****

# 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** | **12/5/2023** | **Mary Eakins** |  |

## 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

Mary Eakins

## Algorithm Cipher

## Our client is a financial institution that is growing into the digital sector. They are modernizing their practices by integrating an API into their public interface. Security has been a major focus for them from the very beginning. Now they turning to us for a secure solution for file verification over the web. This is an important piece of their overall security model. We will offer our recommendations for secure ciphers and hash functions, and detail what we think is the correct choice for Artemis Financial.

## To begin, hashing and encryption are not interchangeable. In broad-strokes, they are actually quite different. Encryption turns plain text into cipher text, which is incomprehensible to third parties. This is done with the use of keys. The receiving party can then decrypt the cipher text to discern the original plain text message. Alternatively, hashing is “a technique to generate a unique fixed-length string… strictly depending on the specific input” (Advocate & Andrea Chiarell, 2022). This means that the process is irreversible, and no two inputs should generate the same output. Algorithm ciphers are ideal for confidentiality, while hashing is more suited to maintaining the integrity of the data. For this reason, I suggest we employ an encryption algorithm – AES - for Artemis’ large data files, while turning to hashing -with SHA-256 - for verification of their file transfers.

## AES stands for the “Advanced Encryption Standard” and has been hailed as the standard by many organizations, such as NIST and the US Government. It is a symmetric key algorithm, meaning one private key is used to both encrypt and decrypt the data. AES offers three key sizes, 128, 192, and 256. As they increase numerically, so do they in security, alas at the sacrifice of speed. Because Artemis Financials’ data requires the strongest security available, I am recommending the 256 bit key.

## SHA-256 is a member of the “Secure Hashing Algorithm” family. SHA-2 has superseded SHA-1 as the standard for hashing, with SHA-3 already at play as well. Like AES, there are different levels available, which are SHA-256, SHA-384, and SHA-512. The bit number represents the size of the hash that is produced by the hashing process. SHA-256 is sufficient for most digital signatures and online transactions. It also is not susceptible to length-extension attacks, like SHA-512.

## Encryption surely dates back to the advent of writing. There has always been some desire to hide sensitive information from prying eyes. It has religious roots in Egypt, and military roots in Greece and Rome. Obviously the creation of the computer spurred cryptography into a new era, especially with World Wars raging around the globe. Encryption is much more widespread now, and happens all around us. There is much data being passed around, in both micro transactions and macro, that the science of cryptography has had to keep pace accordingly (Team, 2023).

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a computer

Description automatically generated

## Deploy Cipher

Here is the checksum after deploying the cipher. You can see that the certificate in use is my own self-signed one, which localhost doesn’t consider secure, hence the security warning.

A screenshot of a computer

Description automatically generated

## Secure Communications

A screenshot of a certificate

Description automatically generatedHere is my certificate:

## Secondary Testing

My code executed without errors:

A screenshot of a computer

Description automatically generated

Here is the results of the Dependency-Check

A screenshot of a computer

Description automatically generated

## Functional Testing

A screenshot of a computer

Description automatically generatedInsert a screenshot below of the refactored code executed without errors.

A close-up of a computer screen

Description automatically generated

## Summary

We added security to this application in several ways. We ran a dependency check, which provided us with a report of all the known vulnerabilities associated with our current dependencies. We updated the API to use HTTPS and a self-signed certificate. HTTPS is the standard for confidentiality and integrity of web browsing. We will need to upgrade to using a CA in production code, of course. We also sought to employ secure coding practices, such a try-catch block, which will handle any unplanned exceptions in a safe way. We used the MessageDigest class to generate a checksum, using SHA-256 to create a unique hash of our sensitive data (in this case my name). Another important point is keeping all of our dependencies up-to-date, as this will often fix the problem of many vulnerabilities.

## Industry Standard Best Practices

The standards we have followed have been set by OWASP, Oracle, and NIST. We researched and chose the best and most widely supposed algorithms at this time, AES and SHA. Also we implemented HTTPS over HTTP, which is a huge standard in today’s age of security. We brought Artemis Financial into the modern age while also keeping a keen eye on security, and I think we have provided them with a very firm foundation on which to continue building their security plan.

Citations

Advocate, A. C. D., & Andrea ChiarelliPrincipal Developer AdvocateI have over 20 years of experience as a software engineer and technical author. Throughout my career. (2022, March 2). *Encoding, encryption, and Hashing*. Auth0. https://auth0.com/blog/encoding-encryption-hashing/

Team, T. (2023, November 23). *History of encryption: Roots of modern-day cybersecurity*. Tresorit Blog. https://tresorit.com/blog/the-history-of-encryption-the-roots-of-modern-day-cyber-security/