****

# Practices for Secure Software Report

**Gage Biros**

**19 February 2023**

**Southern New Hampshire University**

**CS – 305**

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** | **2-19-2023** | **Gage Biros** |  |

## 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

Gage Biros

## Algorithm Cipher

After reviewing all the options for encryption at Artemis Financial, it would appear, based on the needs of the client, that the AES (Advanced Encryption Standard) algorithm cipher. Artemis Financial is looking to encrypt their long-term archive files, which is one of the benefits of this cipher. It is one of the most secure encryption options available, even recommended by the National Institute of Standards and Technology (NIST). The AES algorithm cipher is secure against related-key attacks, brute force attacks, and known plain-text attacks. The cipher uses a block size of 128 bits which provides enough security on that alone. It is known for the symmetric cipher so both the sender and the receiver of the data must have the same key to encrypt/decrypt the sensitive data. It features hash functions, bit levels, and random numbers to encrypt the data. This AES cipher is used by many government agencies, so there is a lot of evidence of the success of this encryption. We do still need to be careful as there are still some security issues with this algorithm cipher we need to consider, but if we adopt best practices, we can ensure all the assets will be protected. We should be concerned if technology advances as advancements in technology could affect how secure all algorithm ciphers are. I do believe this is the most effective for this situation and the best suited encryption cipher for Artemis Financial.

## Certificate Generation

Insert a screenshot below of the CER file.

Text

Description automatically generated

Text

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

I keep getting SpringApplication cannot be resolved. I am unable to resolve this issue. I have worked on this for 2 weeks and can’t figure this out.

## Secure Communications

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

I keep getting SpringApplication cannot be resolved. I am unable to resolve this issue. I have worked on this for 2 weeks and can’t figure this out.

## Secondary Testing

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

Text, letter

Description automatically generated

Text

Description automatically generated

## Functional Testing

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

I keep getting SpringApplication cannot be resolved. I am unable to resolve this issue. I have worked on this for 2 weeks and can’t figure this out.

## Summary

Summary: Discuss how the code has been refactored and how it complies with security testing protocols. In the summary of your practices for secure software report, be sure to address the following:

Refer to the Vulnerability Assessment Process Flow Diagram. Highlight the areas of security that you addressed by refactoring the code.

Discuss your process for adding layers of security to the software application.

To ensure the protection of all systems, we need to implement cyber security processes. As developers in any application, we need to be mindful of how our product could put potential clients at risk for cyberattacks. When we are tasked with developing a product for a client, the integrity of the product isn’t only for the safety of our client, but also it is a direct reflection of our product for future clients, and if we have a lot of clients that have been victims to cyber attacks, then our reputation as a developer will be poor. In this program we have implemented some cryptographic algorithms using hash functions. We are using a state of the art cipher that is relatively lightweight, that provides 128 bit security which is considered adequate for mainstream applications.

The transfer of data is one of the most typical ways hackers can access sensitive data, so we need to make sure every time data is transferred that every single opportunity that hackers can gain access is protected. The process in which we will protect our data is through encryption and decryption of data. The programs will complete this task with a “public key” that is shared between the two nodes. The key will encrypt the sender’s data, and then the key will decrypt the encrypted data for the receiver. This can only happen if the handshaking process is satisfied, which is how the two nodes will confirm their identity so the transfer of data will not happen if someone who isn’t supposed to have access will be denied access.

## Industry Standard Best Practices

We have run some static testing on this code which will show us exactly what opportunities hackers could have if we don’t have our product coded properly. Many issues come from conditions in the functions we use, if the parameters aren’t set up right then hackers can create malicious code through those inputs. We also need to ensure that all of the software and programs we are using to develop this code is up to date, and being supported by their creator; when we implement other programs into our software that is not being maintained by their developers, our code could be put at risk for cyber attacks since there is no one fixing issues in their programs that have been infiltrated by hackers. A lot of this stems back to fundamental software development, and what we consider to be using best practices when it comes to software development.