****

# 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** | **04/27/2025** | **Kaley Lavery** | **Final Project** |

## Client



## 

## Developer

Kaley Lavery

## Algorithm Cipher

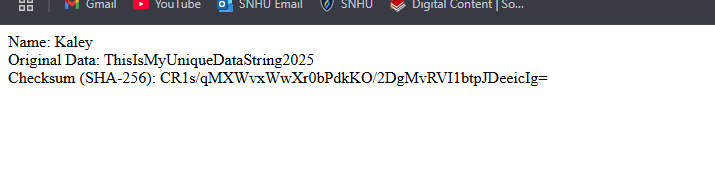
For Artemis Financials’ secure communication needs, I recommend the SHA-256 hash algorithm. SHA-256 produces a 256-bit output, offering strong collision resistance and excellent data integrity verification. It is a one-way cryptographic function, meaning it transforms data into a hash without using symmetric or asymmetric keys. SHA-256 is widely trusted across industries, endorsed by NIST, and used in highly secure environments such as blockchain technology and federal systems. Given its proven reliability and current acceptance, SHA-256 best fits Artemis Financials’ requirement for secure and verifiable file transfers.

## Certificate Generation

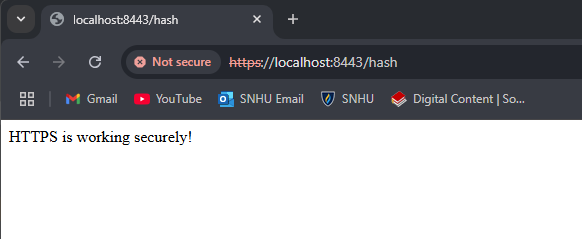
A screenshot of a computer program

AI-generated content may be incorrect.

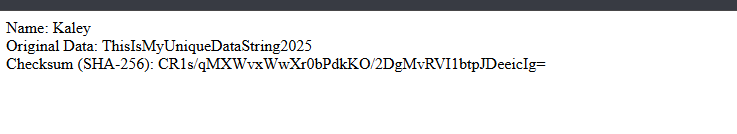
## Deploy Cipher

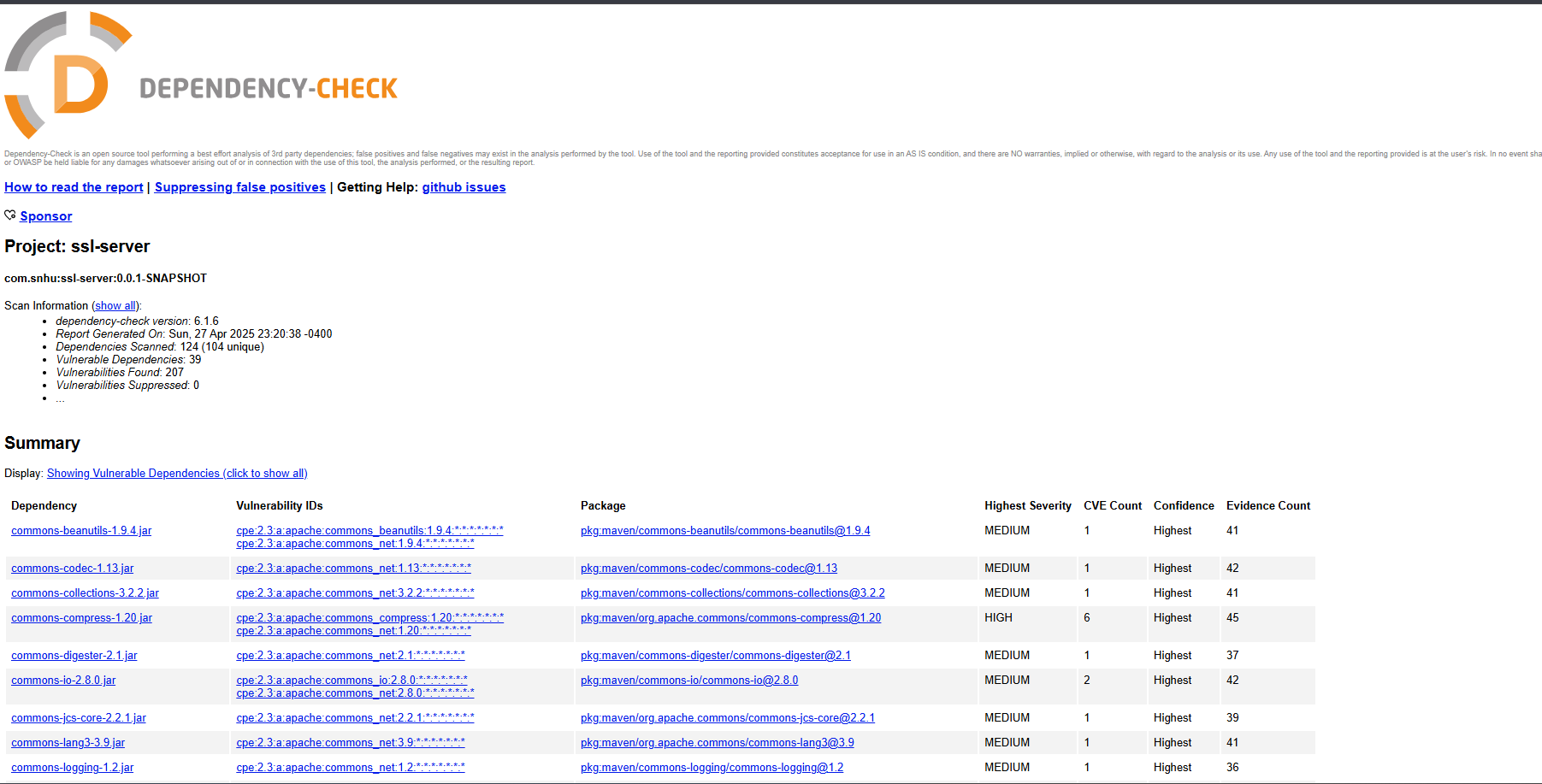


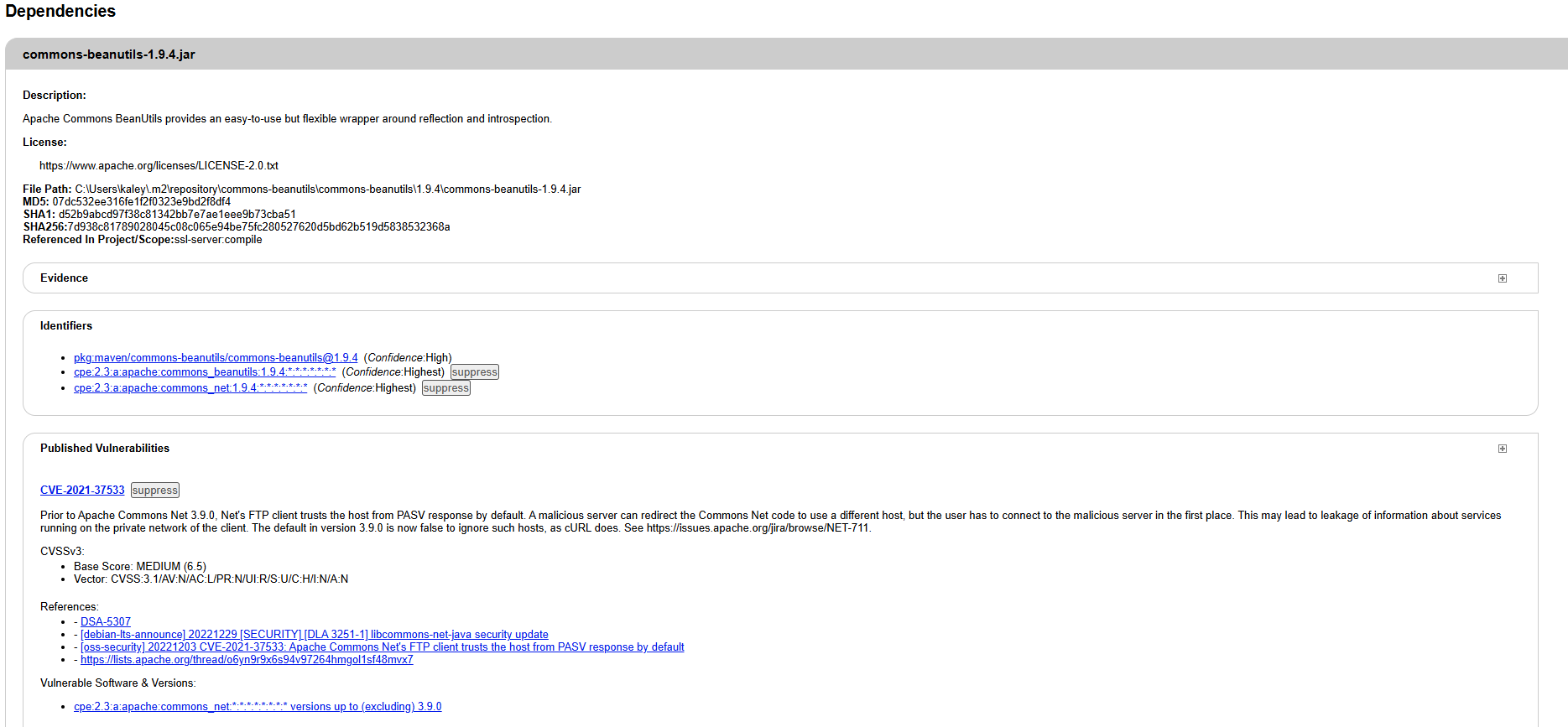
## Secure Communications



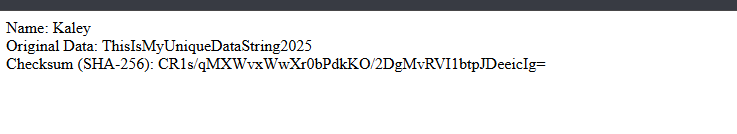
## Secondary Testing







## Functional Testing



## Summary

The refactoring of Artemis Financial’s software focused on strengthening security by implementing the SHA-256 hash algorithm, checksum verification, and secure communication protocols. The SHA-256 algorithm was used to securely hash sensitive financial data, ensuring its integrity and confidentiality during transmission. A checksum mechanism based on SHA-256 was added to verify that data remained unaltered during transfer, protecting against tampering.

The code was also refactored to transition from HTTP to HTTPS, encrypting all communication between the client and server using SSL/TLS to prevent unauthorized access and man-in-the-middle attacks.

These changes address key vulnerabilities identified in the assessment, including insecure data transmission, data tampering risks, and lack of strong encryption. By using SHA-256 for encryption and verification, and enforcing HTTPS for secure communication, the refactored software ensures the confidentiality, integrity, and authenticity of sensitive financial data.

This comprehensive refactoring adheres to modern security standards and best practices, offering a robust defense against data breaches and unauthorized access.

## Industry Standard Best Practices

The application now follows industry-standard best practices for secure coding, such as the use of SSL/TLS encryption for secure communication, implementation of cryptographic hash functions for data integrity, and proper certificate management. The refactoring process has strengthened the overall security posture of the software, mitigating risks associated with data breaches, unauthorized access, and man-in-the-middle attacks.This layered approach to security ensures that Artemis Financial’s software is robust, secure, and capable of protecting the sensitive financial data of its clients.