****

# 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** | **8/15/2024** | **Jordan Santiago** | **Initial Document Creation** |

## 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

Jordan Santiago

## Algorithm Cipher

I would recommend using the Advanced Encryption Standard (AES) encryption algorithm cipher along with the SHA-256 cryptographic hash for message validation for Artemis Financial because of their strength against being broken by an attacker. AES is the gold standard of encryption and has been used by the government for decades because it is considered invulnerable to all attacks except brute force (Simplilearn, 2024). Similarly, SHA-256 is one of the strongest hash functions available, making the combination one of the most powerful that any application could use. A hash function is essentially a method of verifying that something has not been tampered with by providing the receiver a means to check the original has against the document. Neither AES nor SHA-256 have ever been compromised, and the chances of them becoming compromised are extremely low. This is because they can both utilize a high bit level, 256 bits maximum. To put this into perspective, the Digital Encryption Standard (DES) has a 56-bit key and can be cracked in 362 seconds, while AES-128, not even 256, would take around 36 quadrillion years to crack (Rimkiene, 2022). AES and SHA-256 both use randomization in their processes. Randomizing the data, or the hash, creates a security layer that is unique and unable to be reproduced. AES utilizes an asymmetric key and SHA-256 uses a symmetric key. Asymmetric keys create two private keys, where the only way to decrypt the data is to bring both private keys together. Because both keys are unique, this also significantly strengthens the security of the data. Symmetric keys use a public key and a private key. The public key is readily available to everyone, and the receiver will have a unique key that works for their specific data. This is less secure but uses less resources than an asymmetric key. Encryption has been used as long as there has been secret information that needed to be passed. In a more basic form, this could look like encoded messages or secret languages. Today, encryption is sophisticated to the point where it is almost impossible to break. It can also be used to protect much more than messages, as it can be applied to anything that lives on a computer or network.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a computer

Description automatically generated

A screenshot of a certificate

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.

Although my certificate is now being used the webpage is still showing as not secure. The reason for this is that my certificate is self signed and therefore cannot be validated by the browser. I’ve included a the certificate viewer into the screenshot to show that my certificate is in fact connected to the webpage.

A screenshot of a computer

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

The code has been refactored in a few ways to make it more secure. One of the most significant is the addition of a certificate that can be used to validate the authenticity of the application and enable HTTPS protocol instead of HTTP. Additionally, the usage of SHA-256 for hashing helps to ensure that our data is not tampered with. Using the Maven dependency check, we can continually check for new vulnerabilities and react to them. Lastly, we’ve added additional security by using a secure controller and ensuring there are no errors in the code. These security enhancements address many areas of security including input validation, cryptography, APIs, and code quality.

## Industry Standard Best Practices

Industry standard best practices help to ensure that an application’s security remains intact, even when it is being updated and refactored. Some of these best practices that I used were to use the dependency check to find vulnerabilities, use of a cryptographic hash, adding a security certificate and switching HTTP to HTTPS protocol. All of these items are considered best practices in software development. The value of doing things like that is to ensure that threats cannot view sensitive data or manipulate the application in any way. For the company’s overall well-being, this means increased trust from clients and very likely higher retention rates.

References

Rimkiene, R. (2022) What is AES encryption and how does it work? CyberNews. Retrieved from

https://cybernews.com/resources/what-is-aes-encryption/

Simplilearn (2024) What Is Data Encryption: Types, Algorithms, Techniques and Methods.

*Simplilearn*. Retrieved from https://www.simplilearn.com/data-encryption-methods-article#:~:text=The%20Advanced%20Encryption%20Standard%20(AES,for%20very%20demanding%20encryption%20purposes.