****

**CS 305 Project Two**

# Practices for Secure Software Report

**Johnny Jones**

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** | **10/15/2024** | **Johnny Jones** | **Completed algorithm cypher** |
| **1.0** | **10/17/2024** | **Johnny Jones** | **Generated certificate, Deploy Cipher, Secure Communication** |
| **1.0** | **10/18/2024** | **Johnny Jones** | **Secondary & Function testing** |

## 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

Johnny Jones

## Algorithm Cipher

After reviewing Artemis Financials software given the potential security vulnerabilities, I would recommend Advanced Encryption Standard (AES) encryption algorithm cipher to protect it. AES is a symmetric encryption algorithm best known for its strength, efficiency, and versatility. It encrypts data in fixed 128-bit blocks and supports key sizes of 128, 192, or 256 bits, making it adaptable for different security needs. It is also resistant to all known cryptographic attacks which makes it suitable to protect Artemis Financial. AES pair well with hash functions such as SHA-256 to ensure data integrity. If at any time the input changes, this would create a different output. As mentioned before AES supports different bit levels. AES-129 is used for general applications with a good balance of security and performance AES-192 offers a higher level of security but lower performance, while AES-256 provides the highest security and is recommended for protecting highly sensitive data. Since Artemis is a financial company, I would recommend bit SHA-256. AES uses random numbers to create unique initialization vectors. Doing this helps maintain the encryption strength and helps prevent pattern detection for hackers. Being that Artemis is Symmetric key algorithm, this means they use the same key for encryption and decryption. This key is always secret and secure especially when being shared. Asymmetric on the other hand uses public and private keys for encryption and decryption. This form of encryption is slower than symmetric and not used for large amounts of data. Over time encryptions has evolved and will continue to evolve to ensure safety.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a computer

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screenshot of a computer

Description automatically generated

## Secure Communications

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

A close up of a number

Description automatically generated

## Secondary Testing

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

A screenshot of a computer program

Description automatically generated

A screenshot 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

## Summary

I refactored the original code by adding a checksum using the SHA-256 algorithm to generate a hash of a static data string. I included a new route, /hash, that calculates the hash value and returns it along with the original data.

The code refactoring followed secure coding practices by implementing SHA-256 hashing for data integrity checks which protects it without directly exposing raw data, this utilized the MessageDigest that Java have built in. By adhering to the principles of separation of concerns, and enhancing security with a standardized hashing mechanism, I was able to improve the security data verification process. Splitting the functionality in a ServerController class which improved the maintainability and security of the program which fits into the code quality of the vulnerability assessment process. Additionally, planning for security from the design phase and systematically mitigating identified risks contributed to a more robust software solution.

I updated the version of the Maven Dependency check version from 5.3.0 to 10.0.4, This ensured that the dependency check was up to date with current practices making it more secure.

The refactored code ensures no sensitive information is exposed, and it adheres to security standards by not storing or transmitting plaintext sensitive data.

## Industry Standard Best Practices

I applied industry standard best practices in several ways. To start, I utilized SHA-256 algorithm for hashing. This is widely accepted and secure within the industry. Using this secure algorithm helps mitigate the risk of data tampering. I kept the server logic separate from the main application by creating a dedicated ServerController class. This is helpful in the industry as it makes it easier to review, and update. Using HTTPS allowed me to encrypt certain information when data is being transferred, this helps protect against certain attacks. The value of applying industry standard best practices includes having a highly secured program, reduces cost, increases quality, protects reputation, and ensure that you are being compliant when it comes to certain industries.