****

# 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** | **22Feb2025** | **Steven Copeland-Helzer** |  |

## 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

Steven Copeland-Helzer

## Algorithm Cipher

I recommend the SHA-256 encryption algorithm cipher with Artemis Financials objective in mind. The SHA-256’s collision resistant design ensures an extremely small likelihood of different inputs generating the same hash value. When the algorithm takes an input message and processes it into multiple rounds, bitwise logical operations are applied before arithmetic and conditional statements are further added to the input. Shift amounts and constants are also added in each round that, when combined with all other operations, creates a unique hash value which will always have a 256-bit length (Madan, 2023). The features of the SHA-256 hash function properly ensure that even a small difference in input values will result in a very different hash value. Avoiding collisions is important for maintaining data integrity. The absence of collisions will ensure the checksum properly illustrates the input when determining if the data received is a match with the original.

Symmetric encryption algorithms are considered one of the most basic techniques and use the same key for both encryption and decryption (Awati et al. 2024). While simple and less time consuming, symmetric algorithms are at a disadvantage when it comes to security compared to asymmetric algorithms requiring two separate keys. Advanced Encryption Standard (AES) makes up for this with its three block ciphers, a 128-bit key length which undergoes 10 rounds of hash function encryption, a 192-bit key length which has 12 rounds, and a 256-bit key length that has 14 rounds. Each round of encryption utilizes a hash function that mixes the plaintext input, transforming it into ciphertext output.

The beginning of computer encryption was in 1975 with the development of the Data Encryption Standard (DES) by researchers at IBM (Schneider, 2024). Due to technological advancements and DES’s short key length, AES replaced DES as the standard for data encryption in 2001 and is still the standard in present day.

## Certificate Generation

A computer screen shot of a black screen

AI-generated content may be incorrect.

## Deploy Cipher

A screenshot of a computer

AI-generated content may be incorrect.

## Secure Communications

A screenshot of a computer

AI-generated content may be incorrect.

## Secondary Testing

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

## Functional Testing

A screenshot of a computer

AI-generated content may be incorrect.

## Summary

Refactoring the SslServerAplication.java file, I added a secure RestController the hash RESTful endpoint. The ServerController class addresses the secure coding concerns highlighted in the Vulnerability Assessment Diagram. Refactoring the pom.xml file, I updated the OWASP Maven dependency check to its most up-to-date version 12.1.0. Incorporating the SHA-256 algorithm to enable checksum verification, I ensured the data transferred remains unaltered.

## Industry Standard Best Practices

Incorporating best practices to enhance security, addressing the known vulnerabilities, I adopted the use of HTTPS to secure communications by encrypting data passages. By implementing the SHA-256, its high collision resistance maintains data integrity further enhancing the application security measures. Following industry standard best practices is key for ensuring all sensitive data for Artemis Financial and their customers is properly secured. Implementing the necessary security measures and ensuring everything is up to date will create the best application for any client to address all potential security concerns.

**References:**

Madan. (2023). *A Deep dive into SHA-256: Working Principles and Applications.* Medium. <https://medium.com/@madan_nv/a-deep-dive-into-sha-256-working-principles-and-applications-a38cccc390d4>

Awati, R., Bernstein, C., Cobb, M. (2024). *Advanced Encryption Standard (AES).* TechTarget. <https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard>

Schneider, J. (2024). *A brief history of cryptography: Sending secret messages throughout time*. IBM. <https://www.ibm.com/think/topics/cryptography-history>