****

# 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** | **12/11/2022** | **Brandon Lingenfelter** |  |

## 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 Lingenfelter

## Algorithm Cipher

With consideration to the Java Security Standards document the encryption algorithm cipher that I would recommend for Artemis Financial is the Advanced Encryption Standard (AES). There are risk that will have to be considered when implementing AES. The first risk to be considered is that the key size used is large enough not to be cracked. The other risk is that AES has symmetric encryption which gives all the parties involved the decryption key. This means if someone with the decryption key has weak security standards the key could easily be compromised.

The United States government uses AES to secure documents and have stringent regulations for securities and banks. The cipher will be used to encrypt the data of Artemis Financials customers. I believe that the AES is the best ciphers since it has yet to be broken and can be implemented using 128-bit, 192-bit, or 256-bit keys to personalize the level of protection. It is possible using the most secure cipher could cause slower data retrieval.

The hash function can be used to map data of arbitrary size to fixed size values. The bit lengths increase security by requiring a longer key. Random numbers can be generated to help make a more robust key. Symmetric encryption uses a private key that is used to encrypt or decrypt information shared between users. Asymmetric encryption does not require the users to share the key with the other parties. The history of cryptography goes as far back as the Ancient Egyptians. Today Encryption algorithms are going to have to become quantum computing resistant in the near future or they will become obsolete.

## Certificate Generation

Insert a screenshot below of the CER file.

Text

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

Graphical user interface, text

Description automatically generated with medium confidence

## Secure Communications

Graphical user interface, text, application, email

Description automatically generated

## Secondary Testing

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated with low confidence

## Functional Testing

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

Text

Description automatically generated

## Summary

The areas of the vulnerability assessment process flow diagram that I addressed when refactoring the code are APIs, Cryptography, Client/Server, Code Error, and Code Quality. To fix the API I addressed with implementation of HTTPS. To address cryptography encryption algorithm ciphers and hash functions, as well as checksum verification. I ensured the information was sent from the client to the server to be displayed. Code error was addressed by utilizing exceptions within our class like NoSuchAlgorithm exception within the myHash method. Code quality was reviewed to ensure functionality and readability.

I added security to the application using a self-signed certificate to allow for HTTPS utilization. Ensuring that the HTTPS is implemented makes our browser secure. Pom.xml was also refactored to ensure that the vulnerabilities are discovered and reported in the dependency check. I checked the hashing function works by verifying with checksums. Following the dependency check we make sure that everything in our application is up to date.

## Industry Standard Best Practices

It is always imperative to be sure you are utilizing industry best practices when building applications. This practice makes for readable applications that are up to date, secure and maintainable. Using these best practices can be beneficial for companies by making the code more secure.