****

# 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** | **[Date]** | **Brandon Mullins** |  |

## 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

Brandon Mullins

## Algorithm Cipher

The goal for Artemis Financial is to ensure client data stays always protected. For this reason, I chose AESas the main encryption method, paired with SHA-256 for hashing.

AES is a symmetric encryption algorithm trusted around the world for securing sensitive data. It's fast, efficient, and extremely difficult to break with brute force. With key lengths of 128, 192, and 256 bits, it offers flexible levels of security depending on the system needs. In our case, 256-bit encryption provides a strong balance between performance and protection.

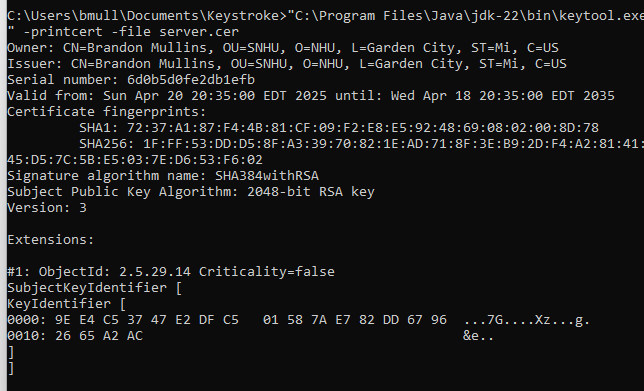
SHA-256, a hash function in the SHA-2 family, was used to validate file integrity. This function converts input data into a fixed-length value, helping us detect even the smallest change to the data. Since hash values are generated based on randomized bits and strong math, it’s almost impossible to reverse-engineer.

AES uses symmetric keys, meaning the same key is used to both encrypt and decrypt data. This is faster than asymmetric encryption, which uses a pair of public/private keys. While symmetric encryption is great for internal systems, it still requires secure key storage and exchange.

Historically, AES replaced DES (Data Encryption Standard) in the early 2000s and is still widely used in government, finance, and tech companies today. It’s one of the most tested and proven algorithms we have, which makes it the right choice for Artemis Financial’s software system.

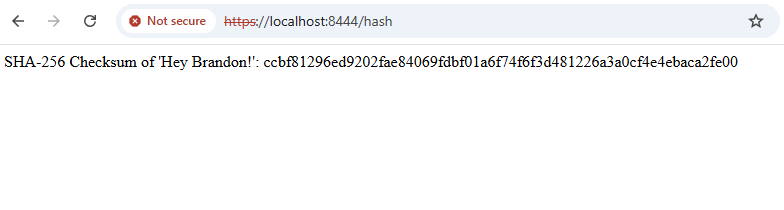
## Certificate Generation

Insert a screenshot below of the CER file.



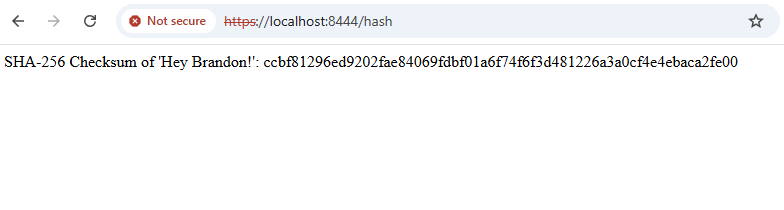
## Deploy Cipher

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.

A screenshot of a computer

AI-generated content may be incorrect.

## Summary

Going through this project, I applied secure software development practices to enhance the safety and reliability of a Spring Boot web application. My primary focus was on implementing data encryption, establishing secure communication through HTTPS, and validating application integrity. I started by generating a self-signed cert using Key tool and configuring the application to serve content over HTTPS. This allowed the application to securely encrypt communication between the server and client.

## Industry Standard Best Practices

Patch regularly,

Use HTTPS everywhere

Avoid Hardcoding

Run scans often