****

# 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** | **February 23, 2023** | **Mihir Patel** | **Modification and Research** |

## Client



## Instructions

## Send in this finished Practices for Safe Software Report that details your procedure for creating secure communications and reworking code in accordance with software security testing guidelines.

## In your own words, respond to the stages listed below and change the bracketed language. Make sure to place any photographs or other materials you decide to include throughout your presentation.

## Developer

Mihir Patel

## Algorithm Cipher

The Advanced Encryption Standard (AES), specifically AES-256, is the encryption algorithm that I believe is most suited given the scope of what we are working with. The Data Encryption Standard was replaced by AES in 2001, and it is now the encryption standard used by the US government. The "gold standard" of data encryption, AES is widely utilized in both the public and private sectors.

Common hashing algorithms are the Secure Hashing Algorithms (SHA). When combined with AES, SHA can increase security since it converts inputted data into ciphertext (Crane), which is highly unlikely to be cracked. The bit level in security informs us that a cipher with that bit level would need 2n operations to decipher (n is the bit level). AES uses one of three bit sizes 128, 192, or 256 with AES-256 being the most secure.

AES uses the same key for both encryption and decryption, making it a symmetric key encryption cipher (Crawford). Asymmetric encryption, which uses distinct keys for encryption and decryption, is thought to be more difficult to use and implement than symmetric encryption. Every data block is encrypted using the same method by AES, although security is increased by using longer key lengths. A key that is simple for one person to memorize is equally simple for an attacker to decipher, hence keys must be random (Martin).

With the development and widespread use of the DES in the 1970s, computer-based encryption has existed. Since Julius Caesar used a substitution cipher around 60 B.C.E., encryption has been around for millennia. However, encryption algorithms have changed over time, from Charles Wheatstone's Play Fair Cipher in 1854 to Arthur Scherbius's Enigma machine in 1918 to the very standard of encryption algorithms we have today.

## Certificate Generation

## Table Description automatically generated with medium confidence

Graphical user interface, text, application, email

Description automatically generated

## Deploy Cipher

Graphical user interface, text, application, email

Description automatically generated

## Secure Communications

## Not able to make it secure.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text

Description automatically generated

## Secondary Testing

Graphical user interface, text

Description automatically generated  
  
Graphical user interface, text, application, Word

Description automatically generated

## Functional Testing

Graphical user interface, text

Description automatically generated  
  
Graphical user interface, text, application

Description automatically generated

## Summary

Self-signed certificates that enabled the usage of HTTPS were the key security measure implemented to our program. To ensure that any vulnerabilities found during the dependency check were fixed, we additionally refactored the pom.xml file. Making sure the certificates were created correctly was the first step in my procedure so that once our application was up and running, we could use HTTPS. By ensuring that our website is secure and users can be certain that they are dealing with us and not an imposter, this security enhances the health of our business. The last step was to check to see if our hashing function was operating correctly and to confirm this using the checksum. Our organization benefits from this security since it gives us peace of mind knowing that our users' data is correctly scrambled and difficult to recover. Making sure that all vulnerabilities were patched was the last step. Having that security guarantees that our business has all of our bases covered and that all of the internal workings of our application are current and functioning as intended.

## Industry Standard Best Practices

One best practice for preserving the security of our application is patching our programs and systems to make sure everything is current. This makes sure that hackers cannot take advantage of outdated systems. The principle of least privilege must be upheld. The organization is protected from attacks from within the group by ensuring users only have access to what they require rather than providing everyone access to everything, even if it is not possible with the current state of our application.