****

# 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]** | **Carl LaLonde** |  |

## Client



## Developer

Carl LaLonde

## Algorithm Cipher

I recommend implementing AES-256 encryption algorithm for this current project. Advanced Encryption Standard – 256 “is a virtually impenetrable symmetric encryption algorithm that uses a 256-bit key to convert your text or data into a cipher” (Kananda, 2024). AES utilizes one key for both encryption and decryption methods. As a symmetric encryption algorithm, this simplifies key management for long-term storage compared to asymmetric encryption, which utilizes two keys, both a public and a private. AES uses differing key lengths of 128, 196, and 256 bits, which offers very strong security against unauthorized access.

Hash functions may be used in conjunction with AES encryption, further protecting the data. Hash functions protect data by creating a fixed-size output from input data. "A cryptographic hash function (CHF) is an equation that is widely used to verify the validity of data” (GeeksForGeeks, 2024). Pairing AES-256 with a hash function such as SHA-256 would be ideal. SHA-256 will create a unique hash of the data and any change in the data will result in a completely different hash, letting us detect if there has been any tampering with the data.

## Certificate Generation

A black screen with white text

Description automatically generated

A computer screen with text on it

Description automatically generated

## Deploy Cipher

A screenshot of a computer

Description automatically generated

## Secure Communications

A screenshot of a computer

Description automatically generated

## Secondary Testing

A screenshot of a computer

Description automatically generated

## Functional Testing

A screenshot of a computer program

Description automatically generated

## Summary

I identified security vulnerabilities in the existing code by performing code analysis. To help strengthen security, I implemented Advanced Encryption Standard (AES-256) for encrypting the data and securing communications for Artemis Financial. Along with AES-256, I integrated a hash function, SHA-256, to help verify the authenticity of the data being processed.

## Industry Standard Best Practices

By applying input validation, the application safeguards against attacks/threats such as SQL injection and command line injection. This process should involve verifying user input and cleansing it so it cannot trigger any malicious activity. Additionally, secure authentication has been enhanced by introducing hash algorithms. Encryption of sensitive data as well as the use of HTTPS for secure data transmission has been applied ensuring that users and applications will operate within the level of access necessary for their role. These best practices are important for securing the application as well as improving the company’s overall well – being and image to the public. These security measures will confirm compliance with regulations and prevent costly data breaches, further building trust among Artemis Financials clients.

**References**

GeeksforGeeks. (2024, May 14). *Cryptography Hash Functions*. GeeksforGeeks. <https://www.geeksforgeeks.org/cryptography-hash-functions/?utm_source=chatgpt.com>

Kananda, V. (2024, November 13). *What is AES 256 encryption & how does it work?*. Progress Blogs. https://www.progress.com/blogs/use-aes-256-encryption-secure-data