****

# 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** | **Oct 20, 2024** | **Mikayla-Joy Botha** |  |

## 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

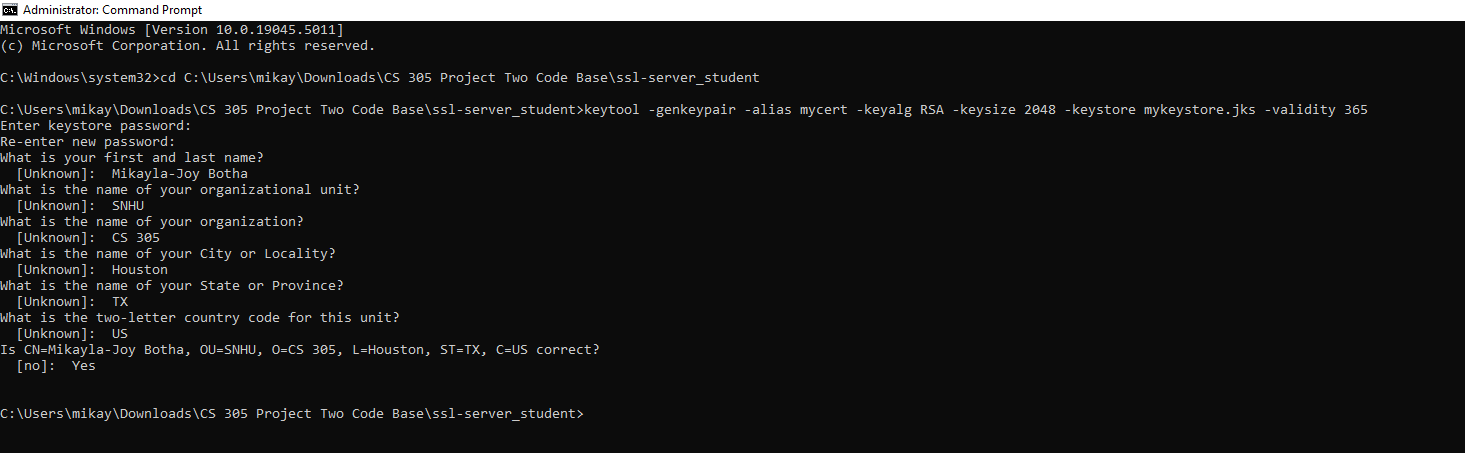
Mikayla-Joy Botha

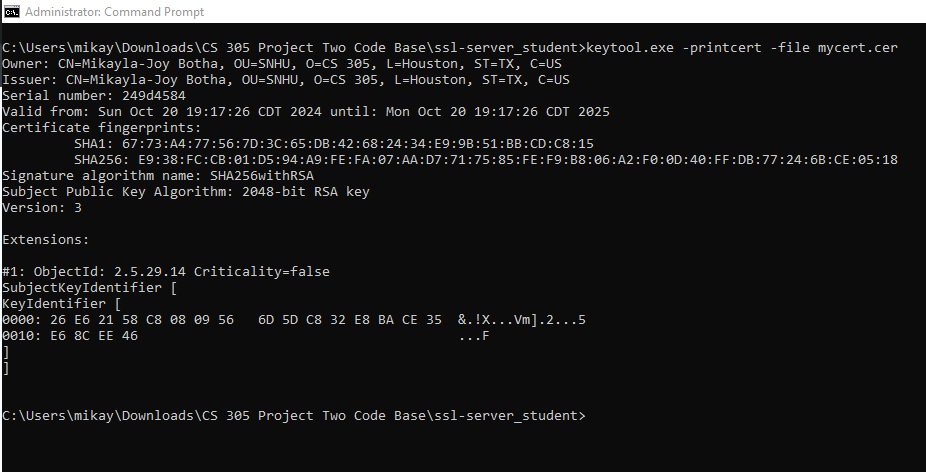
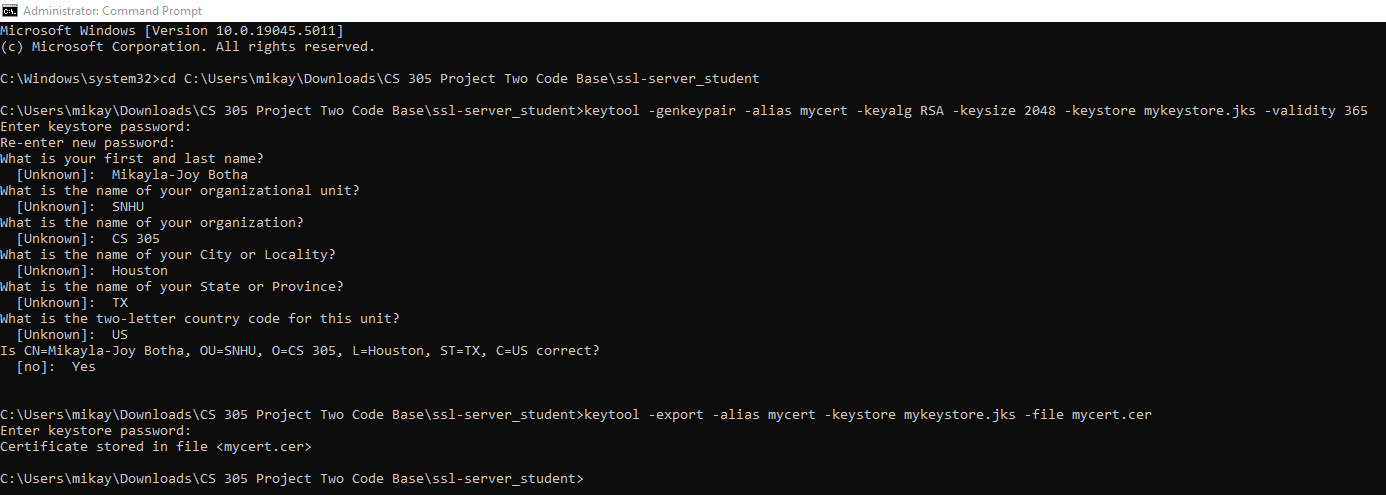
## Algorithm Cipher

For this project I utilized the SHA-256 encryption algorithm cipher for this project, because it is a very secure due option. While this type of algorithm cipher can be kind of ridgid, it has stood the test of time and proven itself very reliable. The abbreviation SHA-256 is actually “stands for Secure Hashing Algorithm, and it is a family of [cryptographic](https://nordvpn.com/blog/what-is-cryptography/) hash functions developed by the National Security Agency (NSA) in the United States. A hash function, in simple terms, is a mathematical algorithm that takes an input (any data, like a file or a password) and produces a fixed-size string of characters, which is the hash value or digest. But keep in mind that it is not encryption. “ (NordVPN, n.d.). This system relies on random factors for generating keys in order to distinguish the difference between single key and key pair encryption. From early times, algorithms started from simple methods like the Cesar cipher and have now blossomed into what we know today as more complex ones such as SHA-256.

## Certificate Generation

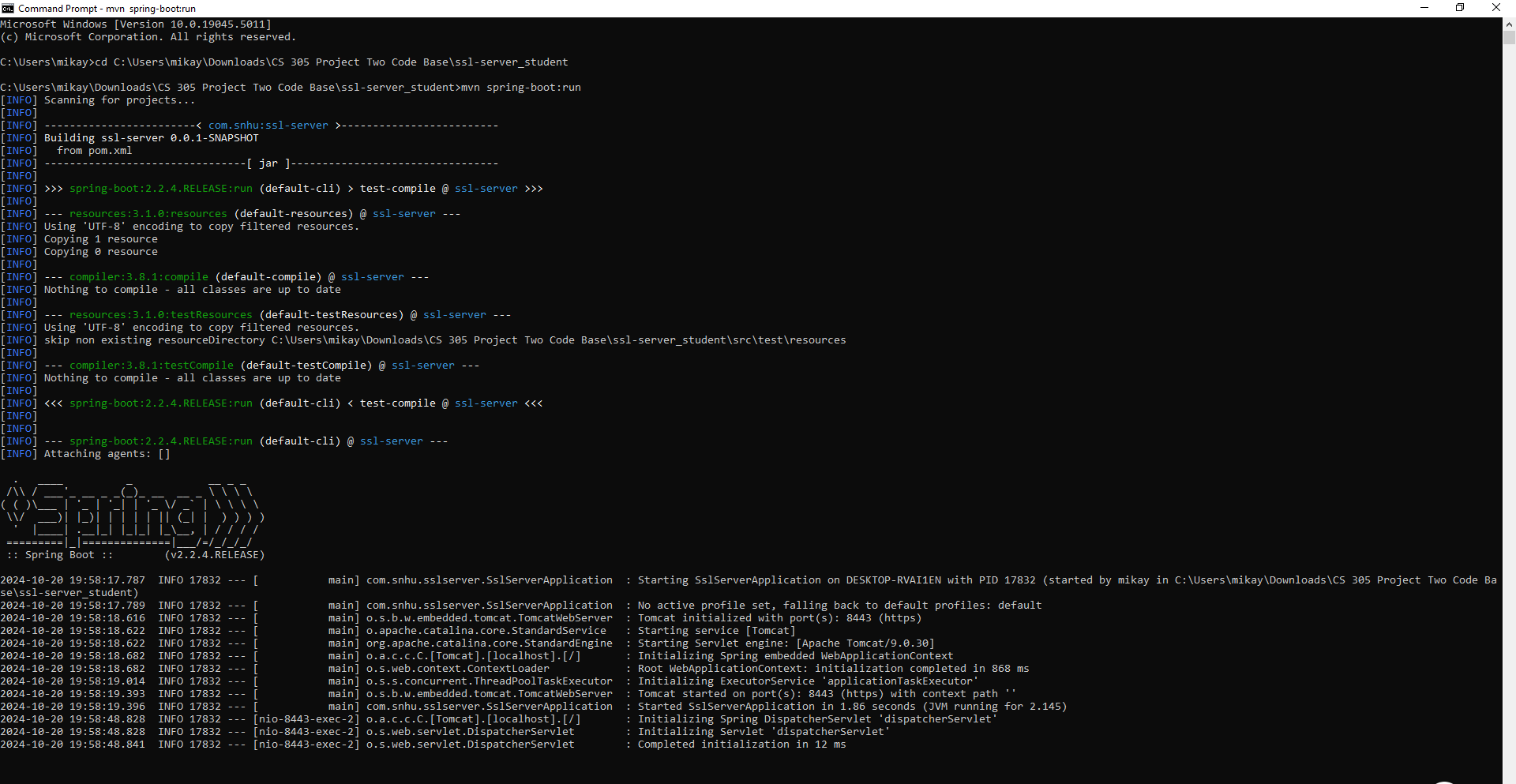
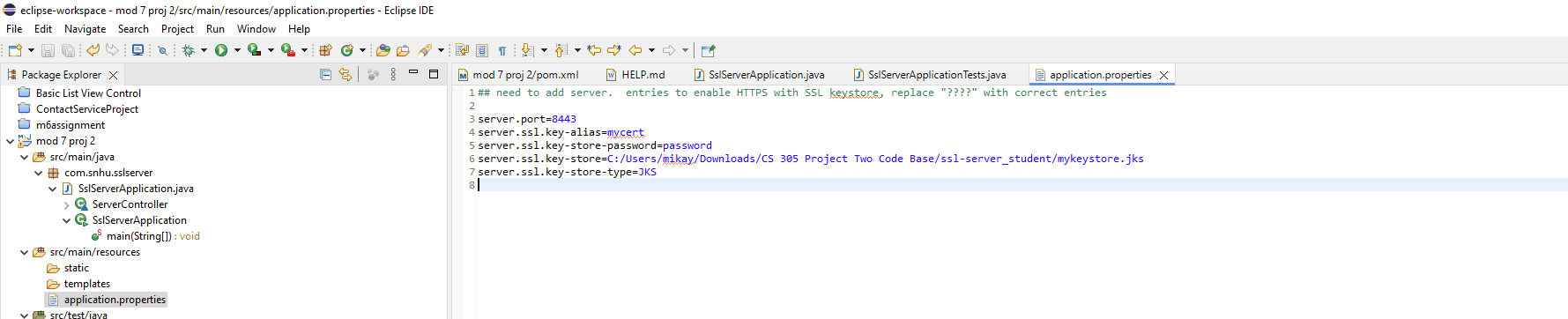
Insert a screenshot below of the CER file.Generate appropriate self-signed certificates using the Java Keytool in Eclipse. Export your certificates as a CER file.Export your certificates as a CER file

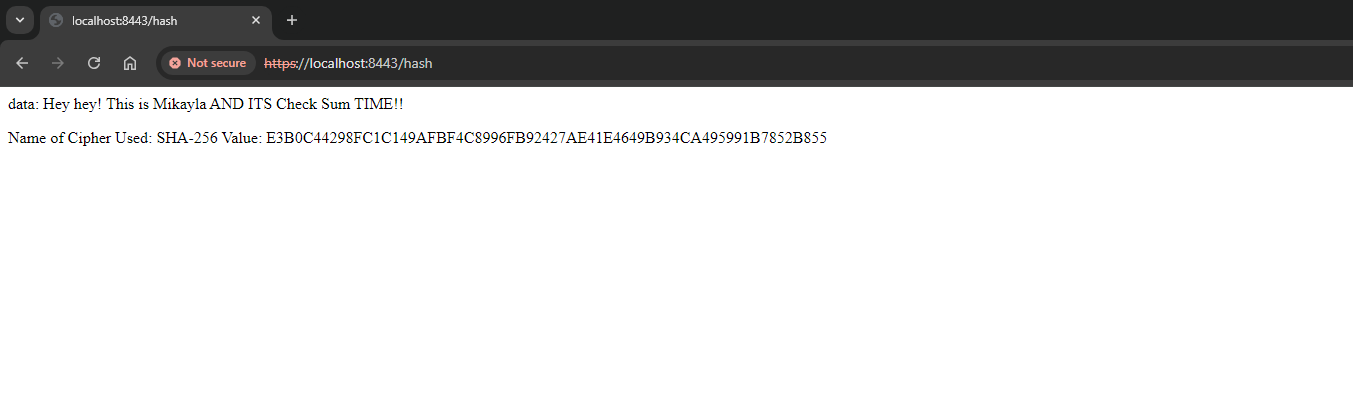




## Deploy Cipher

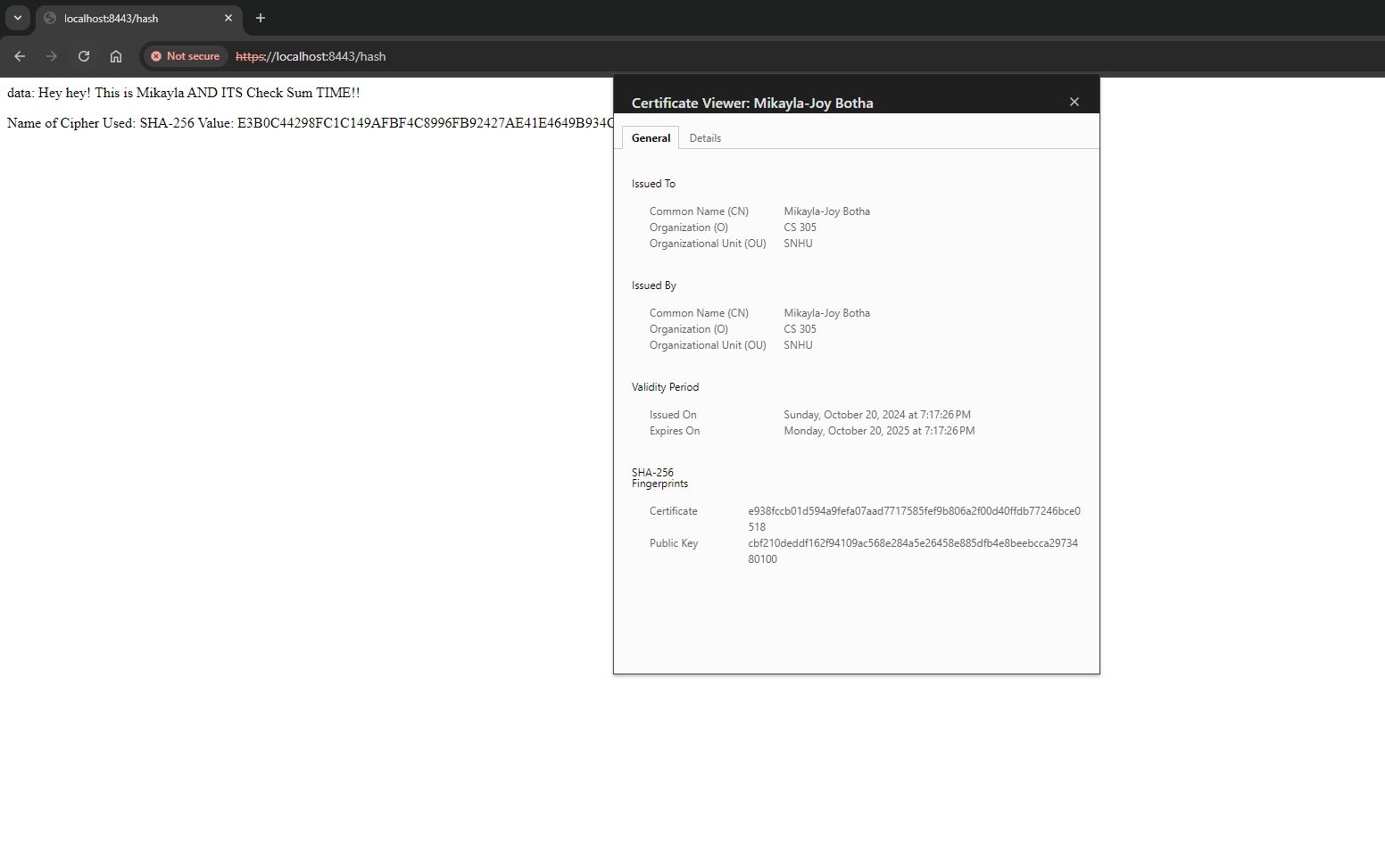
Insert a screenshot below of the checksum verification. Deploy and implement the cryptographic hash algorithm by refactoring code. Demonstrate functionality with a checksum verification.The screenshot must show your name and a unique data string that has been created.





## Secure Communications

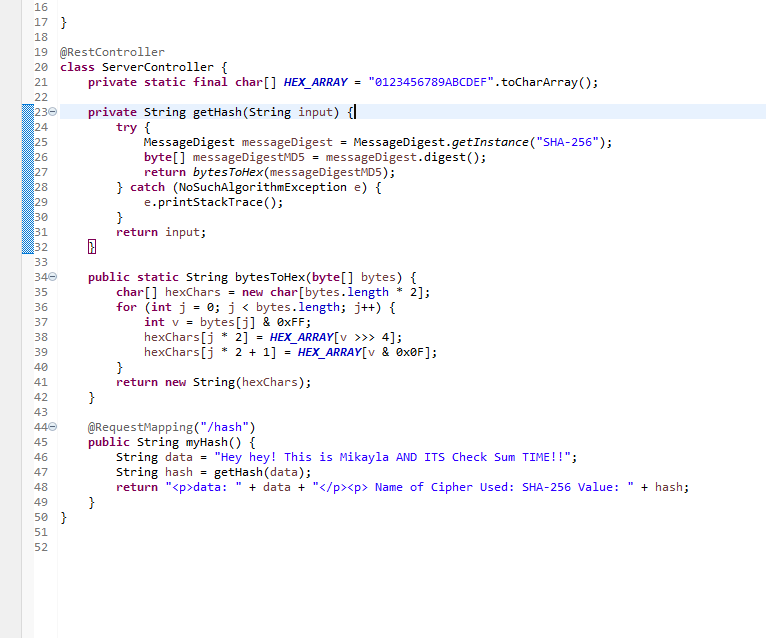
Insert a screenshot below of the web browser that shows a secure webpage.



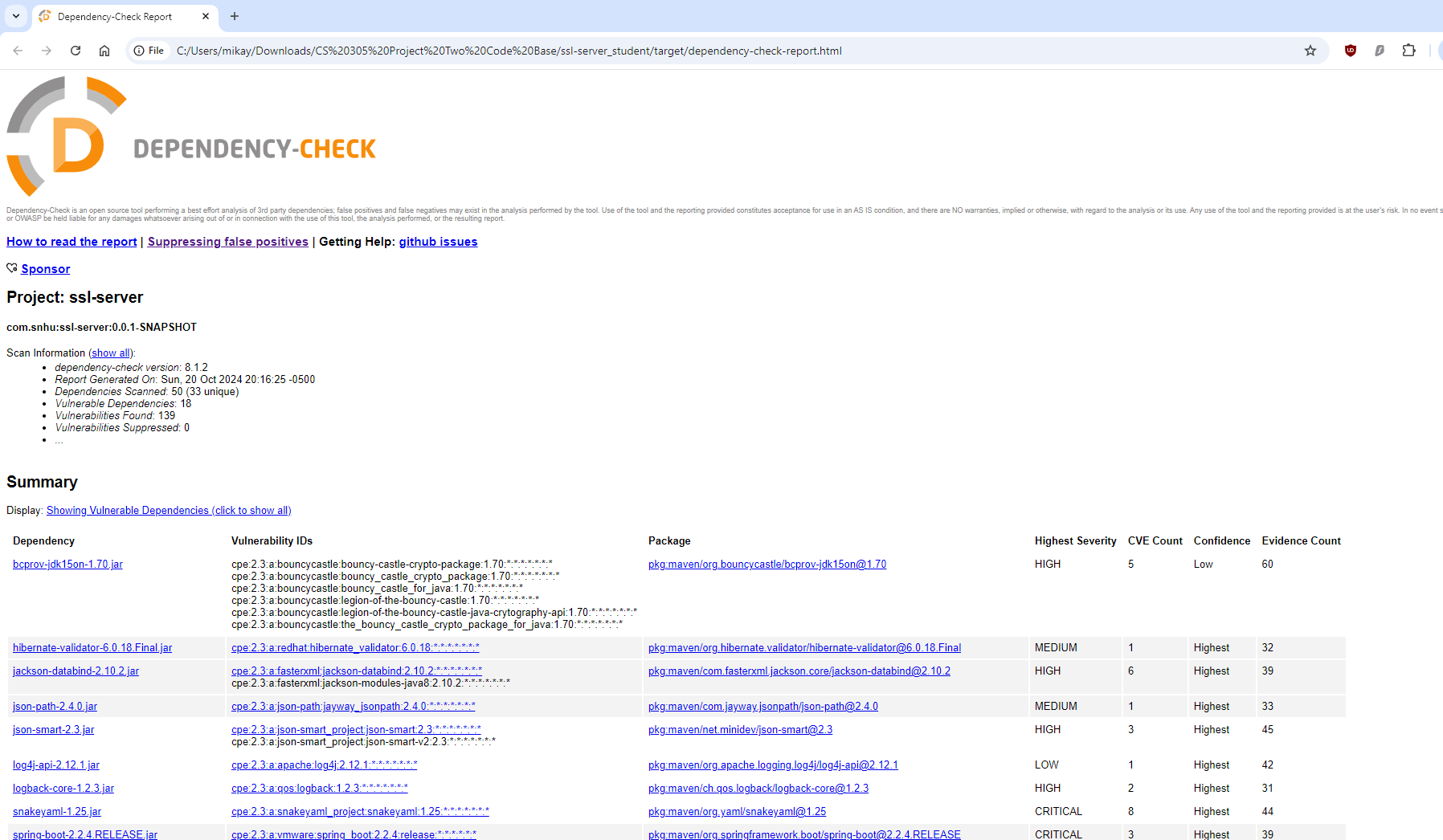
## Secondary Testing

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

Refactored Code:

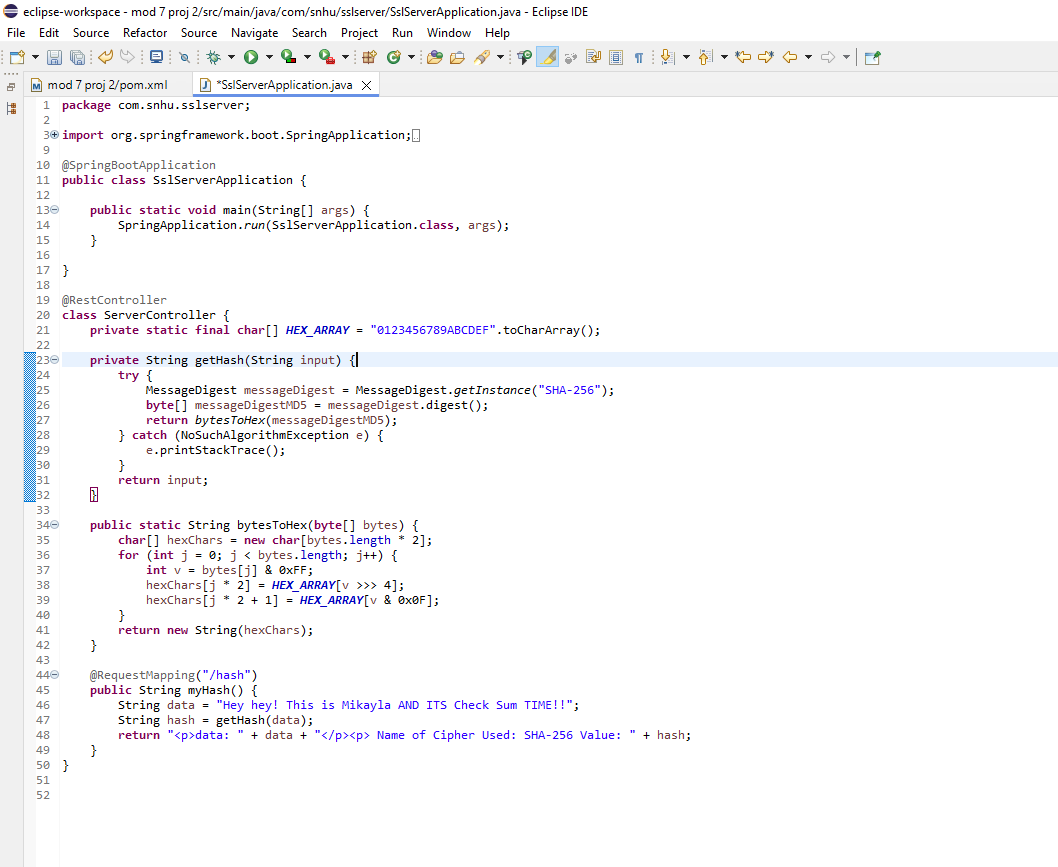


Dependency-check report:



## Functional Testing

Insert a screenshot below of the refactored code executed without errors. Identify the software application's syntactical, logical, and security vulnerabilities by manually reviewing the code.



## Summary

In short, I refactored this code to convert from HTTP to HTTPS, making the communication between servers and clients secure. As a result of the refactoring process, this helped to mitigate potential security concerns like the encryption and integrity of data through the implementation of SSL and TLS due to the self-signed certificates created. From this, having done all these things, it guaranteed that the transmission of data would be secure and protocols will be set in place.

## Industry Standard Best Practices

Ultimately, I applied industry standard best practice such as input validation, and secure algorithms like SHA 256 for hashing data to maintain the software application’s existing security. By applying industry standard best practices for secure coding, we can avoid risks such as security breaches to secure a company’s overall well-being.

Citation:

NordVPN. (n.d.). *SHA-256: What is it and how does it work?*<https://nordvpn.com/blog/sha-256/>