****

# Practices for Secure Software Report

Table of Contents

[Document Revision History 3](#_Toc102040754)

[Client 3](#_Toc102040755)

[Instructions 3](#_Toc102040756)

[Developer 4](#_Toc102040757)

[1. Algorithm Cipher 4](#_Toc102040758)

[2. Certificate Generation 4](#_Toc102040759)

[3. Deploy Cipher 4](#_Toc102040760)

[4. Secure Communications 4](#_Toc102040761)

[5. Secondary Testing 4](#_Toc102040762)

[6. Functional Testing 4](#_Toc102040763)

[7. Summary 4](#_Toc102040764)

[8. Industry Standard Best Practices 4](#_Toc102040765)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **10/18/2024** | **Nathanial Lowe** |  |

## 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

Nathanial Lowe

## Algorithm Cipher

The Advanced Encryption Standard (AES) is a widely used encryption algorithm, offering strong security through symmetric encryption. AES-256, which uses a 256-bit key, provides robust protection against brute-force attacks. It's efficient for encrypting and decrypting large volumes of data, making it ideal for securing sensitive information in financial systems like Artemis Financial’s. AES will ensure secure data transfers and, when combined with a checksum mechanism, can verify the integrity of the data during communication.

AES-256 uses a 256-bit encryption key for secure data protection, making it highly resistant to brute-force attacks. To verify data integrity, hash functions like SHA-256 are used alongside AES. SHA-256 generates a 256-bit hash (checksum) of the data, allowing recipients to verify that the data has not been altered during transmission. Together, AES-256 and SHA-256 ensure both the security and integrity of data in Artemis Financial’s application.

Random numbers ensure unpredictability in encryption, making it harder for attackers to guess keys. Symmetric keys (used in AES) use the same key for both encryption and decryption, making them fast and efficient for large data. Asymmetric keys (used in RSA) use a public key for encryption and a private key for decryption, allowing secure key exchange or identity verification. Often, both methods are combined to balance security and efficiency.

Encryption has evolved from simple ciphers like the Caesar Cipher to complex algorithms like AES and RSA. AES-256 is the current global standard for symmetric encryption due to its strong security, while RSA is widely used for asymmetric encryption. With the rise of quantum computing, research is focused on developing post-quantum encryption algorithms to protect against future threats.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a certificate

Description automatically generated

A screenshot of a computer

Description automatically generated

## 

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screenshot of a computer program

Description automatically generated

## Secure Communications

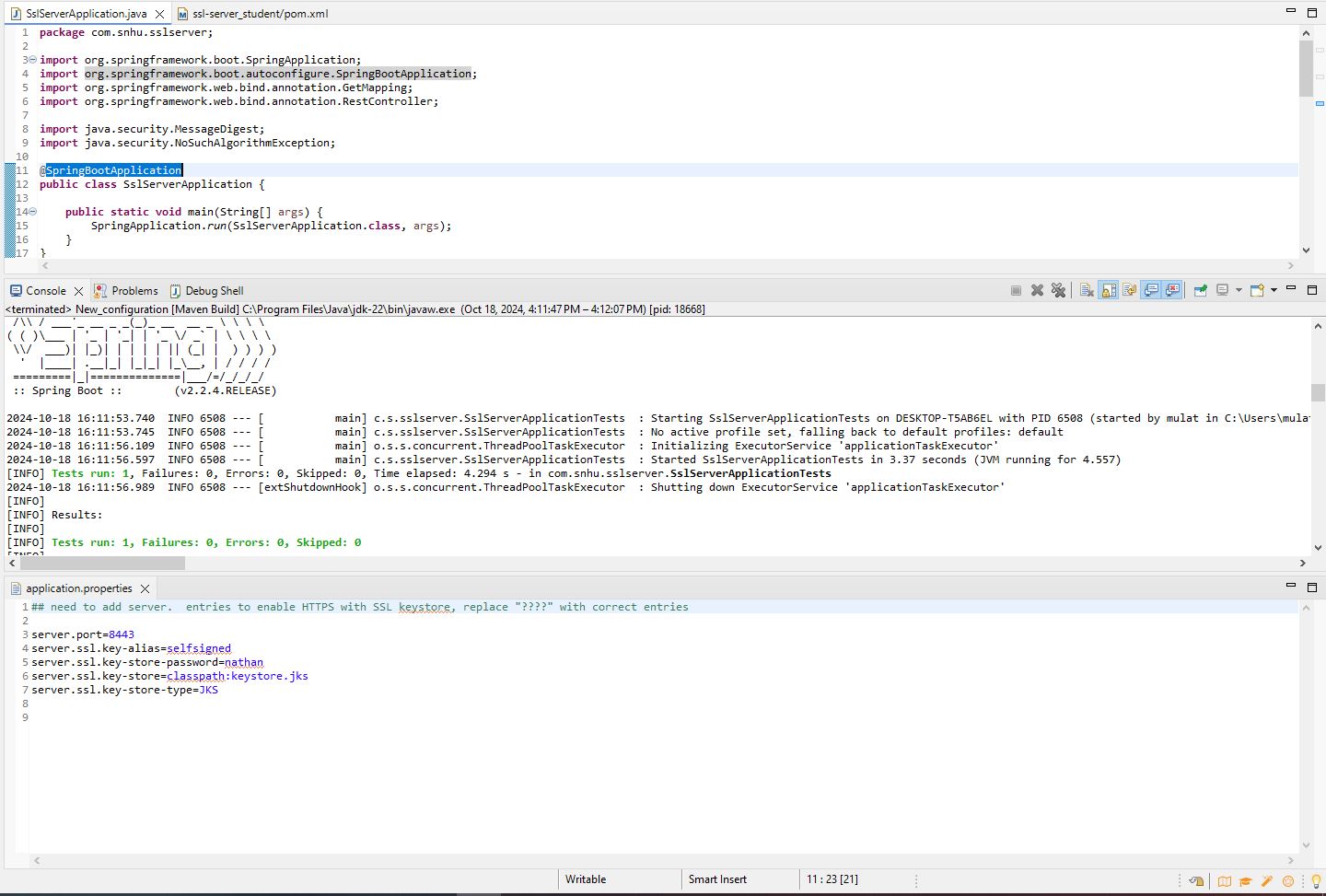
Insert a screenshot below of the web browser that shows a secure webpage.

A screenshot of a computer

Description automatically generated

## Secondary Testing

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



A screenshot of a computer

Description automatically generated

## Functional Testing

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

A screenshot of a computer code

Description automatically generated

## Summary

The code was refactored to enhance security for Artemis Financial by addressing key areas such as data encryption, integrity verification, and vulnerability scanning. HTTPS (SSL/TLS) was implemented to secure data in transit, and a SHA-256 checksum was added to verify data integrity. The OWASP Dependency-Check plugin was integrated to scan for vulnerabilities in third-party libraries. These layered security measures ensure the application is secure, protecting sensitive client data during transmission and preventing tampering.

## Industry Standard Best Practices

Industry-standard secure coding practices were applied to maintain and improve Artemis Financial’s software security. Key measures include enabling HTTPS/SSL for encrypted communication, using SHA-256 checksums for data integrity verification, and integrating OWASP Dependency-Check for vulnerability management. These practices protect sensitive client data, ensure regulatory compliance, reduce vulnerabilities, and minimize long-term costs by preventing potential security breaches. Overall, they strengthen the company’s security posture, build client trust, and ensure operational stability.