Southern New Hampshire University

Project Two

Emily Nagorski

Dr. Lyon

CS305-T1165 – Software Security

10/13/22

****

# Practices for Secure Software Report

Table of Contents

[Document Revision History 4](#_Toc102040754)

[Client 4](#_Toc102040755)

[Instructions 4](#_Toc102040756)

[Developer 5](#_Toc102040757)

[1. Algorithm Cipher 5](#_Toc102040758)

[2. Certificate Generation 6](#_Toc102040759)

[3. Deploy Cipher 6](#_Toc102040760)

[4. Secure Communications 6](#_Toc102040761)

[5. Secondary Testing 8](#_Toc102040762)

[6. Functional Testing 10](#_Toc102040763)

[7. Summary 11](#_Toc102040764)

[8. Industry Standard Best Practices 12](#_Toc102040765)

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.0** | **10/13/22** | **Emily Nagorski** | **file verification security plan** |

## 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

Emily Nagorski

## Algorithm Cipher

After reviewing the client’s requests, and potential algorithm ciphers, I believe AES 256 would best fit the needs of the client at this time. AES 256 is also known as the Rinjndael algorithm created by Joan Daemen and Vincent Rijmen. (Oracle) This is a commonly used encryption tool, and can be found in three different key sizes, 128, 192 and 256. AES is a symmetric encryption system, meaning that the same key encrypts and decrypts the secure files, as appose to an asymmetric key where the initial user and receiving user do not need the same key. (Ahmad 2022) The only difference in key size is the number of rounds used to encrypt and decrypt, therefore making 256 more secure than 128, this however does not imply that 128 is not secure. (Tobias 2022) AES is widely regarded at the safest encryption tool and is commonly used by government agencies and other companies with classified data and material. (Crawford 2019) I am recommending 256-bit for our client because with the number of combinations it would take to try and brute force attack it, it is nearly impenetrable. (Crawford 2019) The history of encryption can be traced back long before computers and other technology evolved. Around the 20th century encryption really took off and was frequently used in war time communications with tools such as the enigma machine. (Crawford 2019) NIST, or the national institute of standards and technology adopted AES all the way back in 2001 and is still being used today. (Crawford 2019) AES remains the industry standard, and I believe it is the best fit for the client’s needs at this time.

## Certificate Generation

Insert a screenshot below of the CER file.

Text

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

Graphical user interface, text, application

Description automatically generated

## Secure Communications

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

I tried numerous ways to create a secure local connection. A few of the ways I tried were following the steps provided for the previous assignments, additionally I tried making a key pair, I tried to change my browser settings to trust a localhost. After reading many articles, and questions and responses on stack overflow, it was clear that this is a common issue, and from my understanding the certificate is not made to be trusted by local host.

Table

Description automatically generated with medium confidence

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.

Text

Description automatically generated

Graphical user interface, text, website

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

First run

Graphical user interface, text, application, email

Description automatically generated

After refactoring

## Functional Testing

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

Text

Description automatically generated

Text

Description automatically generated

I do not understand the purpose of a manual review for this portion of the assignment. As outlined in the project, the code we added and refactored needed to be done so in a way that did not add to the already existing vulnerabilities. As shown in the initial and final dependency checks that are submitted with this assignment, no additional vulnerabilities were created or found.

## Summary

Throughout the process of working with our client and their security needs many of the concepts from the vulnerability assessment process flow diagram were factored in. Of course, I took into consideration the client and server needs by generating a certificate and using SSL protocol insuring that communication is secure. With the inclusion of the SHA-256 hash function and checksum verification we were ensuring that cryptography was taken into consideration. And as always with and development or security work code quality is of the utmost importance. It was important during the process of working on this client’s product that I was not introducing any additional security risks through the addition of any code, and the refactoring of existing code. We were able to make sure security was maintained by running numerous OWASP dependency check reports. Additionally providing and maintaining clean code that is clear for other developers to step in and work on in the future. Security is important for all companies so that their clients and customers know they can trust them with their personal information. A trusted company is sure to be a successful one, and security is a great place to start when working to gain trust. Best practices to maintain the current security software would include security awareness training for all employees, frequent code reviews to ensure there are no new security threats, and continuing the use of well maintained libraries. (Tran 2021)

## Industry Standard Best Practices

I was able to maintain the application’s current security by writing clean and concise code. Throughout the process of adding and refactoring I also made sure to run multiple OWASP dependency check reports to verify that I was not introducing any additional vulnerabilities to the existing application. Industry best standards overall benefit the company by ensuring the code is clear and concise so anyone can come in, understand what it does and update and maintain it. Additionally best practices ensure that there is no unused or redundant code in the application that could cause a security concern. If industry standard best practices are followed you know the application will be secure, can easily be tested regularly and therefore will ultimately protect the company and the customers of that company.

Citations

Ahmad, K. (2022, June 20). *What is AES-256 encryption? how does it work?* MUO. Retrieved October 13, 2022, from https://www.makeuseof.com/what-is-aes-256-encryption-how-does-it-work/

Crawford, D. (2019, February 4). *AES encryption: Everything you need to know about AES*. ProPrivacy.com. Retrieved October 14, 2022, from https://proprivacy.com/guides/aes-encryption

Oracle. (n.d.). *Java Security Standard Algorithm Names*. Java security standard algorithm names. Retrieved October 14, 2022, from https://docs.oracle.com/javase/9/docs/specs/security/standard-names.html#cipher-algorithm-names

Tobias, E. (2022, February 11). *128 or 256 bit encryption: Which should I use?* Ubiq. Retrieved October 13, 2022, from <https://www.ubiqsecurity.com/128bit-or-256bit-encryption-which-to-use/#:~:text=AES%2D128%20and%20AES%2D256%20use%20an%20almost%20identical%20encryption,and%20AES%2D256%20uses%2014>.

Tran, T. (2021, November 29). *10 best practices for software development security*. Top Software Outsourcing Company in Vietnam - Orient Software. Retrieved October 16, 2022, from https://www.orientsoftware.com/blog/software-development-security/

Other webpages used in problem solving the localhost issue

<https://stackoverflow.com/questions/7580508/getting-chrome-to-accept-self-signed-localhost-certificate>

<https://www.baeldung.com/spring-boot-https-self-signed-certificate>

<https://stackoverflow.com/questions/35565278/ssl-localhost-privacy-error>

<https://dgu2000.medium.com/working-with-self-signed-certificates-in-chrome-walkthrough-edition-a238486e6858>