****

# 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** | **12/10/2022** | **Adam Sissoko** |  |

## 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

Adam Sissoko

## Algorithm Cipher

Since we are are working with a large-scale project, I feel the best encryption algorithm cipher to use would be AES, which is shorthand for Advance Encryption Standard. Advanced Encryption Standard has been used by the United States Government since 2001, over two decades. It has stood the test of time, and proven itself to be a very useful cipher for large organizations. Within both government agencies and private companies, AES is regarded as the best way to encrypt data. (Crawford).

The Secure Hashing Algorithms, or SHA for short, are a set of commonly used hash functions. SHA can be used with Advanced Encryption Standard to enhance user security, and make it even more difficult for hackers to access sensitive data. SHA accepts input as plaintext and turns it into ciphertext is nearly impossible to decipher. Among algorithm ciphers, the bit level tells you that a hacker would need to perform 2^x operations ( x= bit level) before decrypting the cipher. Advanced Encryption Standard has three bit levels, 128, 192, and 256.

Advanced Encryption Standard is a symmetric key encryption cipher, so you use to same key to encrypt data, and then decrypt it later. Symmetric encryption is usually easier to use than asymmetric methods since you only need to remember one key. AES encrypts every data block in the same fashion, with varied key lengths, leading to more user security.

Humans have been finding ways to encrypt data for thousands of years, with the earliest examples being found in ancient Egypt, where wealthy people would have secret codes written on their tombs to conceal information about their personal lives. Modern, compute-based encryption started in the 70s with the creation and later widespread use of the DES.

## Certificate Generation

Graphical user interface, text, application

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, application, Word

Description automatically generated

## Secondary Testing

Graphical user interface, application

Description automatically generated

## Functional Testing

A screenshot of a computer

Description automatically generated with medium confidence

After functional testing, the main issues I found were the password used in the application.properties file, and the hash function’s hard-coded plaintext input string.

## Summary

1. APIs: I addressed APIs by using HTTPS and adding a usable browser interface that the users can see.

2. Cryptography: I addressed cryptography by using encryption algorithm ciphers, hash functions, and checksum verification.

3. Client/Server: Although the client and server were the same during this application’s testing phases, this step was addressed, since data was sent from the client to the server in order to get displayed.

4. Code Error: I addressed code error by using exceptions in the class, such as a NoSuchAlgorithm exception in the myHash method.

Code error was addressed by utilizing exceptions within our class, mainly a NoSuchAlgorithm exception within the myHash method.

5. Code Quality: I checked the code to ensure that it was functional and easily readable.

The biggest security change I made was adding self-signed certificates that allowed for me to use https. I also made changes to the pom.xml file in order to fix all of the issues detected during the dependency check.

The first step of this process was making sure the certificates were set up properly. This helped me ensure that the company would be able to use HTTPS once the application was ready. This adds to the company’s wellbeing by enhancing user security.

The second stage was verifying that the hash function was working properly. I did this by using checksum. This improves our company’s wellbeing by ensuring that user data is being hashed properly, and that it is difficult for hackers to access it.

The last stage was performing a thorough check to make sure that all vulnerabilities were eliminated. This ensures that he company and it’s users are safe from attacks, and allows us to make sure that our application is working as designed.

## Industry Standard Best Practices

One crucial best practice is checking code for vulnerabilities and exploits frequently to make sure that everything is working smoothly. This prevents malicious users from compromising our systems. It is also important to make sure that users are only given access to information that they need. If people receive too much information about how the program works, it will be easier for them to exploit flaws in the system’s security.

**References**

Crawford, D.. (2019, February 4). *AES Encryption: Everything you need to know about AES.* Retrieved December 11, 2022, from AES Encryption: Everything you need to know about AES.

Bernstein, C., & Cobb, M. (2021, September 24). *What is the Advanced Encryption Standard (AES)? definition from searchsecurity*. Security. Retrieved December 11, 2022, from https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard

Daniel, B. (2022, March 24). *Symmetric vs. asymmetric encryption: What's the difference?* Trusted Computing Innovator. Retrieved December 11, 2022, from https://www.trentonsystems.com/blog/symmetric-vs-asymmetric-encryption

Team, T. (2022, January 31). *The history of encryption: The roots of modern-day cyber-security*. Tresorit Blog. Retrieved December 11, 2022, from https://tresorit.com/blog/the-history-of-encryption-the-roots-of-modern-day-cyber-security/

Encyclopædia Britannica, inc. (n.d.). *Cryptographic hieroglyphic writing*. Encyclopædia Britannica. Retrieved December 11, 2022, from https://www.britannica.com/topic/hieroglyphic-writing/Cryptographic-hieroglyphic-writing