****

# 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** | **10/15/23** | **Lindsey DeLorenzo** | **Thank you for your time.** |

## 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

Lindsey DeLorenzo

## Algorithm Cipher

The algorithm Cipher selected is SHA-256. This is mainly to check the checksum through Windows PowerShell. This is the first method that worked. I originally wanted to have the cipher in AES because of its many benefits. I unfortunately would not find success with this.

## Certificate Generation

## Insert a screenshot of the certificate generated.

A screenshot of a video game

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 computer screen shot of a computer code

Description automatically generated

Cannot connect to local host. I do have the properties encoded after applying the certificate.

## Secondary Testing

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

-A checksum was added to the java main file.

-The version of Spring framework boot was changed to a newer version as the dependency-check suggested.

-I was unable to figure out how to update the Jackson or Apache Log4j for the API

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

## Functional Testing

Insert a screenshot below of the refactored code executed without errors.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

## Summary

I used the higher of message algorithms SHA-256 to encrypt the string data given. This would give the key length of at least 256 bits.

I self-signed a certificate to secure the https connection. I did this through creating a certificate with the property file.

I updated the system in the areas I could find in the code, spring frameworks, to a newer version.

## Industry Standard Best Practices

Security is a major part of coding. It allows both the user and host to remain safe when creating software. Following secure practices will allow us to help the code stay as safe as possible. We can maintain this safety through identifying and correcting vulnerabilities, using safe coding practices, securing libraries through encrypting, implementing safety controls, and monitoring security incidents.