****

# 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** | **6/14/24** | **Zach Chilgren** |  |

## 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

Zach Chilgren

## Algorithm Cipher

I feel that the best encryption algorithm cypher to use for this application would be to use AES-256. It is currently the standard encryption algorithm used by United States Government for over the last 20 years and is also widely considered to be the standard for data security.

## Certificate Generation

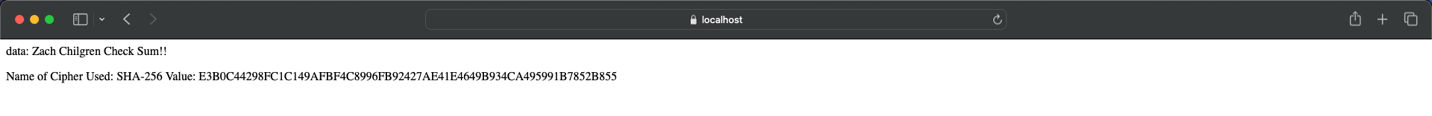
Insert a screenshot below of the CER file.

A screenshot of a certificate

Description automatically generated

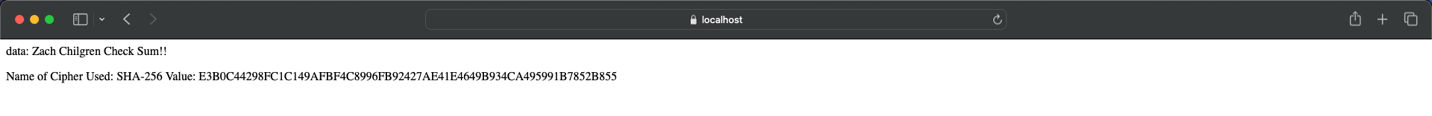
## Deploy Cipher

Insert a screenshot below of the checksum verification.



## Secure Communications

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



## 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

A screen shot of a computer program

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 security that was added to the application was the implementation of self-signed certificates that lets users use HTTPS, as well as also solving vulnerabilities by editing the pom.xml file to suppress any needed errors.

The main benefit to this is that it allows for a secure connection that can be verified to let the user know that they are interacting with the company and not some other individual as well as allow the company to prevent users from accessing one another sensitive data when using the platform

## Industry Standard Best Practices

One main thing to keep up with for ongoing security is to make sure that the latest versions are being used as newer releases of java and other parts of the software can have less security issues than older ones as well as being more likely to continue working properly in the future as software continues to evolve. This can help the overall well-being of the company by keeping users faith that their data is being properly used and protected as well as being able to prevent any negative events from happening for the company that could potentially push away current or future users.