****

# Practices for Secure Software Report

**Table of Contents**

[**Document Revision History 3**](#_7vqzzdai17p9)

[**Client 3**](#_x7mopgh79fjb)

[**Instructions 3**](#_e2rvp2666y3v)

[**Developer 4**](#_ipmfvm42gpq5)

[**1. Algorithm Cipher 4**](#_c94iuh4ry3ye)

[**2. Certificate Generation 4**](#_zdg4nzofrybh)

[**3. Deploy Cipher 4**](#_tdznnr59oqkw)

[**4. Secure Communications 4**](#_rl8qot4ly5z1)

[**5. Secondary Testing 4**](#_tla0xw73iqhp)

[**6. Functional Testing 4**](#_71db6ocwkt26)

[**7. Summary 4**](#_ehvu5z8i11ft)

[**8. Industry Standard Best Practices 4**](#_w22zt0xfdziv)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **08/15/2025** | **Terrionna McGhee** |  |

## 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

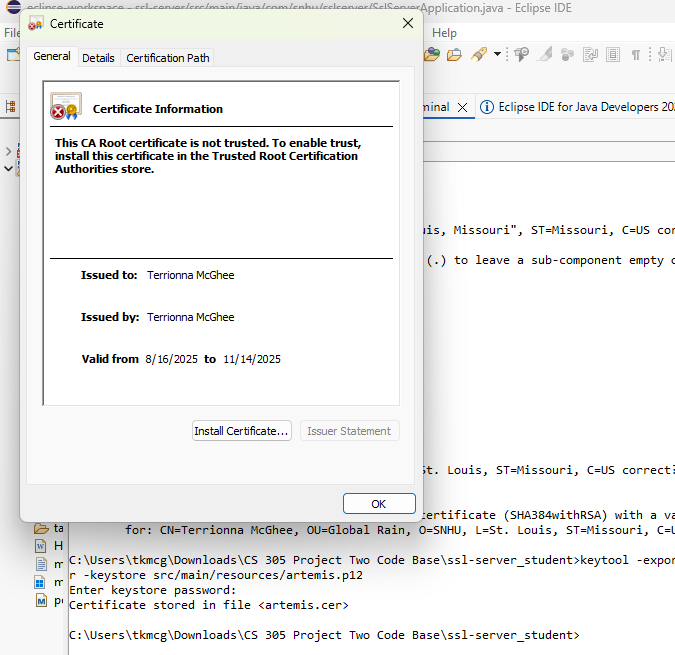
Terrionna McGhee

## Algorithm Cipher

I recommend SHA-256 for Artemis Financial's checksum verification. SHA-256 is a cryptographic hash function that produces a 256-bit hash value, making it highly secure against collision attacks. It uses symmetric operations with deterministic output - the same input always produces the same hash. SHA-256 provides strong data integrity verification with no known practical attacks, and it's widely supported in Java's MessageDigest library. The algorithm is computationally efficient while being infeasible to reverse, making it ideal for checksum verification in financial applications.

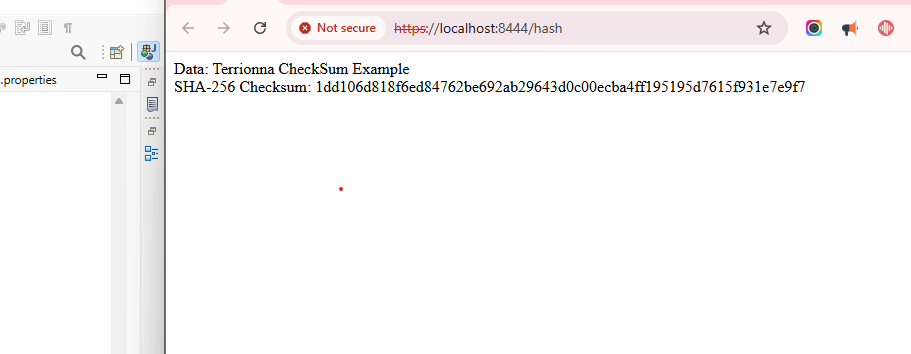
## Certificate Generation

Insert a screenshot below of the CER file.



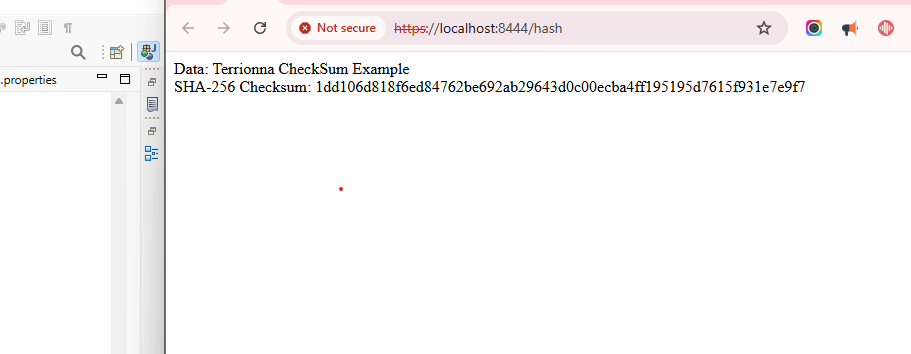
## Deploy Cipher

Insert a screenshot below of the checksum verification.



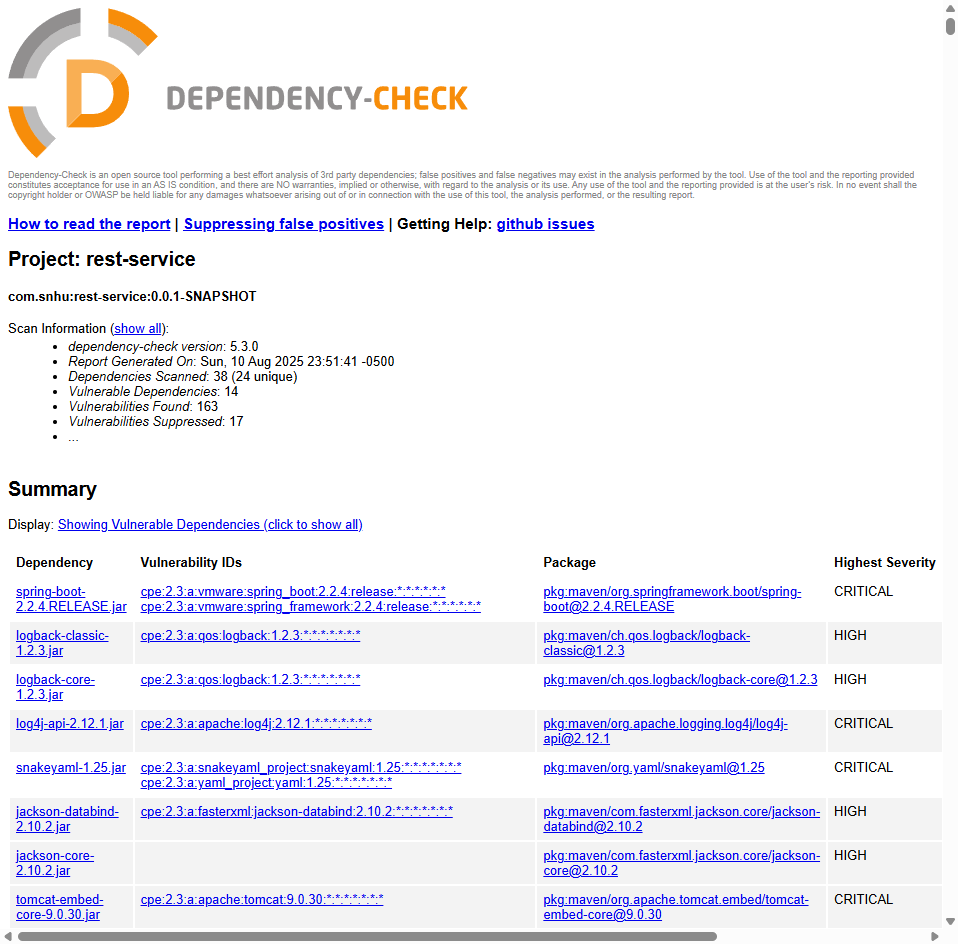
## Secure Communications

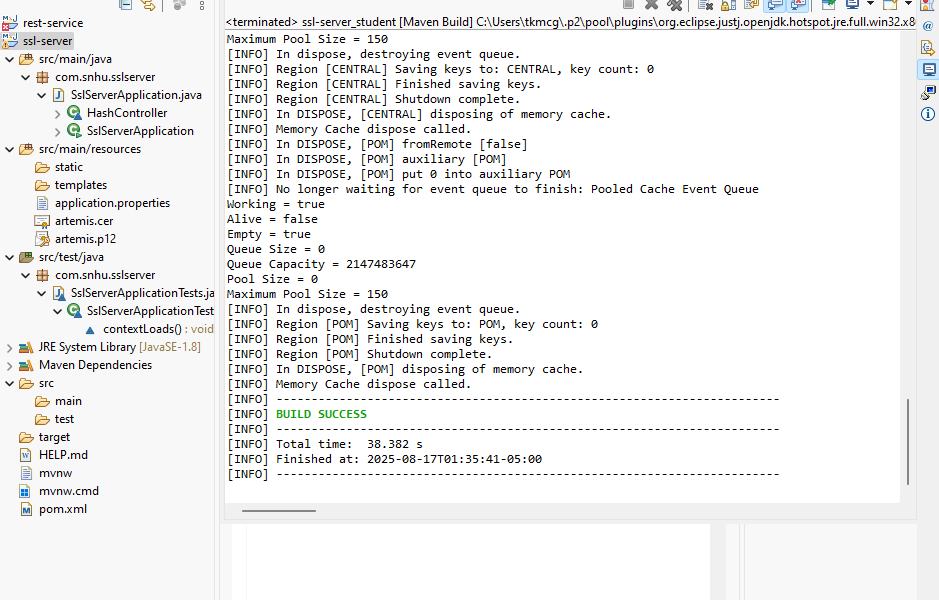
Insert a screenshot below of the web browser that shows a secure webpage.



## Secondary Testing

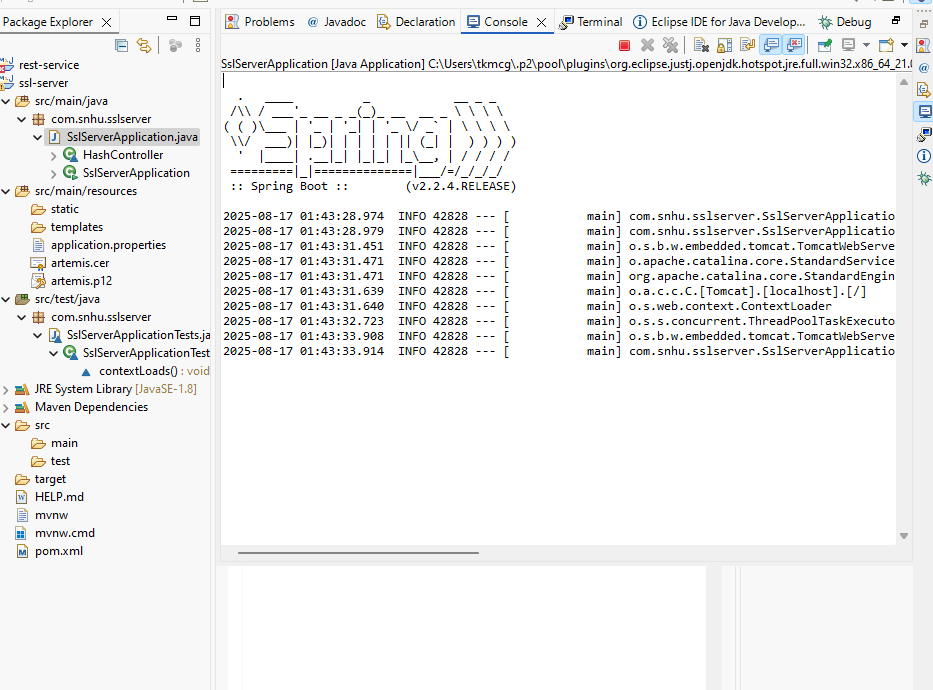
Insert screenshots below of the refactored code executed without errors and the dependency-check report.





## Functional Testing

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



## Summary

I refactored the SslServerApplication.java code to implement SHA-256 checksum verification and HTTPS secure communications. The key security areas addressed include: data integrity verification through cryptographic hashing, secure transport layer encryption using SSL certificates, and input validation through proper exception handling. I added layers of security by implementing the MessageDigest API for hash generation, configuring SSL/TLS certificates for encrypted communications, and enabling HTTPS protocol on port 8444. The refactored code now provides both data verification capabilities and secure transmission protocols required for financial data protection.

## Industry Standard Best Practices

I applied industry standard secure coding practices by using Java's built-in MessageDigest API rather than custom hash implementations, which maintains security through proven cryptographic libraries. The SSL certificate implementation follows standard PKI practices with RSA 2048-bit encryption. I maintained existing security by preserving the application's structure while adding security enhancements incrementally. These practices provide value to Artemis Financial by ensuring client data integrity, protecting against man-in-the-middle attacks, and meeting regulatory compliance requirements for financial data transmission. Proper implementation of these standards reduces security vulnerabilities and builds customer trust in the platform.