****

# 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** | **October 20, 2024** | **Tashi Anderson** | **Final update** |

## 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

Tashi Anderson

## Algorithm Cipher

After reviewing the scenario and the Java Security Standard Algorithm Names, an appropriate encryption algorithm cypher is the AES, or advanced encryption standard, which was first used by the US government in 2001. AES uses key lengths from 128 to 256 bits, which allows for stronger security based on the amount of time it would take to crack the password/key.

Some best practices to use alongside the AES would be following the nationally/internationally accepted standards and codes of practice. Another practice to partake in is backing up data, which could be lost in a ransomware attack or a server outage.

AES is a symmetric encryption algorithm, which uses the same key for the encryption and decryption of both the plaintext and cyphertext. AES also uses the SPN structure, which takes in plaintext and keys to apply alternating layers of s-boxes and p-boxes to produce the final cyphertext. By AES using these structures proves how secure the standard truly is, which is why it is the most widely used algorithm for encryption.

Encryption structures like AES have been in use for centuries, for example the Caesar cipher is a type of substitution cypher that was used by Julius Caesar to send secret messages to his generals in the field. There have been improvements on these encryption algorithms which are much more complex than simply substituting letters in for one another, such as AES and RSA.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a certificate

Description automatically generatedA computer screen with text on it

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screen shot of a computer

Description automatically generated

## Secure Communications

Insert a screenshot below of the web browser that shows a secure webpage.

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 screen shot of a computer program

Description automatically generatedA 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

Within this code, I updated and used @RestController, which introduced the SHA-256 hash because of it being one of the most secure hash protocols and the low risk of collisions. With this, I could return the data string of “Hello Tashi Anderson!” and the checksum value string which was the SHA-256 value that was used. By using a self-signed certificate, I was able to implement the usage of HTTPS as a secure way of hosting potential sensitive data. A big part of ensuring that this program was secure was the testing, which uncovered potential security threats and dependencies.

## Industry Standard Best Practices

I applied industry standard practices in this program by making sure that my code was up and running with no dependencies, as well as making sure that my code was easy to read and followed typical formatting. Another industry best practice that should be used moving forward while running this program is constant testing and updating the software, which can prevent future bugs from being introduced.