****

# 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 6](#_Toc102040760)

[4. Secure Communications 6](#_Toc102040761)

[5. Secondary Testing 6](#_Toc102040762)

[6. Functional Testing 7](#_Toc102040763)

[7. Summary 8](#_Toc102040764)

[8. Industry Standard Best Practices 8](#_Toc102040765)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **02/17/2023** | **Sulochana Pradhan** | **Final Report** |

## 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

Sulochana Pradhan

## Algorithm Cipher

I would recommend the SHA-256 message digest algorithm for hash generation. Artemis Financial deals with a different range of clients from the private and public sectors. Moreover, they work with national and international clients. As the business model is to help their customer to find the optimal financial plan, the nature of the business is sensitive and always at risk of security threats. There are different hashing algorithms available. We have already seen a reported collision in MD5, and SHA1 algorithms. SHA3 is the latest series of SHA digest algorithms that were developed in response to the successful attack on the SHA-1. As this is the latest algorithm, the hardware and software support might still be lagging. It is much slower than SHA2. The SHA-256 also belongs to the SHA algorithm family which contains SHA-1, SHA-2, and SHA-3. The SHA-256 belongs to SHA-2 and uses 256 bits in generating hash digest. There are 2256 possible hash combinations which are huge numbers for the collision to occur. The number of bits is not only the factor that makes the encryption secure. There are other factors such as the algorithm behind it, randomness, etc. SHA-256 algorithm is one of the recommended hash functions which is used by different financial institutes and the major cryptocurrency Bitcoin. Considering all these factors, the SHA-256 algorithm makes more sense for Artemis Financial.

Randomness creates unpredictability. The quality of the random number determines the difficulty to break the encryption system. It makes sure the secret keys are random and unpredictable. It is used everywhere in cryptography. Even storing the password hash generated by the hash algorithm, it is recommended to use the random salt to prevent the dictionary attack and ensure more security. Symmetric keys algorithms use the same key for encryption and decryption and are typically faster than asymmetric keys algorithms. Online and digital signatures frequently employ asymmetric key techniques, which use both public and private keys to confirm ownership. The earliest evidence of encryption dates to 1900 BC. During World War II, Alan Turing's decipherment of the message generated by the Enigma machine changed the course of history. As technology advances, cryptography, and security risks also change continuously. There will always be an insider trying to weaken the system's security and someone working to strengthen it.

## Certificate Generation

Insert a screenshot below of the CER file.

Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

Graphical user interface, application, Word

Description automatically generated

## Secure Communications

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

Graphical user interface, text, application

Description automatically generated

## Secondary Testing

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

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated

## Functional Testing

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

Text

Description automatically generated

## Summary

Referring to the VAPFD, I have made several changes to make sure the security issues are addressed properly. The main thing is to ensure the message won’t get decoded and altered by an intruder. For that, I have used the SHA-256 cipher to generate the hash and use it instead of the plain text message. Moreover, I have implemented the certificate validation to ensure the client will be communicating with the right web application. The certificate was generated using SHA256withRSA algorithm. We can see the HTTPS protocol instead of HTTP. This encrypts the information so that the intruder won’t be able to decrypt the message even if they can read it. I have made sure the code is logical, simple, and easy to follow. Even if we encounter a defect, it is pinpointed and fix it. Finally, there is not any server information exposed to the client.

## Industry Standard Best Practices

Nowadays, SSL is mandatory for web applications. Artemis Financial deals with the financial information of public and private sector clients. It is essential for them to make sure the data are secure. I would recommend using the proper encryption and security layers to ensure the integrity of the data. For that, they can utilize the cipher algorithm accepted by the NIST. Furthermore, it is essential to check the dependencies of the libraries being used in the application periodically. They can utilize the CI/CD pipeline for this. Writing clean code is also an important factor to prevent security risks. So, I would highly recommend following the industry standard coding standard and checking the code quality using a code analysis tool. It is always important to make sure we prevent SQL injection and XSS attacks while dealing with parameterized requests. Finally, I would recommend they encrypt all sensitive information before storing them in the database or any other storage.

**References:**

* Jena, B. K. (2022, November 11). A Definitive Guide to Learn The SHA-256 (Secure Hash Algorithms). Simplilearn.com. Retrieved February 16, 2023, from <https://www.simplilearn.com/tutorials/cyber-security-tutorial/sha-256-algorithm>
* Jung, J. (2022, August 3). What are Salted Passwords and Password Hashing? Okta, Inc. Retrieved January 25, 2023, from <https://www.okta.com/blog/2019/03/what-are-salted-passwords-and-password-hashing/>
* Lake, J. (2022, March 30). What is a collision attack? Comparitech. Retrieved January 25, 2023, from <https://www.comparitech.com/blog/information-security/what-is-a-collision-attack/>
* How Alan Turing Cracked The Enigma Code. (n.d.). Imperial War Museums. Retrieved January 25, 2023, from <https://www.iwm.org.uk/history/how-alan-turing-cracked-the-enigma-code>
* A Brief History of Cryptography. (2023, January 12). Retrieved January 25, 2023, from <https://www.redhat.com/en/blog/brief-history-cryptography>