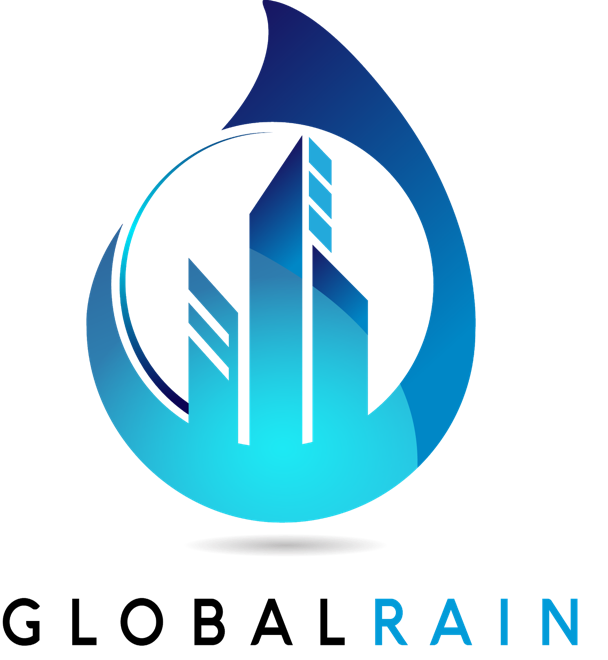
****

# CS 305 Project Two

**Practices for Secure Software Report**

**Table of Contents**

[Document Revision History 3](#_heading=h.30j0zll)

[Client 3](#_heading=h.1fob9te)

[Instructions 3](#_heading=h.3znysh7)

[Developer 4](#_heading=h.2et92p0)

[1. Algorithm Cipher 4](#_heading=h.tyjcwt)

[2. Certificate Generation 4](#_heading=h.3dy6vkm)

[3. Deploy Cipher 4](#_heading=h.1t3h5sf)

[4. Secure Communications 4](#_heading=h.4d34og8)

[5. Secondary Testing 4](#_heading=h.2s8eyo1)

[6. Functional Testing 5](#_heading=h.17dp8vu)

[7. Summary 5](#_heading=h.3rdcrjn)

## Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
| **1.0** | **06/20/21** | **Drew Townsend** | **Initial doc creation** |

## Client



## Instructions

Deliver this completed Practices for Secure Software Report documenting your process for writing secure communications and refactoring code that complies with software security testing protocols.

Respond to the steps outlined below and replace the bracketed text with your findings in your own words. If you choose to include images or supporting materials, be sure to insert them throughout.

## Developer

Drew Townsend

## 1. Algorithm Cipher

The encryption algorithm cipher provides a way to encode data that is being sent to and from the application. Some algorithms are more secure than others, and some take longer than others to encrypt and decrypt. We need to find the balance between speed and security, without taking any shortcuts. The cipher I would recommend is AES 256.

There are various bit levels of the AES cipher, like 128 and 256, but 256 provides the most security while not being too slow. The bit size is the number of random characters available for encoding and decoding.

Hash functions work with public encryption keys, which are unique based on the data being encrypted, to create a hash. This hash is embedded in a certificate, which can be checked to ensure the data has not been tampered with. If a fake or incorrect hash is embedded, the certificate check will show that the data may have been tampered with.

Ciphers fall into two different categories, symmetric and asymmetric. Symmetric ciphers use only private keys, and the same private key is used to both encrypt and decrypt the data. Asymmetric ciphers use a public and private key. The encryptor uses a public key to secure the data. The data cannot be unencrypted unless the receiving application has the private key. However, without the use of random numbers these keys do not matter.

Randomness on a computer has to be simulated, because there is no true way to have a computer pick a random number. Nowadays, it is common for an encryption method to use a cryptographic pseudorandom number generator (CPRNG). This algorithm generates what appears to be random numbers, based on a seed value.

Nowadays, certain industries must comply with regulation on protecting sensitive data. One of these industries is the financial industry, which must do all in its power to prevent a data leak. It is important that the company not only follows these regulations to prevent fines, but also to keep the trust of their clients. The need for secure ciphers has grown exponentially as companies continue sending sensitive information over the internet. Hackers make advancements on cracking some of these ciphers every day, however developers are constantly working to better the algorithms as well.

## 2. Certificate Generation

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

## 

## 3. Deploy Cipher

Text

Description automatically generated with medium confidence

## 4. Secure Communications

Text

Description automatically generated with medium confidence

## 5. Secondary Testing

A picture containing text, screenshot, monitor, indoor

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

## 6. Functional Testing

Text

Description automatically generated

Text

Description automatically generated

## 7. Summary

Due to the code not requiring user input via HTTPS, I did not need to provide input validation. I did add some comments to introduce the program and some of the functions, so developers can understand what I am doing and why.

Regarding the Vulnerability Assessment Process Flow Diagram, I addressed the following areas of security:

* APIs
  + In this project I worked with Spring Boot, an application framework typically used for website development. Spring Boot supports the use of APIs, and therefore I needed to secure any potential security threats coming from user requests to the website.
* Cryptography
  + Within the application, I was asked to recommend and provide a way to encrypt data. This involves cryptography, as the data to be secured uses an encryptor. Although I didn’t provide decryption services, this would also fall under cryptography
* Client/Server
  + In this project I was mostly focused on the backend development for the application, however I did need to provide a client certification for HTTPS, and I needed to send data to the client’s browser from the server.
* Code Error
  + Code error was addressed, as I had implemented error handling into the application. I made sure to catch exceptions for application stability and debugging purposes.
* Code Quality
  + Last, but not least, I addressed code quality. Code quality is inherent to any project where code needs to be planned and written. I used the best practices in terms of security and readability.

It’s important for companies to practice common security techniques and check their applications against vulnerability databases. For Artemis Financial, it’s actually necessary for them to do this, as the government regulates security for the financial industry. Within this application, I did two main things; I forced the client to use HTTPS and I offered an encryption service for sensitive information. HTTPS is a protocol that provides an encrypted connection between clients and a server to send and receive data. This is great and provides security, however in most situations it isn’t enough. For Artemis Financial I also used the SHA-256 algorithm to encrypt data at rest. Sensitive information must be decrypted using a private key before being able to read data. From here forward, Artemis should maintain security by checking against the vulnerability database and updating their software. They should also suppress any false positives, as to not create noise where there is no real issue. Also, if by some miracle SHA-256 is cracked, Artemis should take action to upgrade their encryption algorithm.