****

# 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** | **04/17/2024** | **Ethan M.E. Mills** |  |

## 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

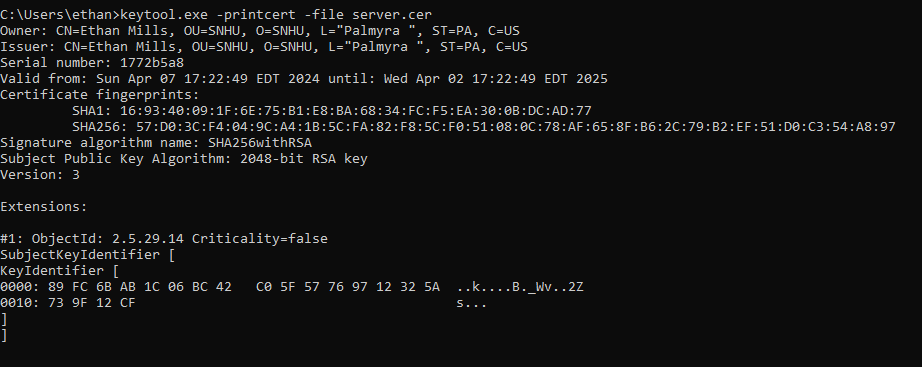
Ethan Marc Emerson Mills

## Algorithm Cipher

* Recommend an appropriate encryption algorithm cipher to deploy, given the security vulnerabilities, and justify your reasoning. Review the scenario and the supporting materials to support your recommendation. In your practices for secure software report, be sure to address the following:
  + Provide a brief, high-level overview of the encryption algorithm cipher.
  + Discuss the hash functions and bit levels of the cipher.
  + Explain the use of random numbers, symmetric versus non-symmetric keys, and so on.
  + Describe the history and current state of encryption algorithms.

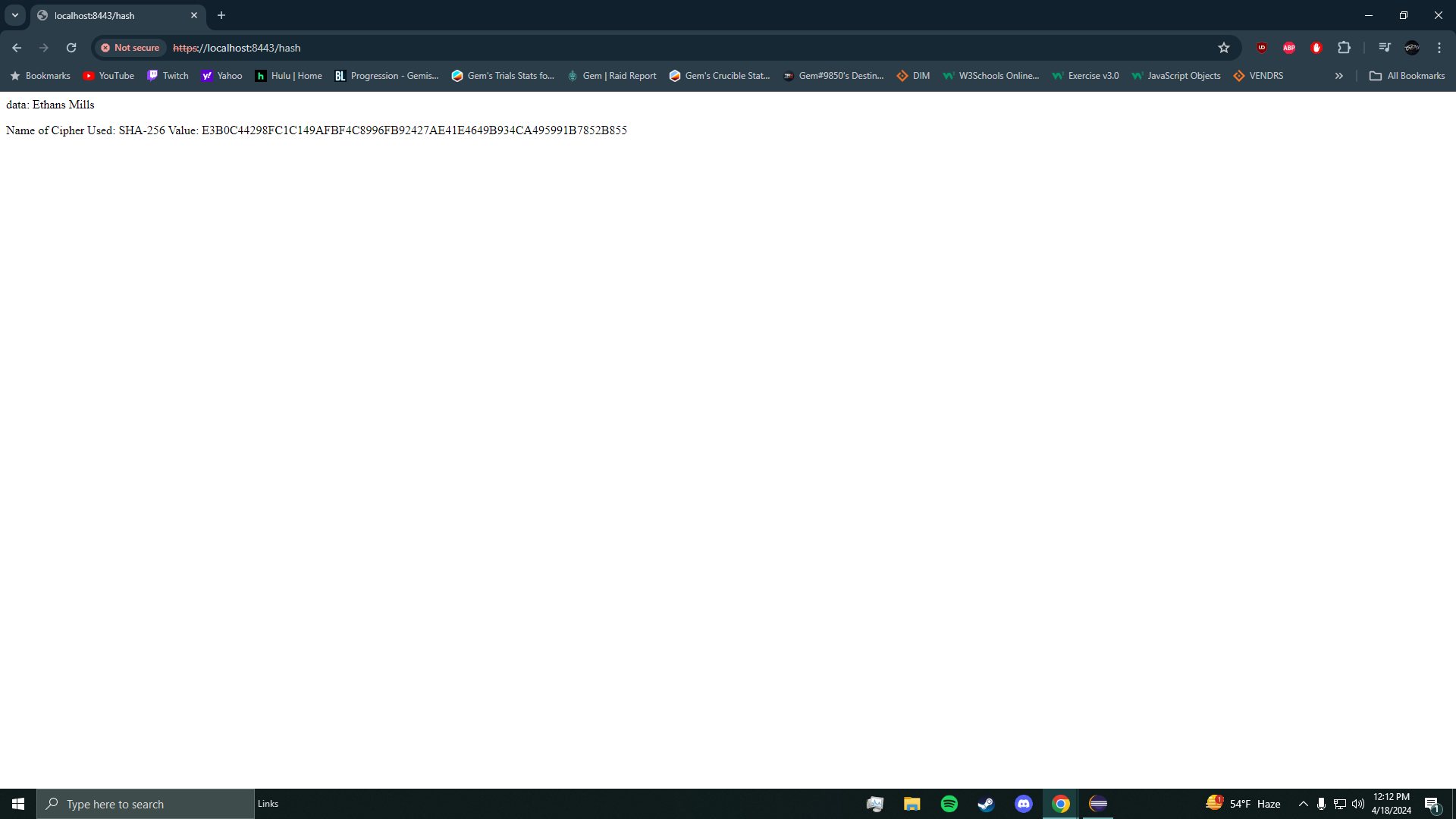
The Encryption algorithm cipher that I will be recommending for this project is SHA-256.SHA-256, a successor of the hash function sha-1 and is one of the strongest hash functions available. SHA-256 has never been compromised in any way. The SHA-256 algorithm is also highly resistant to both cryptographic attacks and collisions. The 256 bit key also makes a good partner function for AES. Even if there are the smallest changes in the data, it will produce different checksums. SHA has been used to protect passwords for decades. SHA 256 usage ranges from protecting blockchain wallets to securing passwords for customers and digital signatures. A brief overview of the SHA-256 is as follows: It takes input data from an input message and then processes it into blocks of 512 bits. Next is padding, message padding is when the total length of a padded message is a multiple of 512, it also ensures it aligns with the block size. Next is message schedule, message schedule is padded data that is divided into 512 bit blocks and each block goes through a series of rounds. The purpose of a message schedule is essential for security by showing that it is possible to find collisions and with complexity 264 hash operations for a variant without it. Next is compressing the function. SHA compression is when the SHA-256 function takes a 512-bit message block and a 256-bit intermediate has value as an input and produces a 256-bit new intermediate has value as an output. Last is the final hash, the final hash is after all blocks have been processed, the intermediate hash values are then combined to produce a final 256-bit hash value. When it comes to avoiding collisions one way to do so is to reduce the load factor on the hash table, to do this you can choose a large enough hash table size and a good hash function. It is good practice to reduce collisions as much as possible to ensure that a malicious user cannot take advantage, copy, access or alter data.

## Certificate Generation

Insert a screenshot below of the CER file.

[Insert screenshots here.]

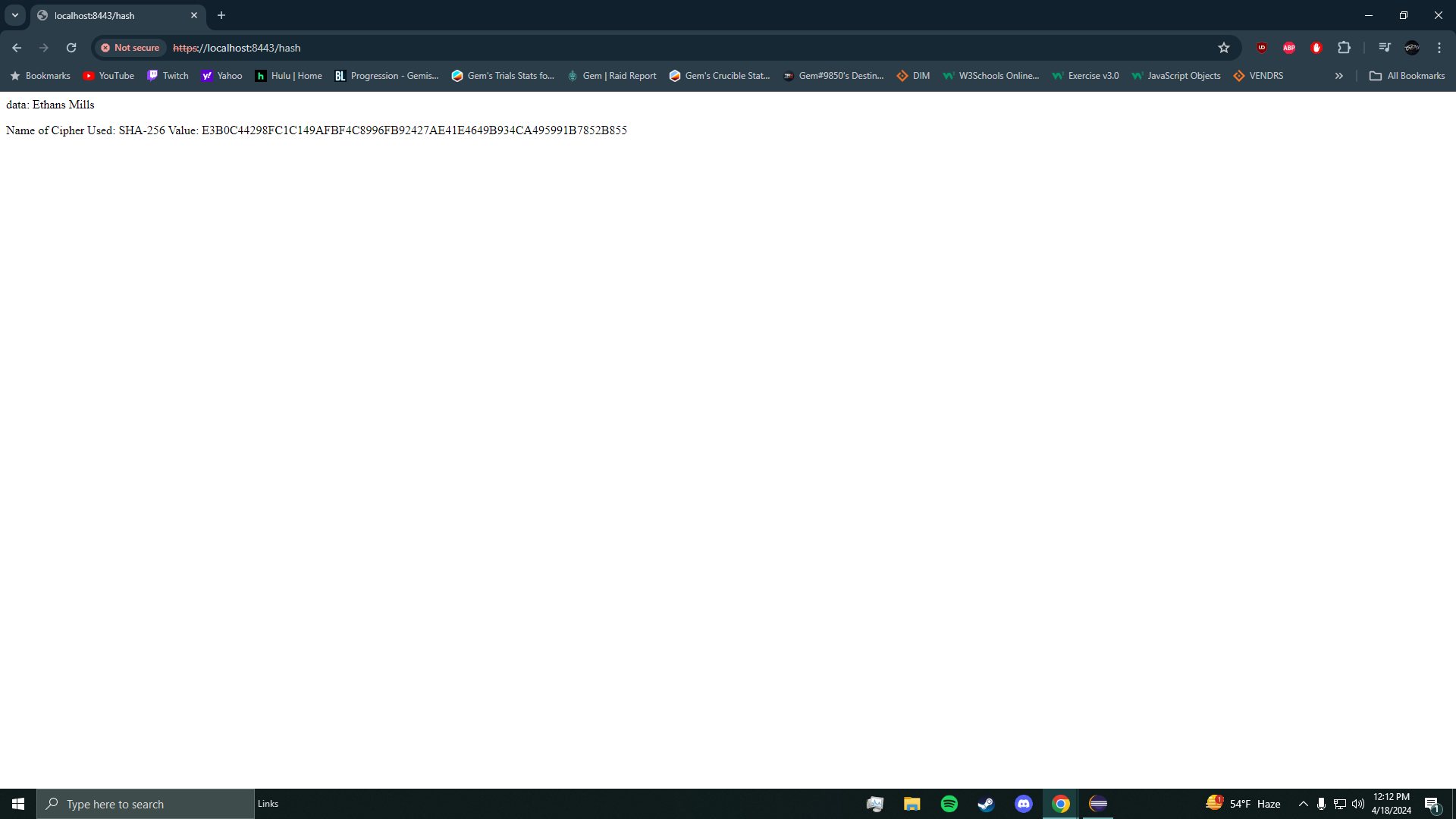
## Deploy Cipher

Insert a screenshot below of the checksum verification.

[Insert screenshots here.]

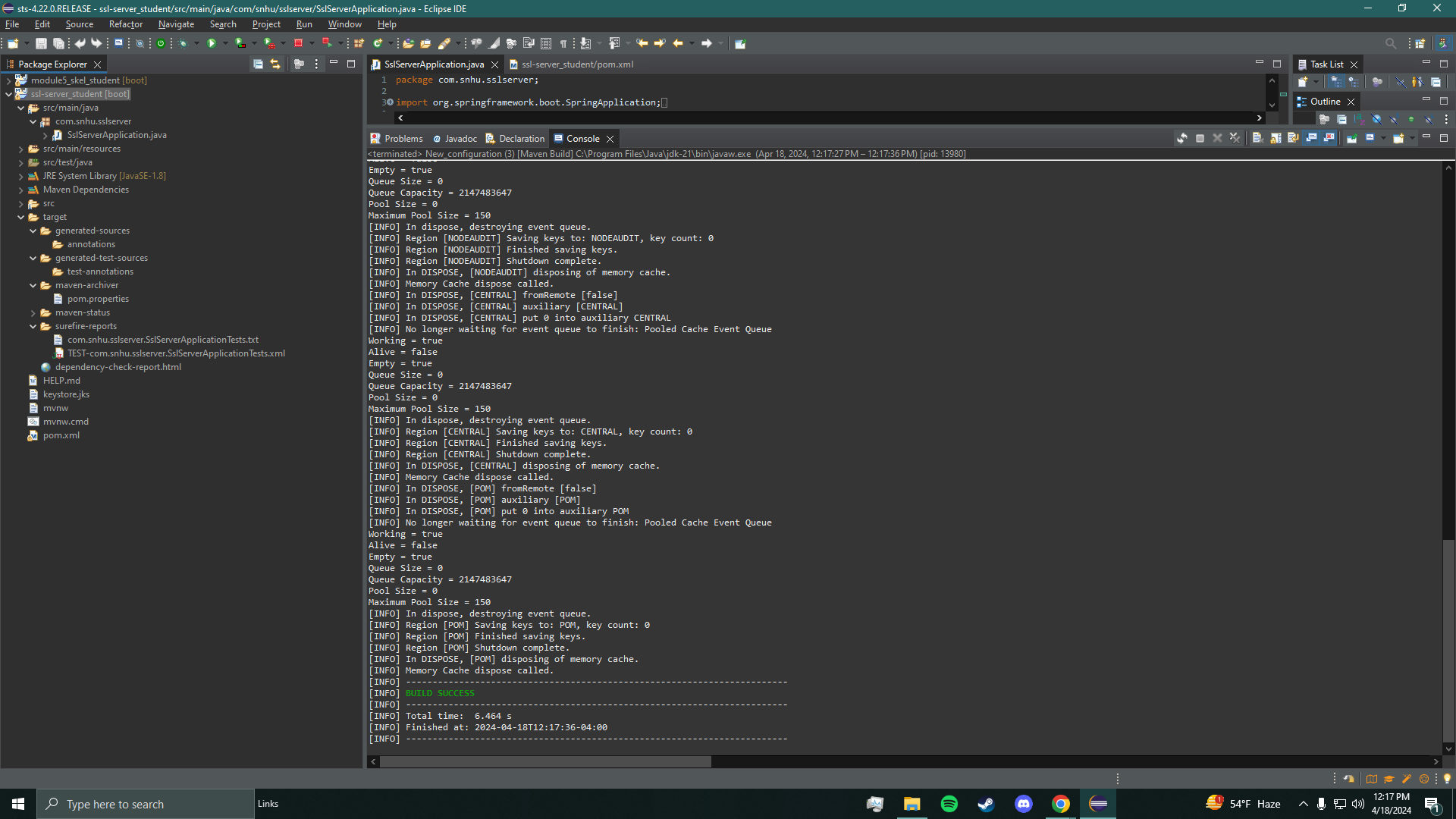
## Secure Communications

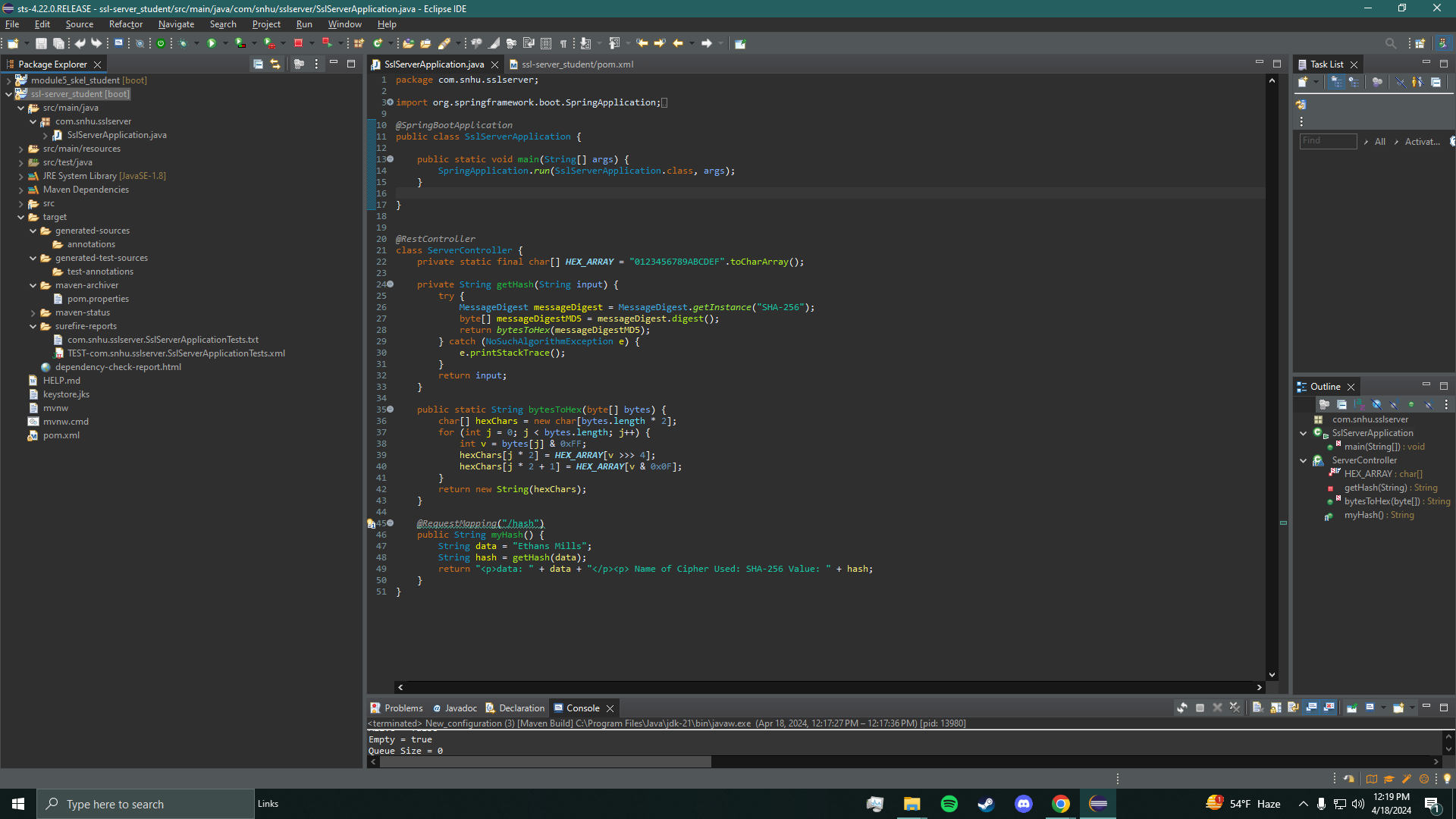
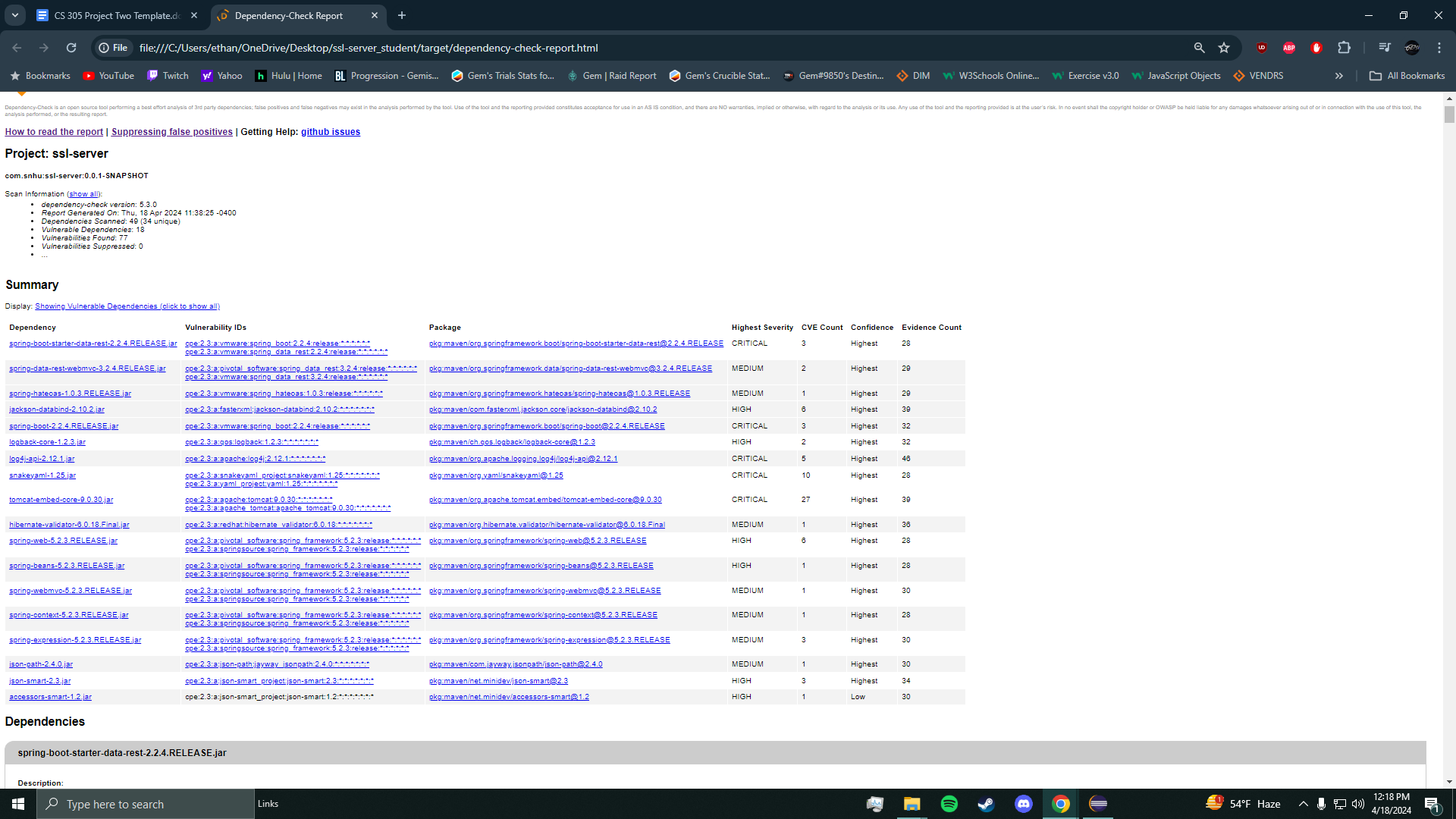
Insert a screenshot below of the web browser that shows a secure webpage.

[Insert screenshots here.]

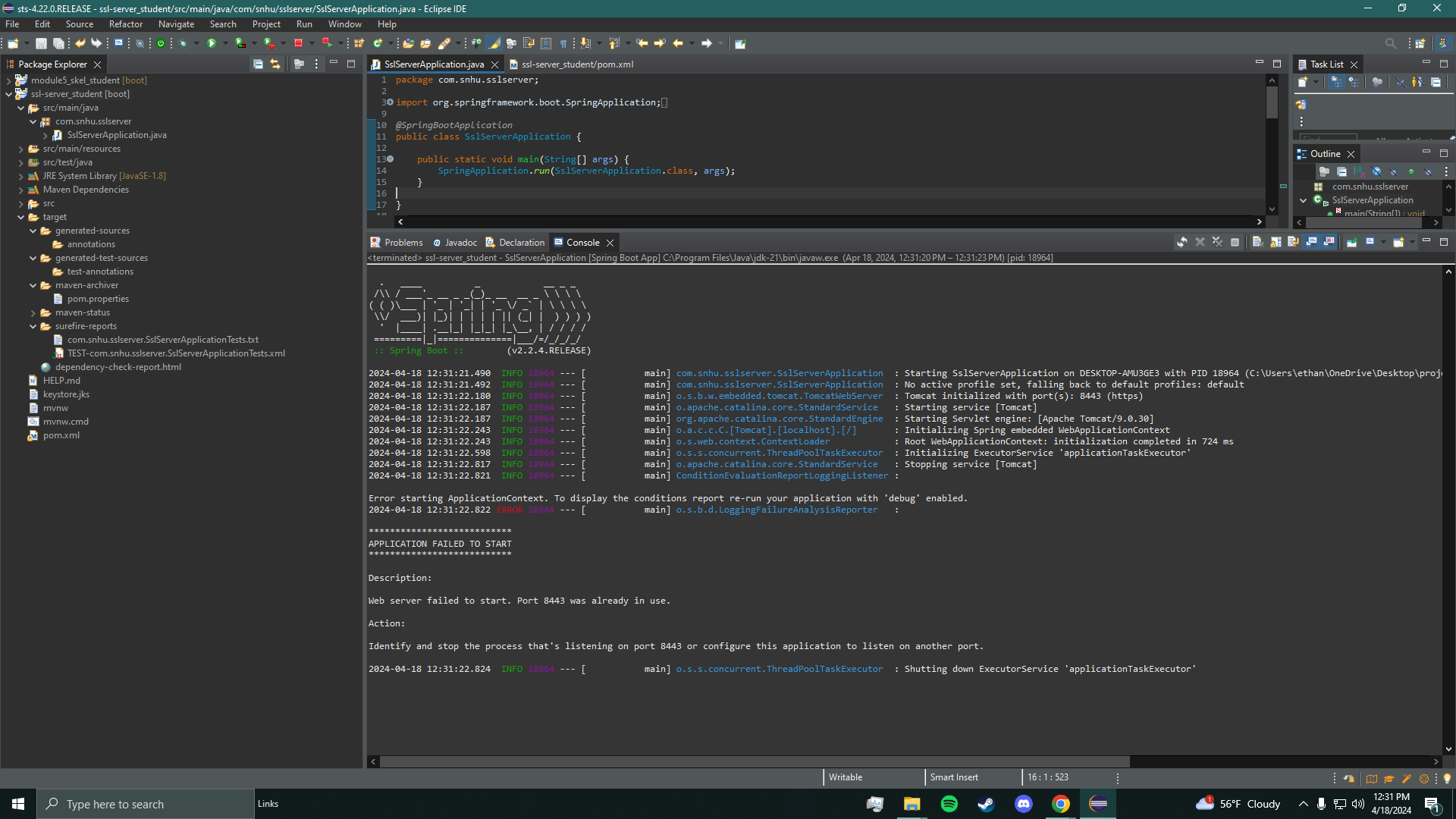
## Secondary Testing

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

[Insert screenshots here.]



## Functional Testing

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

[Insert screenshots here.]

## Summary

I added a secured restController to work as a secure controller. The way this works is the server controller class matches problems shown through the VAD(vulnerability assessment diagram). As I said before I have gone with the SHA-256 as it is one of the most secure hashing ciphers you can use and is one of the most used. To be able to maintain the best level of security it would be best to run bi weekly or max monthly dependency checks to stay on top of updates and vulnerabilities. If we keep the plugins in the pom.xml file this allows for higher security.

## Industry Standard Best Practices

To use the industry's standards best practices to maintain software applications current security I utilized the SHA-256 encryption algorithm. We want to protect people's personal and sensitive information so the use of SHA-256 I thought would be the best option. I also used restful apis to secure the exchange of that information. Lastly using the best practice for security integrated HTTPS, this encrypts data during transmission. This makes sure that said data that is to be shared goes through the correct process and secure protocols. As someone who aims to work in the software engineering field, I know how important it is to make sure that the work you do its as best as it can be and as secure as it can be. Applying best industry standards best practices should be something that everyone no matter what field they are in, should apply those in every computer science field they practice.