****

# 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** | **April 25, 2024** | **Victoria Keyser** |  |

## 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

Victoria Keyser

## Algorithm Cipher

The Cipher encryption algorithm is a cryptographic technique used to secure data by transforming it into an unreadable format. It does this by substituting or transposing characters according to a specified key, making the encrypted data unintelligible without the corresponding decryption key. There are two main types of ciphers: symmetric and asymmetric. Symmetric ciphers use the same key for encryption and decryption, while asymmetric ciphers use different keys. Examples of symmetric ciphers include the Data Encryption Standard (DES) and the Advanced Encryption Standard (AES), while examples of asymmetric ciphers include RSA and Elliptic Curve Cryptography (ECC). The effectiveness of a cipher depends on several factors, such as key length, algorithm complexity, and resistance to cryptographic attacks.

Hash functions are critical to the Cipher encryption algorithm for verification of key generation and data integrity. They facilitate the generation of cryptographic keys from inputs such as passwords, enhancing security by obviating the need to store or transmit sensitive information directly. Hash functions also ensure data integrity by producing fixed-size hash values, which are compared before and after data transmission or storage to detect any alterations. At the binary level, hash functions process data as binary strings, yielding fixed-size hash values typically represented as sequences of bits. This essential functionality underscores the foundational role of hash functions in bolstering security and reliability within the Cipher algorithm framework.

Random numbers play a crucial role in the context of encryption algorithms like Cipher. They generate keys and initialization vectors, adding an element of unpredictability essential for security. Random numbers ensure that cryptographic keys are unique and challenging to guess, enhancing the resilience of encryption against brute force attacks. Symmetric encryption algorithms like AES use the same key for encryption and decryption, providing fast and efficient data encryption. However, managing and securely distributing symmetric keys can be challenging, especially in distributed systems. Non-symmetric or asymmetric encryption algorithms such as RSA address this challenge by using pairs of public and private keys. The public key is encrypted, while the private key is kept secret and used for decryption. This asymmetric key approach simplifies key management and enables secure communication between parties without prior key exchange. However, asymmetric encryption tends to be slower and computationally intensive than symmetric encryption. As a result, a common practice is to use symmetric encryption for bulk data encryption and asymmetric encryption for key exchange and digital signatures, leveraging both approaches' strengths to balance security and performance in cryptographic systems.

Encryption algorithms have evolved for millennia, from ancient techniques like substitution ciphers to modern advancements driven by computer science and cryptography. Early 20th-century developments such as the Data Encryption Standard (DES) gave way to stronger algorithms like the Advanced Encryption Standard (AES), which remains widely used today. Concurrently, asymmetric encryption algorithms like RSA and Diffie-Hellman emerged for secure key exchange. Recent efforts focus on post-quantum cryptography to resist quantum computer attacks. Encryption algorithms are pivotal in cybersecurity, safeguarding data across platforms, and continue to evolve in response to emerging threats and technological progress.

## Certificate Generation

Insert a screenshot below of the CER file.

A computer screen with white text

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screenshot of a computer

Description automatically generated

## Secure Communications

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

A screenshot of a computer

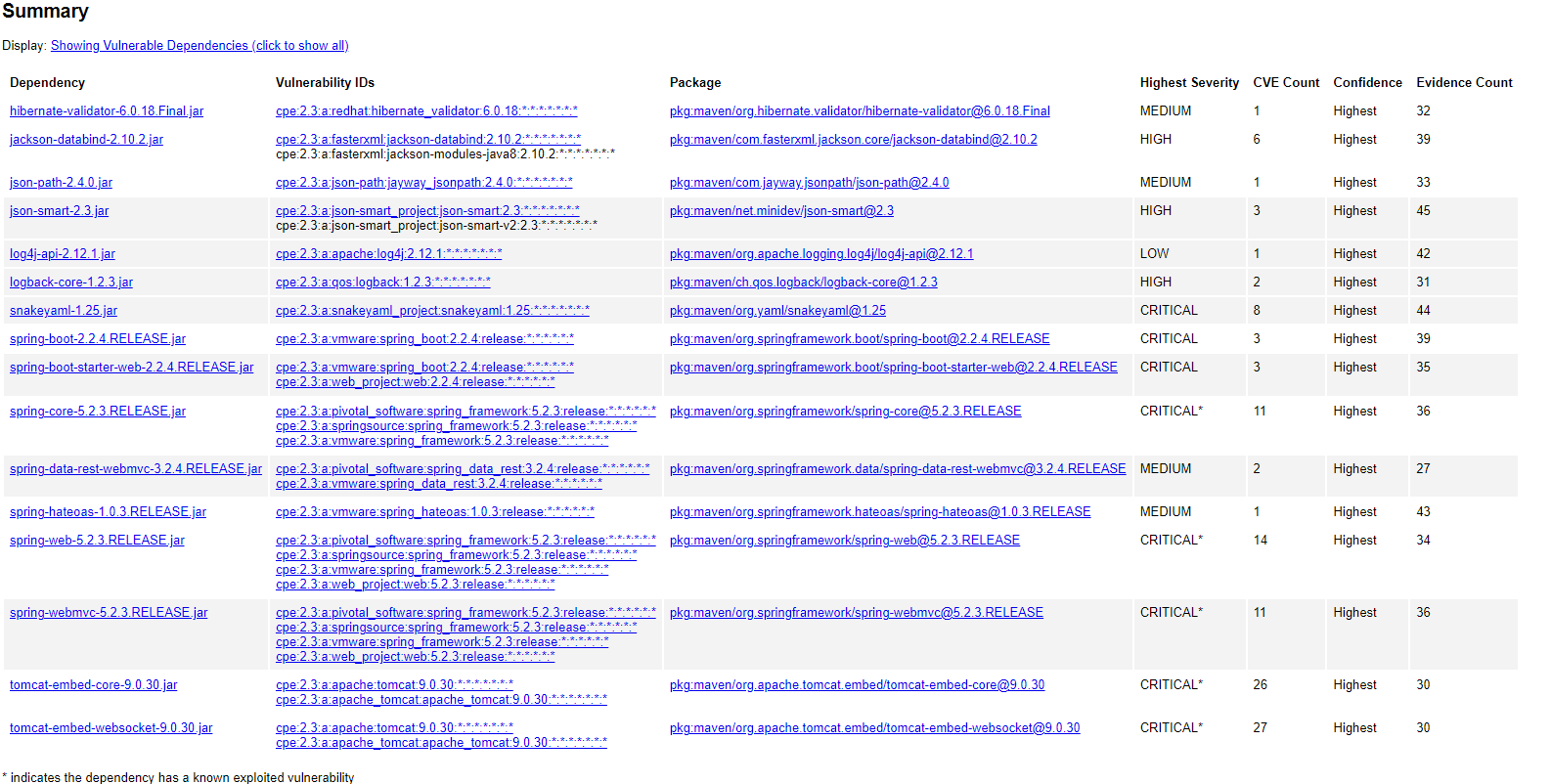
Description automatically generated

## Secondary Testing

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

A screen shot of a computer program

Description automatically generated



## Functional Testing

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

A screen shot of a computer program

Description automatically generated

## Summary

In refactoring the code, I meticulously followed security testing protocols outlined in the Vulnerability Assessment Process Flow Diagram. By implementing cryptographic hash algorithms, I ensured that sensitive data underwent secure processing, mitigating potential risks associated with data manipulation. Additionally, upgrading our communication protocol from HTTP to HTTPS significantly enhanced the security of data transmission, aligning with best practices for safeguarding against unauthorized access and data breaches. Furthermore, I conducted comprehensive static testing using OWASP dependency check to identify and rectify vulnerabilities within our software dependencies, thereby adding layers of security to the application. These proactive measures were instrumental in fortifying the safety of our software and maintaining compliance with our security guidelines.

## Industry Standard Best Practices

I took measures to maintain the software application's current security by implementing industry-standard best practices to mitigate known vulnerabilities. I employed proper authentication and access control mechanisms to regulate access to sensitive resources within the application. In addition, I utilized encryption algorithms to protect sensitive data during transmission and at rest, which aligns with data security industry standards. Regular code reviews and testing, including static analysis tools and penetration testing, were crucial to identify and address potential vulnerabilities proactively.

Secure coding practices have numerous benefits for the company's overall well-being. First and foremost, they reduce the risk of security breaches and data leaks, protecting the company's sensitive information and intellectual property. Maintaining a secure software application builds trust and credibility with its customers, fostering long-term relationships while mitigating reputational damage. Moreover, adhering to industry-standard best practices demonstrates the company's commitment to regulatory compliance and data protection laws, minimizing legal liabilities and potential fines. Finally, prioritizing secure coding practices contributes to the company's resilience against cyber threats, ensuring business continuity and sustained growth in an increasingly digital landscape.