****

# 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** | **4/14/2023** | **Willem Kroeger** |  |

## 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

Willem Kroeger

## Algorithm Cipher

I would recommend using the RSA stored within the Java encryption library.

The RSA encryption algorithm cipher is a widely used asymmetric key encryption algorithm that is based on the mathematical properties of incredibly large prime numbers. It was invented by Ron Rivest, Adi Shamir, and Leonard Adleman in 1977 and is named after their initials. The RSA algorithm is used for secure data transmission and is widely used for secure communication over the internet.

RSA encryption involves a public key and a private key. The public key is used for encryption and is widely distributed. The private key is used for decryption of messages and relies on the secrecy of the private key. The RSA algorithm relies on the fact that it is computationally infeasible to factor the product of two large prime numbers, making it mathematically secure by preventing the public key from revealing the private key’s decryption.

Hash functions are often used in conjunction with RSA encryption to ensure data integrity. Hash functions are used to generate a fixed-length output (the hash) from an input of arbitrary length. A common hash function used with RSA encryption is the Secure Hash Algorithm (SHA), which generates a hash of either 160, 256, 384, or 512 bits.

Random numbers are used in RSA encryption to generate public and private keys. The choice of random numbers is critical to the security of RSA encryption, as the security of the encryption relies on the difficulty of factoring the public key to determine the private key.

## Certificate Generation

Insert a screenshot below of the CER file.

Graphical user interface, text, application

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

Graphical user interface, text, application, email

Description automatically generated

## Secure Communications

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

Graphical user interface, text, application, email

Description automatically generated

## Secondary Testing

Insert screenshots below of the refactored code executed without errors and the dependency-check report.

Graphical user interface, text, application, email

Description automatically generated

## Functional Testing

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

Text

Description automatically generated

## Summary

With the refactored code, I have addressed the API security by involving RSA encryption, allowing only those with private keys to access the API. The RSA cryptography also addresses encryption use and is widely known to be secure.

Adding layers of security to the software is an iterative process. Once one layer is secure, I moved on to the next layer, while making sure that any new layers don’t circumvent existing layers.

## Industry Standard Best Practices

I followed standard naming conventions and practices and ran both static and dynamic testing and dependency checks to identify risks in standard Java libraries. To maintain security, regularly updating libraries and dependencies to eliminate known vulnerabilities is a highly advised process. Error handling and input validation are also necessary to avoid information leaks and access to sensitive data. Additionally, the performance of regular security audits and dependency checks with penetration testing can identify vulnerabilities before they can be exploited by outside attackers.

Applying best practices for secure coding can help prevent costly cyberattacks. Companies that fail to implement a robust cybersecurity posture can risk costly data breaches, fines, reputation loss, and loss of customer trust. Following industry standard best practices can ensure that software is secure and sensitive data is protected appropriately.