****

# 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** | **8/15/2024** | **Aspen Koreny-Crawford** |  |

## 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

Aspen Koreny-Crawford

## Algorithm Cipher

**Chosen Algorithm: Advanced Encryption Standard (AES)**

* **Overview:** AES is a symmetric encryption algorithm widely recognized for its security and efficiency. It supports key sizes of 128, 192, and 256 bits, making it a versatile choice for securing data transmissions.
* **Hash Functions and Bit Levels:** AES uses symmetric keys, meaning the same key is used for both encryption and decryption. The SHA-256 hash function can be employed for generating checksums, providing a 256-bit hash value.
* **Symmetric vs. Asymmetric:** Symmetric encryption (e.g., AES) is suitable for encrypting large data volumes quickly, while asymmetric encryption (e.g., RSA) is used for secure key exchange. In this scenario, symmetric keys are ideal for encrypting the data, ensuring quick and secure communications.
* **History and Current State:** AES has been the encryption standard since it was established by NIST in 2001. It remains unbroken and is trusted for securing sensitive data in government and financial institutions.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a certificate

Description automatically generated

A computer screen shot of a black screen

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.

A screenshot 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

## Functional Testing

Insert a screenshot below of the refactored code executed without errors.A screen shot of a computer program

Description automatically generated

## Summary

The code was refactored to address vulnerabilities identified during the testing process. The refactoring included updating the code to implement secure HTTPS communication and securely handling cryptographic operations using industry-standard algorithms like AES. Security testing protocols were also followed, including dependency-check static testing, which confirmed that no new vulnerabilities were introduced. These efforts enhanced the application's overall security posture by ensuring that data at rest and data in transit are secure.

## Industry Standard Best Practices

Throughout the development process, industry-standard best practices were consistently applied to ensure secure coding. This included implementing HTTPS for secure data transmission, using strong encryption like AES, and employing secure coding techniques such as input validation and exception handling to mitigate vulnerabilities. Regular security testing, including dependency-check static analysis, was conducted to identify and address potential security flaws, contributing to the company’s overall security strategy.