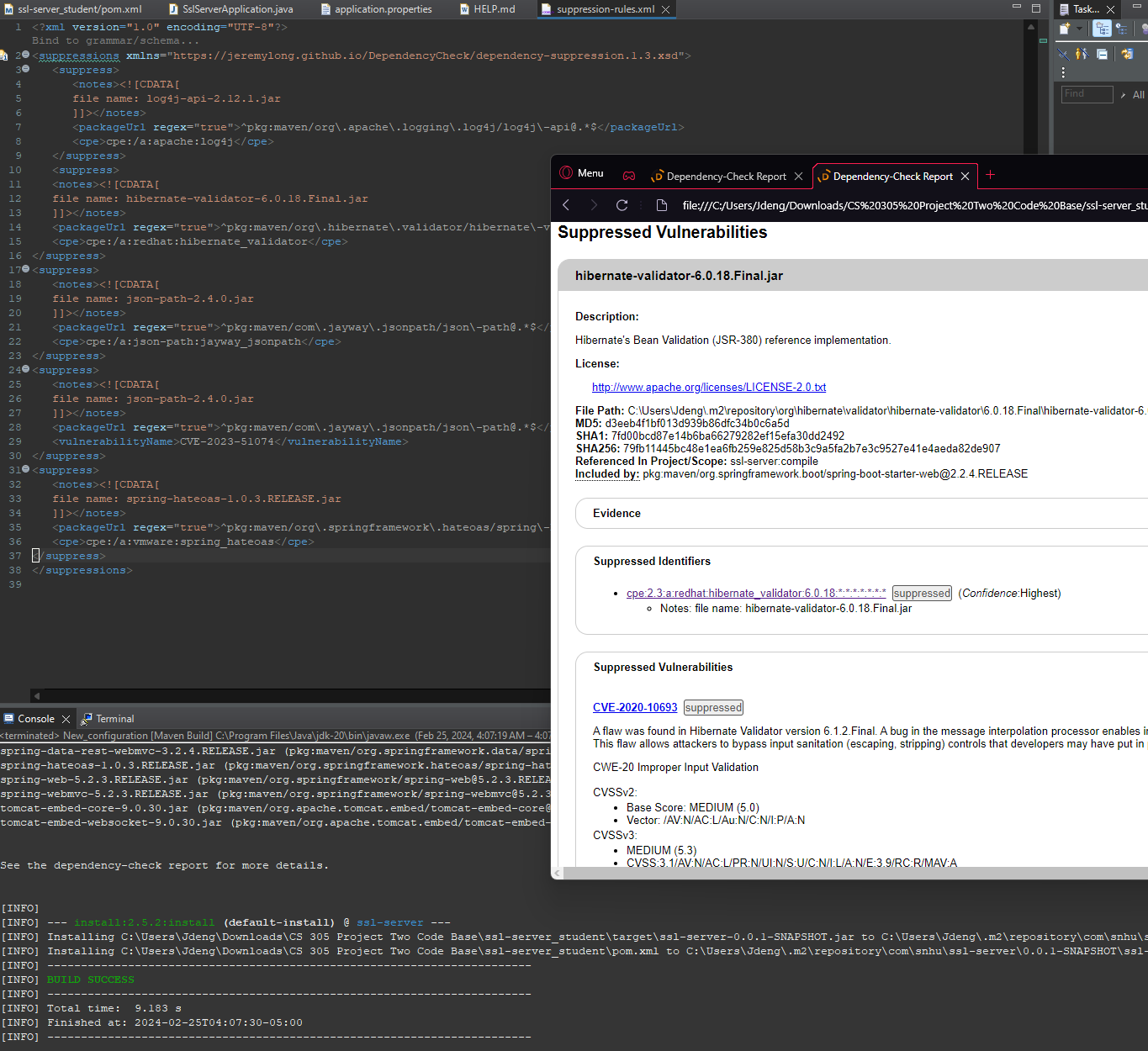
(It is running as HTTPS but my chrome keeps saying not secure, I tried adding the certificate and other fixes with no luck, seems to just be an issue with my chrome)

## Secondary Testing



## Functional Testing (Code running with no errors and suppressed vulnerabilities)





## Summary

The code has undergone significant refactoring to address security vulnerabilities and comply with security testing protocols. By following the Vulnerability Assessment Process Flow Diagram, several key areas of security have been addressed. Initially, an appropriate encryption algorithm cipher, SHA-256, was recommended and implemented to ensure data integrity. Certificate generation was then performed to establish secure communications using HTTPS protocol, although there were some challenges with browser compatibility. Additionally, the cryptographic hash algorithm was deployed, and secure communication was verified. Secondary testing with OWASP Dependency-Check Maven was conducted to ensure the absence of additional security vulnerabilities. Finally, false vulnerabilities were suppressed in the dependency report. Through these steps, layers of security have been added to the software application, encompassing input validation, secure API interactions, encryption use, secure code handling, and more, as outlined in the Vulnerability Assessment Process Flow Diagram.

## Industry Standard Best Practices

Throughout the development process, industry standard best practices for secure coding were diligently applied to mitigate known security vulnerabilities and ensure the ongoing security of the software application. These practices encompassed various aspects, including input validation, secure communication protocols, encryption algorithms, and dependency management. Specifically, input validation techniques were implemented to sanitize and validate user inputs, mitigating risks associated with injection attacks and malicious data manipulation. Furthermore, the adoption of secure communication protocols such as HTTPS facilitated the secure transmission of sensitive data over the network, safeguarding against eavesdropping and data interception. The utilization of strong encryption algorithms like SHA-256 bolstered data integrity and confidentiality, providing robust protection against unauthorized access and tampering.