****

# 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** | **8.13.2023** | **Tyler H** |  |

## 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

Tyler Hilliard

## Algorithm Cipher

Advanced Encryption Standard (AES) is a commonly used encryption algorithm that provides strong encryption. It is a symmetric key algorithm, which means that the same key is used for encryption and decryption. AES is widely used in various applications, including government, military, and commercial settings.

Hash functions are used in encryption algorithms to transform data into a fixed-size output, which is typically used as a digital fingerprint or signature of the data. Hash functions can be used to verify the integrity of data, as well as to create digital signatures.

The bit level of a cipher refers to the size of the key used for encryption. The larger the key size, the more secure the encryption. For example, AES-256 uses a 256-bit key, which is considered to be very secure.

Random numbers are often used to generate cryptographic keys or to add randomness to encryption algorithms. Random numbers are important for security, as they make it more difficult for attackers to crack encryption algorithms.

Symmetric keys are used for encryption and decryption by the same key. This makes symmetric keys easier to use than asymmetric keys, but it also makes them less secure. While Asymmetric keys use different keys for encryption and decryption. This makes asymmetric keys more secure than symmetric keys, but it also makes them more complex to use.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a computer

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.

Could not get checksum to run the internal server on my machine

## Secondary Testing

Insert screenshots below of the refactored code executed without errors and the dependency-check report.

No new vulnerabilities introduced with refactored code

A screenshot 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.

Could not get everything to work. CheckSum is lacking.

## Summary

By implementing these code quality practices, the developers of the SslController made the code clearer and easier to understand for devs. This is important because it allows devs to find the information quickly and easily they need to maintain and extend the code. It also makes the code less likely to contain bugs, which can lead to system failures. SHA-256 is a cryptographic hash function that produces a 256-bit hash value. This means that it is virtually impossible to reverse the hash function and find the original data. SHA-256 is a very secure encryption algorithm and is used by many organizations to protect sensitive data. When the developers of the SslController used SHA-256 to encrypt their code, they made it very difficult for anyone to crack the code and steal the sensitive data that it contains. This is because even a small change to the code will result in a completely different hash value, making it impossible to find the original code.

## Industry Standard Best Practices

I referenced code from Github and stack overflow to cobble the SSLController.java together. I used the parts of code that was successfully working for others and verified that it was secure through the Maven. No vulnerabilities were detected with the code used. We intended to implement Sha-256 which is a very secure method of encryption. Using best practices ensures your company is ahead of the new standards coming. If you are already using secure methods then being an early adopter for the next secure method will be easier for the developers as they already have the proactive mindset.