****

# 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** | **24 April 2024** | **Brianna De La Riva** |  |

## 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

Brianna De La Riva

## Algorithm Cipher

Artemis Financial is a banking institution that provides a vast array of financial services to clients ranging from banking, savings, investments, and retirement on a global scale. This means that security is a top priority to protect clients sensitive data and practice secure encryptions. According to Linkedin (n.d), "to ensure a strong encryption algorithm and key, you should use a well-known and widely used encryption algorithm, such as AES, RSA, or SHA. The key length should match the encryption algorithm's strength, such as 256 bits for AES or 2048 bits for RSA" (para. 1). Due to the enhanced encryption algorithms and security, I would recommend the SHA-256 encryption cipher algorithm. The SHA-256 is a commonly utilized cipher algorithm for secure transactions and data encryption. The hash functionality and bit levels allow for randomness which can help prevent hackers from accessing key information. The 256-bit encryption provides additional security due to its larger size, additionally the use of random numbers, characters, and symbols can be useful in further protecting information from hackers since it will be harder to decrypt with the randomness.

Furthermore, symmetric as well as non-symmetric keys are typically used to increase protection. The differences between symmetric vs non-symmetric is that symmetric "uses a single key that needs to be shared among the people who need to receive the message while asymmetric encryption uses a pair of public keys and a private key to encrypt and decrypt messages when communicating" (Gupta, 2023, para.4). Due to asymmetric keys requiring two keys, are they often recognized as being more secure than the symmetric keys. Encryption algorithms have a long history, ranging as far as 1900 BC. According to Sidhpurwala (2023) "The first known evidence of the use of cryptography was found in an inscription carved around 1900 BC, in the main chamber of the tumbling of the nobleman Khnumhotep II, in Egypt" (para. 1). Data Encryption is a very well used tool to help protect data throughout the ages.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a computer

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.

Screenshot below showcases the security self-sign certificate active. I kept receiving an error in various browsers for it being unsecure because it was a self-signing certificate, however it is active during the page load. I also included a screenshot of the terminal showcasing detailing more information. I believe I am sending encrypted data and it is working as self sign certificate, however since there is no third-party certificate authority claiming it is registered and safe, it is listing the unsecure warning. I believe the screenshots below can help address this.

A blue and black rectangular object

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot 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

A computer screen shot of a program

Description automatically generated

## Functional Testing

A screenshot of a computer program

Description automatically generated

## Summary

The Artemis Financial application was programmed to have self-signed certificates to convert the HTTP to HTTPS for enhanced security. The pom.xml file was revised to ensure that the most up-to-date maven dependency check version was being utilized. By bringing the webpage to a more secure version, certificate generations were created using the SHA-256 cipher algorithm. This helped enhance the security of the website and ensure users are handling a secure webpage. To ensure that the hash function was functionally correctly, I created a check sum verification. During the refractor of the code there was a RestController created for the hash function and a ServerController used. The cipher algorithm of SHA-256 was chosen for its reputation for being secure and having a higher bit-level. As a recommendation, the pom.xml file should be frequently checked to ensure they are using up-to-date versions of tools, as well as within the SslServiceApplication.java file.

Furthermore, I went through various testing using Postman, the MAC terminal, and Eclipse runs to ensure things were at optimal functionality. Due to the sensitive client information Artemis Financial is working with, executing data security is a top priority. We will be able to ensure that only authorized users are able to access their respectful authorized information. This helps ensure less there are less data leaks, attacks, and more from hackers against Artemis Financials’ clients.

## Industry Standard Best Practices

Some industry best standards that were utilized included having code comments, debugging with break points, running security vulnerability checks, and performing testings.