****

# 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** | **06/15/23** | **Ali Alshara** |  |

## 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

Ali Alshara

## Algorithm Cipher

Upon evaluating Artemis Financials' requirements and the overall system, I believe the Advanced Encryption Standard (AES) is the optimal encryption algorithm. AES, known as the "gold standard" for data encryption, originated from the US government in 2001 and is extensively utilized in various everyday applications such as social media and financial transactions. AES ensures robust security as recovering its key would be time-consuming when implemented correctly. AES involves key generation, encryption, and decryption processes. The algorithm can divide data into blocks and applies a network to each block.

Hash functions compress input values, transforming this numerical data into an a much more compressed numerical data. AES offers multiple security options for the company, with key lengths of 128-bit, 192-bit, and 256-bit. Symmetric keys, simpler compared to nonsymmetric keys, require less execution time. For enhanced security, keys must be random rather than easily memorable, as their simplicity makes them susceptible to cracking. AES operates as a symmetric key encryption cipher, distinct from asymmetric encryption where different keys are used for decryption and encryption. It is crucial to understand the disparity between asymmetric and symmetric encryption. AES excels in using longer key lengths to enhance security, but a drawback is that every data block is encrypted in the same manner.

Computer-based encryption has been prevalent since the 1970s when the US government adopted the Data Encryption Standard (DES). DES remained the encryption standard until it was compromised in the 1990s, leading to its replacement by AES, which has been the standard since the early 2000s. Even as early as ancient Egypt there have been some form of encryption used in their messages.

## Certificate Generation

Insert a screenshot below of the CER file.

A picture containing text, screenshot, font

Description automatically generated

A screenshot of a computer

Description automatically generated with low confidence

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screenshot of a computer

Description automatically generated with medium confidence

## Secure Communications

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

A screenshot of a computer

Description automatically generated with medium confidence

## 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 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

Description automatically generated

## Summary

Our application's security was significantly improved by implementing self-signed certificates for HTTPS and addressing vulnerabilities identified during the dependency check. Ensuring appropriate certificate creation was the initial step to establish a secure connection, enhancing the user’s trust and benefiting our business. Verifying the accuracy of our hashing function and using checksums provided reassurance regarding data confidentiality. Lastly, patching vulnerabilities ensured our application's internal operations were up to date, maintaining a secure environment and safeguarding against potential attacks. Regular software and system patching is a crucial practice to ensure application security and protect against vulnerabilities. This also lets the organization protect its users by granting users access based on their specific needs rather than providing unrestricted access to all resources. This approach ensures that access permissions are aligned with the requirements of the program's current state, reducing the risk of unauthorized actions within the organization.

## Industry Standard Best Practices

So by incorporating the above security methods we were able to ensure the well being of the company by mitigating risk of security breaches, data leaks and unauthorized access to our systems. One example is we deployed secure communication protocols such as HTTPS to help encrypt the data being sent from the application to the users. This helps prevent sensitive information from being exposed while getting the user the information they need. The value of applying the industry’s best practice is creating trust between the company and the user as well mitigating risk of a security breach happening. The business’s reputation is also at risk if we don’t implement these practices. It is very important of the health of the company to always use the industries best practices when it comes to secure coding.

References

* Development of the advanced encryption standard - NIST. (n.d.). <https://nvlpubs.nist.gov/nistpubs/jres/126/jres.126.024.pdf>
* Stec, W. by: A. (2023, March 16). *Symmetric cryptography vs asymmetric cryptography*. Baeldung on Computer Science. <https://www.baeldung.com/cs/symmetric-vs-asymmetric-cryptography>