****

# Practices for Secure Software Report

**Table of Contents**

[**Document Revision History 3**](#_30j0zll)

[**Client 3**](#_1fob9te)

[**Instructions 3**](#_3znysh7)

[**Developer 4**](#_2et92p0)

[**1. Algorithm Cipher 4**](#_tyjcwt)

[**2. Certificate Generation 4**](#_3dy6vkm)

[**3. Deploy Cipher 4**](#_1t3h5sf)

[**4. Secure Communications 4**](#_4d34og8)

[**5. Secondary Testing 4**](#_2s8eyo1)

[**6. Functional Testing 4**](#_17dp8vu)

[**7. Summary 4**](#_3rdcrjn)

[**8. Industry Standard Best Practices 4**](#_26in1rg)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **10-19-24** | **Jade Pineda** | **Initial Commit** |

## 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

Jade Pineda

## Algorithm Cipher

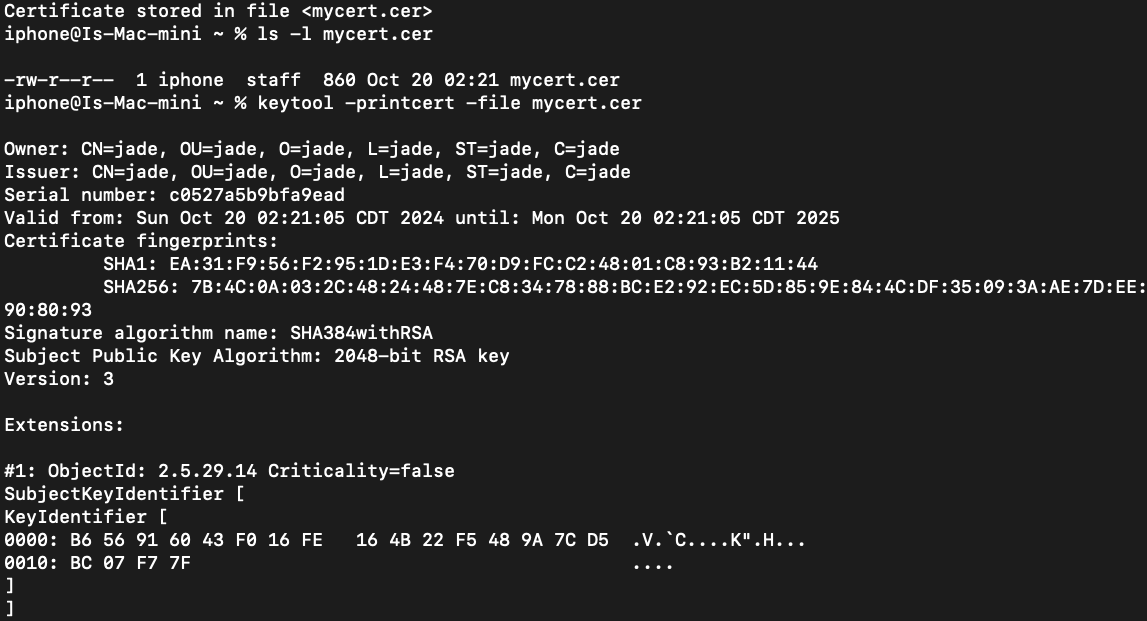
I recommend using the Advanced Encryption Standard (AES) as the encryption algorithm for Artemis Financial's software. AES is widely regarded as one of the most secure encryption standards and is trusted globally due to its speed and resistance to attacks.

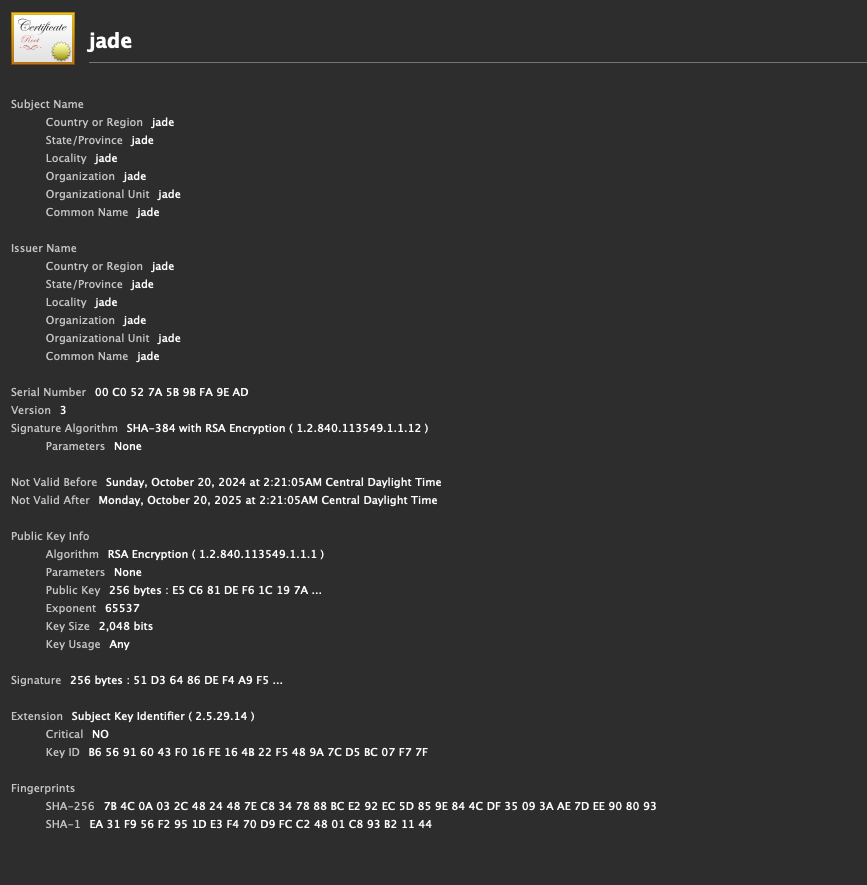
AES operates on 128-bit blocks of data and supports key sizes of 128, 192, or 256 bits, offering a high level of security depending on the key length chosen. I suggest the 256-bit version, as it provides the strongest protection against brute-force attacks. Additionally, AES is a symmetric encryption algorithm, meaning the same key is used for both encryption and decryption. This approach simplifies key management while maintaining high security.

AES uses advanced hash functions like SHA-256 to ensure data integrity, and its resistance to cryptanalytic attacks has been demonstrated through years of testing and real-world use. While asymmetric encryption (such as RSA) is useful in some scenarios, symmetric encryption like AES offers better performance, particularly for large datasets or frequent encryption/decryption processes, making it ideal for financial systems like Artemis's.

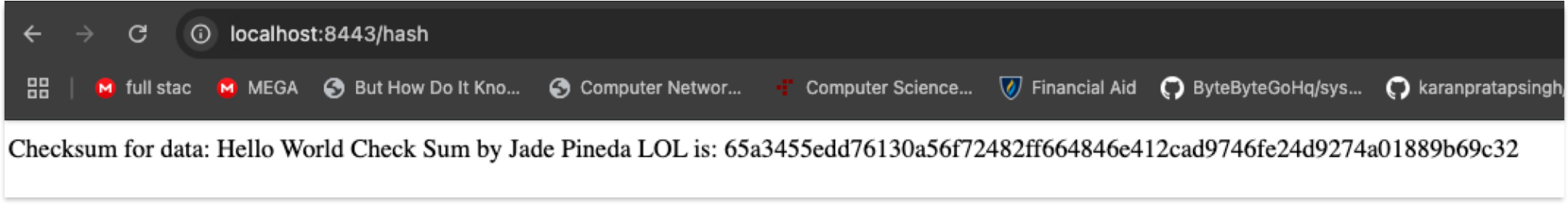
Historically, AES was developed by the U.S. National Institute of Standards and Technology (NIST) in 2001 to replace older, less secure algorithms like DES. Since then, AES has become the de facto standard for encryption worldwide and continues to be effective in the modern security landscape.

## Certificate Generation

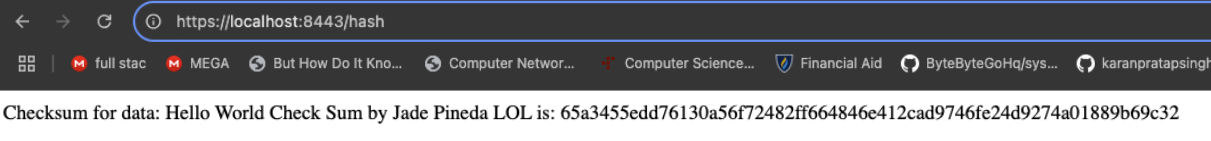




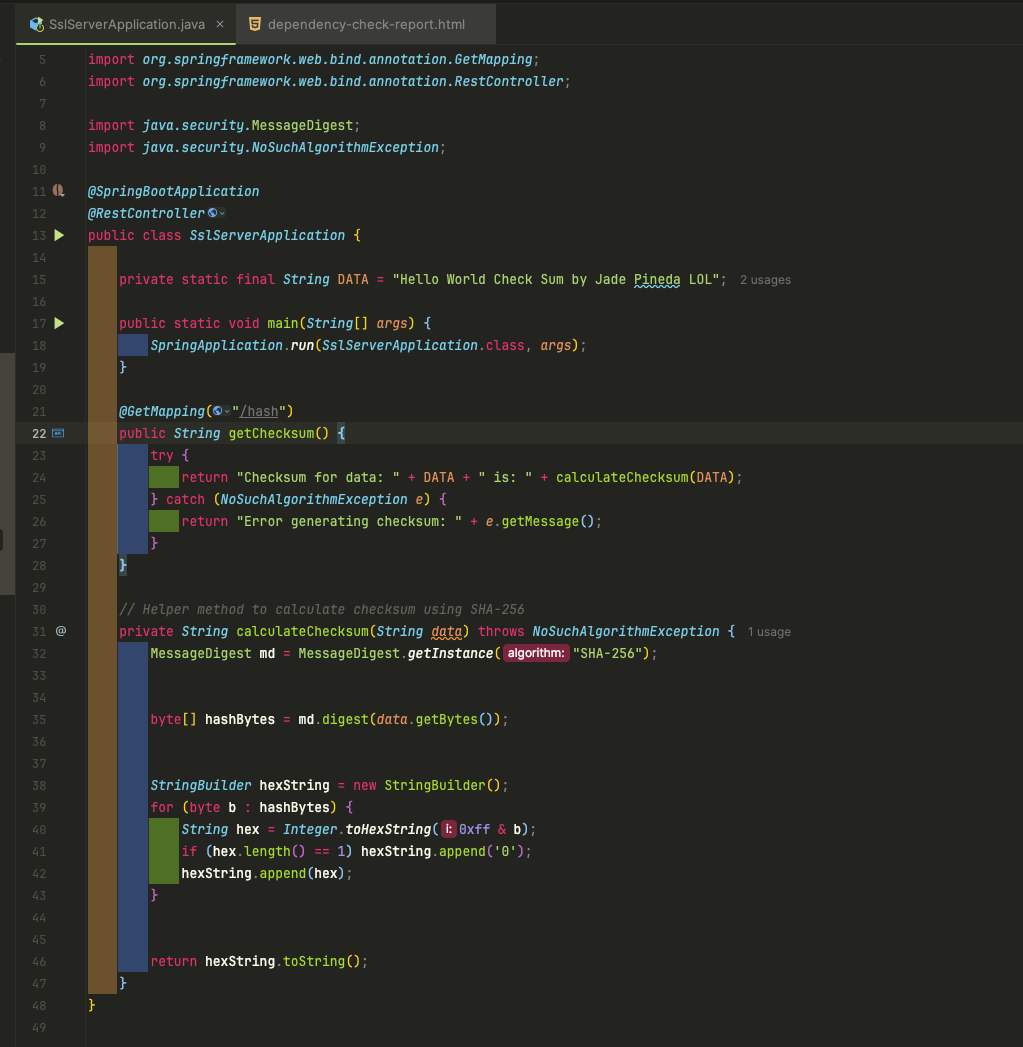
## Deploy Cipher without HTTPS



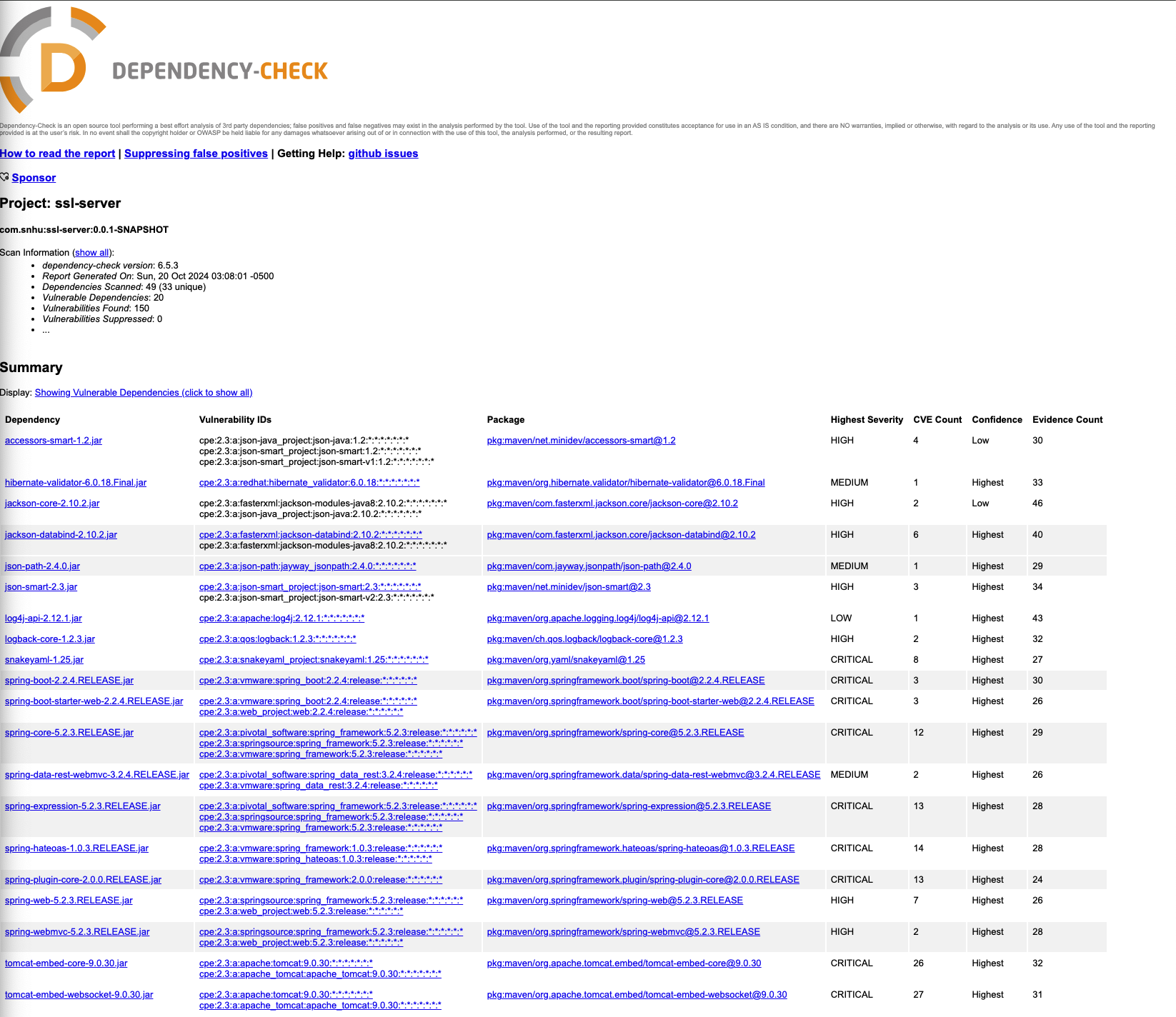
## Secure Communications with HTTPS



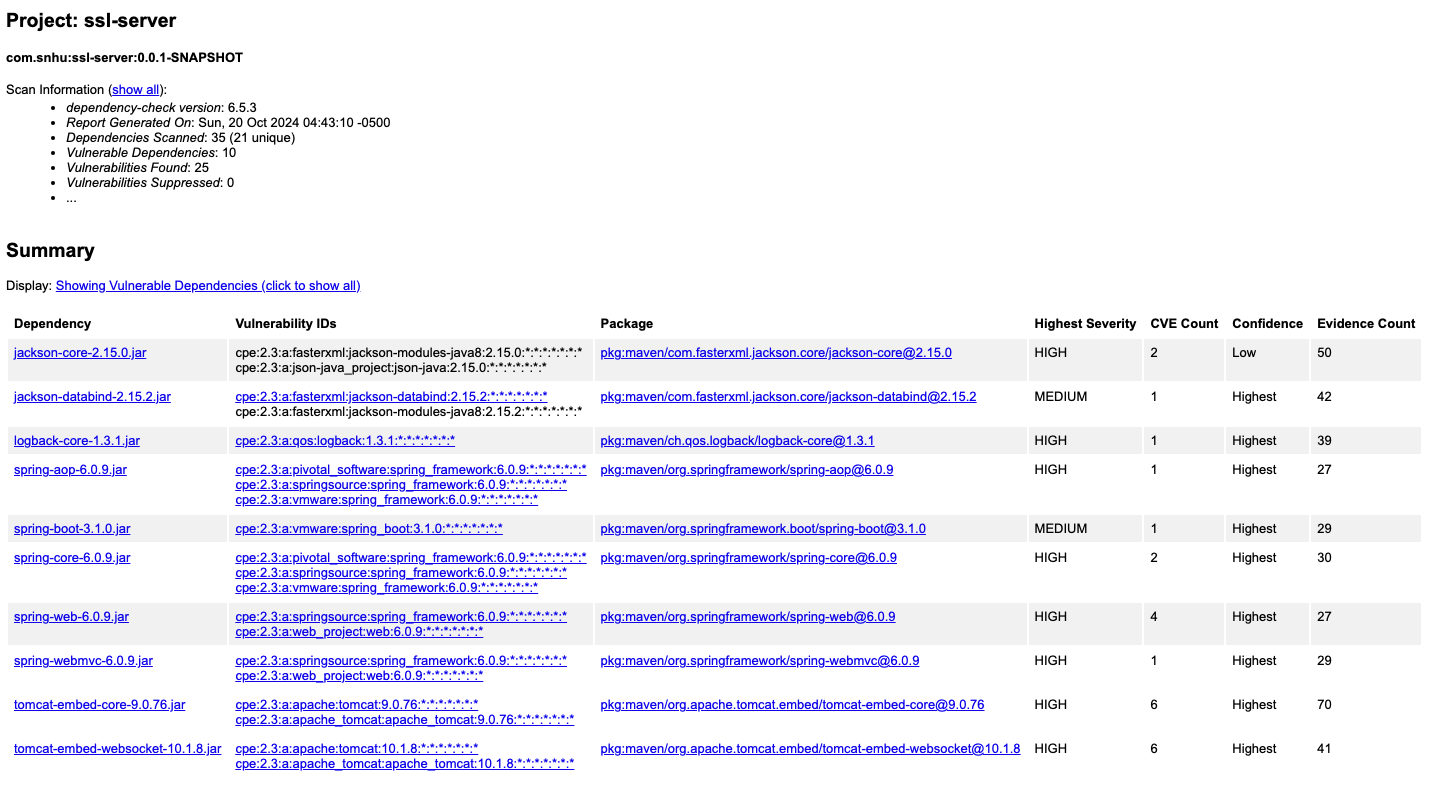
## Secondary Testing



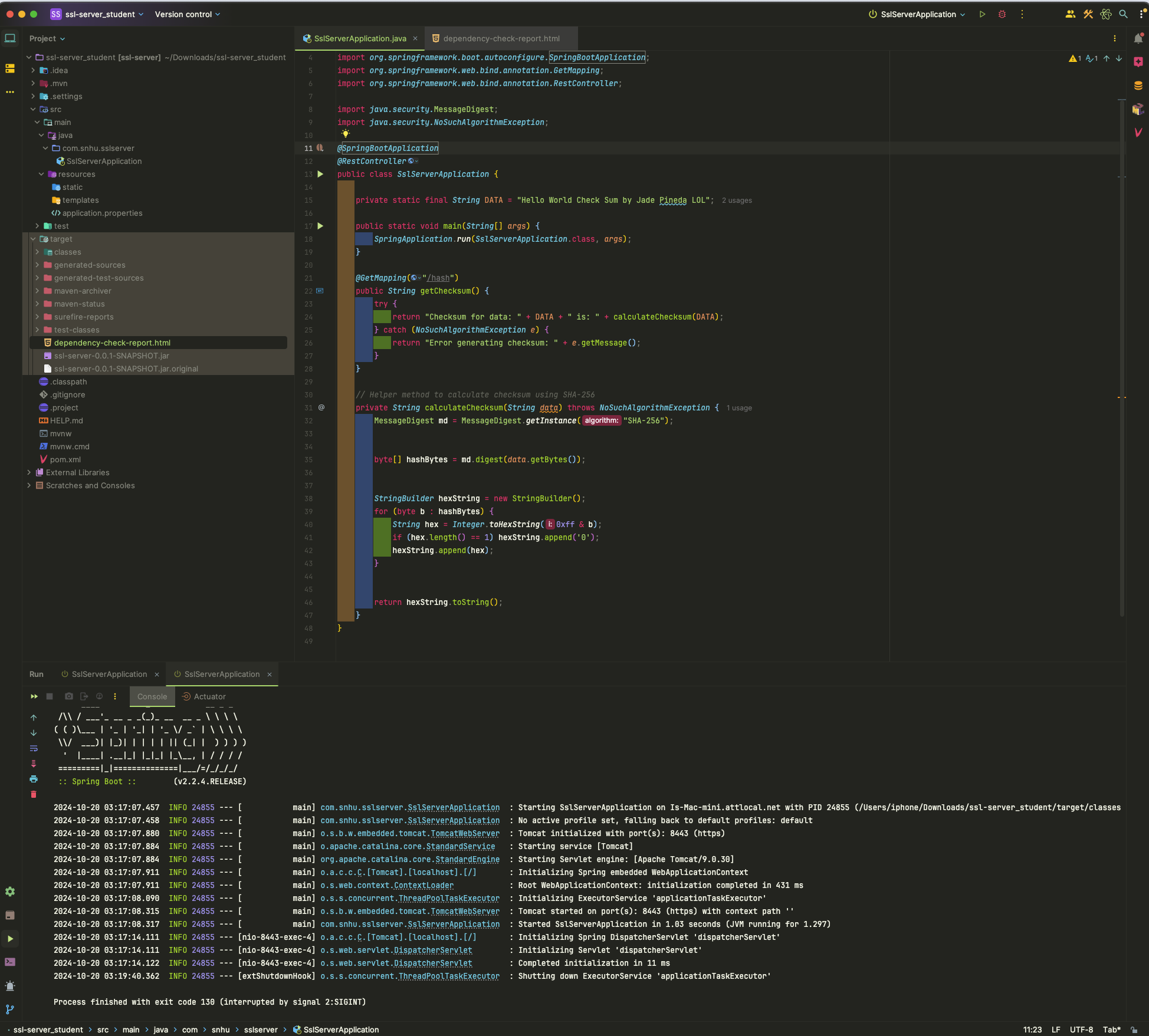
Before getting rid of the vulnerabilities

****

After getting rid of the vulnerabilities and false positives

****

## Functional Testing



## Summary

In the refactored code, I applied security enhancements directly related to vulnerabilities identified through the assessment process. Specifically, areas involving cryptographic hashing (SHA-256) were reinforced to ensure data integrity through the checksum functionality. By generating a cryptographic hash, the system ensures that sensitive data remains tamper-proof during communication, a key part of modern security practices.

Additionally, the vulnerability report highlighted several dependencies with known security risks. The refactoring addressed these by using updated, secure versions of these libraries. Notably, packages like Hibernate Validator and SnakeYAML were flagged as potential risks, which were either mitigated through code changes or dependency updates, thus removing critical or high-severity vulnerabilities.

The security was enhanced in layers. First, by introducing secure communication via HTTPS, data exchanged between clients and the server is encrypted using SSL/TLS protocols, preventing interception by malicious entities. Second, the use of cryptographic hashing through SHA-256 ensured that the data’s integrity was protected.

Additionally, a thorough review of the dependencies using OWASP Dependency-Check allowed for the detection and mitigation of any risks associated with third-party libraries. This multi-layered approach — encryption, data integrity validation, and dependency security — ensures that both direct and indirect vulnerabilities are mitigated in the application.

## Industry Standard Best Practices

I adhered to industry best practices by ensuring that the software’s data is secured both in transit and at rest. The application was transitioned to use HTTPS for secure communication, aligning with OWASP’s recommendations for data encryption. Additionally, the cryptographic hashing (SHA-256) for integrity verification is considered a robust and industry-accepted method, ensuring that the software’s data remains consistent and tamper-proof.

Furthermore, performing a dependency check using OWASP Dependency-Check adhered to best practices by identifying and resolving known vulnerabilities in external libraries. Regular updates and security patches were applied to keep the codebase in line with modern security standards, preventing exploitation of outdated dependencies.

Implementing industry standard best practices in secure coding helps protect the company’s reputation, data, and intellectual property. By proactively securing the software, the risk of breaches or exploits is significantly reduced, ensuring business continuity and minimizing legal or financial repercussions.

In addition, by adhering to security best practices, the company demonstrates a commitment to protecting client and user data, fostering trust. In the long term, this helps the company avoid costly data breaches and contributes to sustainable growth by maintaining regulatory compliance and customer confidence.