****

# 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** | **2/17/2023** | **Ahmad Omar** |  |

## 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

Ahmad Omar

## Algorithm Cipher

Our client Artemis Financials needs a strong and tested algorithm cipher in which is why I would recommend they use SHA-256 as the algorithm cipher of choice, the reason that I would recommend that they use SHA-256 since it is one of the most secure encryption methods currently out there that has not been cracked and if someone tries to brute force SHA-256 it will take a very long time to do it. The other reason why I recommend they use SHA-256 is that this encryption method is the most used currently by other financial companies and even with the wide use of SHA-256 it has not been cracked which in turn shows us that it is a secure and tested method to use in encrypting the data for Artemis Financials.

The reason that we would want to use SHA-256 is that it will take the data and encrypt it in a way that is unreadable unless you have the key to decrypt it. In our case SHA-256 will take an input such as “Ahmad Omar” and then output it as a bunch of random of numbers that are 256 bits long which means that the chance of any data being the same and colliding is almost 0%. SHA-256 means that it uses 256 bits which means that the quantity of the encryption will be 256 bits long making it much harder to crack the encryption to get into the data.

In encryption we have two types of key types symmetric and non-symmetric keys, with symmetric keys we will use the same key to encrypt our data as well as the same key to decrypt our data, on the other hand a non-symmetric key will have a key that encrypts the data and then another key that will be used to decrypt the data. They are usually used for different things on why you would want to use them, in the case of a symmetric key it is great for encrypting large number of encryptions in a very fast manner, and with non-symmetric keys they are usually slower and are usually used with messages over the internet in which someone shares a public key, but you keep a private key.

As I stated SHA-256 has not been cracked yet and the chance of it being cracked is so small that it is almost a 0% chance, but like anything there is always a chance of it being cracked one day even if it’s a small chance, so experts continue to work on encryption to make them harder and harder to crack.

## Certificate Generation

Insert a screenshot below of the CER file.

Text

Description automatically generated

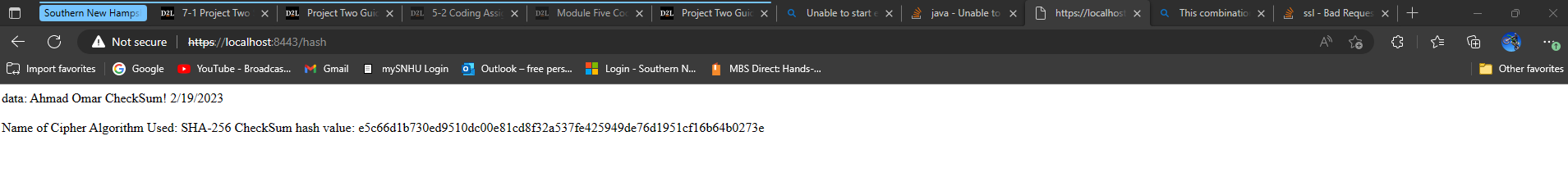
Graphical user interface, text, application, email

Description automatically generated

I was running into some issues when I was trying to do the 8443 port request so I remade my certificate which seemed to have fixed my issue since I am now able to do those calls.

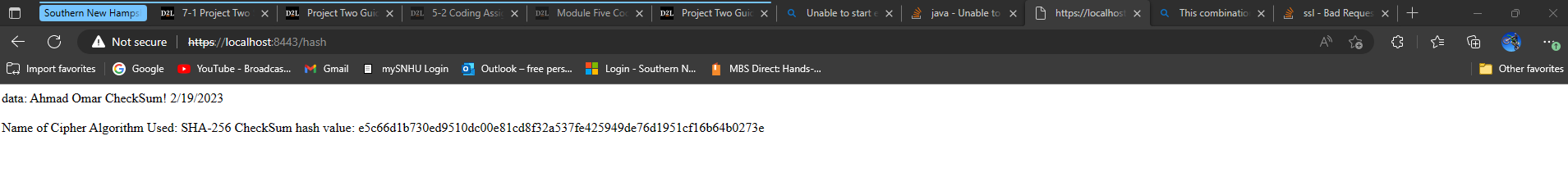
## Deploy Cipher

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.

Graphical user interface, text

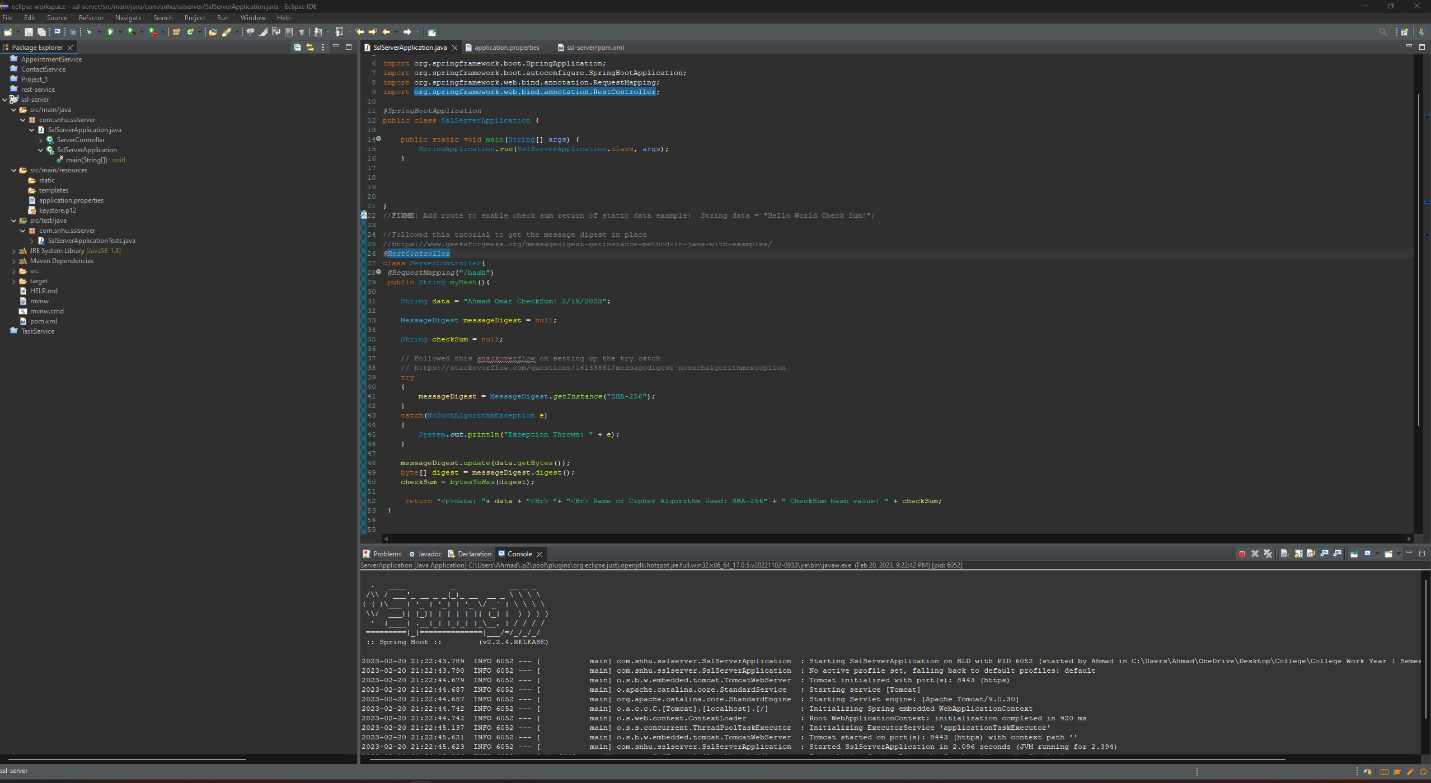
Description automatically generated

Text

Description automatically generated

## Functional Testing

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



## Summary

The code had been refactored to include the use of HTTPS which was an added security measure that we included by using a self-signed certificate which in the actual code base we would get a signed certificate from a reputable certificate authority which will help ensure that our connection is secure and that anyone interacting with the website is able to ensure that the connection is secure and safe.

I also edited the POM.xml file to have it check for owasp to ensure that there where no security vulnerabilities in the frameworks we were using and if the owasp report did find any security issues we were able to fix those to ensure that our software was as secure as possible.

For the use of the HTTPS I had to, as stated above create a self-signed certificate and to ensure that was working correctly I edited the code base we had to do a checksum verification in which I ensured that the data that was being passed through was being changed into the security measure we were using which in this case was SHA-256. By doing this I ensured that the hashing function was working as intended and that the data we showed was being encrypted so that others were unable to read it without knowing the decryption method.

By doing all these steps I was able to ensure the security of the software but also by making sure that our encryption was working as intended.

## Industry Standard Best Practices

The best practice for the industry is to keep checking frameworks for vulnerabilities and updating them as soon as possible to ensure that there are no possible security breaches that can occur due to old or outdated framework that was being used. The best way to ensure this is to have a dedicated IT team that will regularly check for any of these vulnerabilities and update them as needed, but by also having an IT team we will be able to add more protection in the way of user privileges so that they are unable to access information that they shouldn’t have access to.