****

# 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** | **10/11/23** | **Tyreen Borden** |  |

## 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

Tyreen Borden

## Algorithm Cipher

I recommend using AES 256 with an accompanying SHA 256 hash algorithm. AES 256 is the Advanced Encryption Algorithm, and it has a size of 256 bits. AES is a NIST standard encryption algorithm that is even used by the U.S. Government. The algorithm was created as a replacement to the Data Encryption Standard that suffered vulnerabilities due to its short bi size. AES is a symmetric key algorithm. This means it uses the same key for both encryption and decryption as opposed to a non-symmetric key algorithm. Since it is a symmetric key algorithm, it is important to have a secure key because if the key is compromised so, is the encryption. This can be done using a random number generator to create the said key. The combination of AES 256 and SHA 256 should ensure the security and integrity of the file verification needed by Artemis Financial.

## Certificate Generation

Insert a screenshot below of the CER file.

A black screen with white text

Description automatically generated



A computer screen with white text

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screenshot of a computer

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 screen shot of a computer

Description automatically generated

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 program

Description automatically generated

## Summary

The code has been refactored to address the security needs of Artemis Financial. The client/server communication has been secured by adding HTTPS protocol using a self-signed certificate generated using the java keytool. The cryptography implemented was SHA-256 hash algorithm to secure the data. A try-catch code block is used to handle the code errors in the checksum verification.

## Industry Standard Best Practices

We ensured secure communications by implementing HTTPS protocols. Using the SHA-256 cryptographic hash algorithm we protected the integrity of the data being transmitted. Used proper error-handling in the checksum verification to ensure no sensitive error information was leaked. The dependencies were checked for vulnerabilities as well and managed accordingly. These practices are valuable to the company because they allow customers to have trust in the company and therefore be more likely to use their services. It protects the company from any liability that may arise by not meeting regulatory security requirements. It also saves costs associated with the fallout of security breaches.