****

# 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** | **08/13/2023** | **Rayyan Abdulmunib** |  |

## 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

Rayyan Abdulmunib

## Algorithm Cipher

Going through the algorithm ciphers provided in Java Security Standard Algorithm Names, I believe the Advanced Encryption Standard (AES) to be the best cipher for Artemis Financial. The AES is a popular symmetric-key encryption technique for protecting sensitive data. Moreover, the National Institute of Standards and Technology (NIST) chose it in 2001 to replace the earlier Data Encryption Standard (DES). AES is a block cipher, which means it encrypts data in fixed-size blocks of 128 bits or less. The algorithm transforms the input plaintext into ciphertext through a sequence of substitution and permutation operations known as rounds. The number of rounds utilized depends on the key size, with 128-bit keys using 10 rounds, 192-bit keys using 12 rounds, and 256-bit keys using 14 rounds. Lastly, AES's strength lies in its resistance to numerous attacks, such as brute force, differential, and linear attacks. It is also quick and efficient, making it appropriate for a variety of applications (National Institute of Standards and Technology [NIST], 2001).

Some of the reasons I believe Advanced Encryption Standard (AES) would be the best cipher for Artemis Financial, being a financial institution, are the AES having robust encryption, efficiency, is commonly used, is regulatory-compliant. Firstly, AES is a very strong encryption technique that has been carefully tested and confirmed to be resistant to a variety of attacks. This safeguards the institution's critical financial data against illegal access. Secondly, AES is a fast and efficient technique for encrypting and decrypting data, making it appropriate for use in high-volume financial transactions. Thirdly, AES is a popular encryption standard that is supported by a wide range of software and hardware providers. This ensures that the financial institution's existing systems can be easily integrated with AES. Furthermore, most financial institutions like Artemis Financial utilize AES to protect sensitive information such as financial transactions, personal data, and intellectual property. AES is a highly secure and efficient encryption technique that has been approved for use in protecting sensitive information by the US government. To prevent unwanted access and protect the confidentiality and integrity of their data, most financial institutions utilize AES to encrypt their online transactions, mobile banking, and other electronic communications. Lastly, many regulatory agencies, such as the Payment Card Industry Data Security Standard (PCI DSS), require financial data to be encrypted. AES is an encryption standard that can assist the institution in meeting these standards (Gritzalis & Katsikas, 2003).

The Advanced Encryption Standard (AES) encryption is designed to be versatile and flexible, allowing it to comply with the most recent regulatory rules. The AES cipher can be upgraded to ensure compliance as new legislation and standards are released. Changes to the key length, mode of operation, or other factors may be necessary to meet the requirements of the new standards. The National Institute of Standards and Technology (NIST) in the United States, for example, has produced standards for the use of cryptography in government systems, including precise requirements for key sizes and encryption algorithms. To meet these needs and remain consistent with government rules, the AES cipher can be changed (Gritzalis & Katsikas, 2017).

Artemis Financial may use Advanced Encryption Standard (AES) cipher to encrypt critical information such client credentials, account information, and transaction data. To prevent unauthorized access to the data, the encryption keys must be properly handled. Additionally, they can utilize AES to protect communication channels including email, texting, and file transfers. This can aid in the prevention of data breaches and the protection of customer data. Also, Artemis Financial, like many other financial institutions, provides mobile banking services that allow consumers to access their accounts and conduct transactions from their mobile devices. The AES algorithm can be used to encrypt data exchanged between the mobile device and the financial institution's computers, assuring the transaction's security. To end, in order to store consumer data, Artemis Financial may use cloud storage solutions. AES can be used to encrypt data on the cloud, adding an extra layer of security.

Although Advanced Encryption Standard (AES) is widely used and considered to be a highly secure cipher, it has its risks as well. To start, Poor key management is one of the most serious hazards connected with the AES cipher. If the encryption key is not correctly handled, it may be compromised, resulting in the decryption of sensitive data. Next, AES is subject to side-channel attacks, which target flaws in the algorithm's implementation rather than the algorithm itself. An attacker, for example, could utilize timing or power analysis to obtain the encryption key. Last, The AES cipher's strength is related to the length of the encryption key utilized. Encryption may be vulnerable to brute-force assaults if a shorter key length is utilized (Gritzalis & Katsikas, 2017).

There are a couple reasons that come to mind as to why sometimes the best or most secure cipher may not be used. One, the performance aspect of it. More secure ciphers frequently necessitate more computing power, slowing down the encryption and decryption processes. This can have a significant impact in some circumstances, particularly in high-volume environments such as financial transactions or network connections. Two, the compatibility of the cipher. Some outdated systems or devices may be incapable of supporting the most secure ciphers, making their use difficult or impossible. To maintain interoperability, a less secure cipher may be required in several instances. Three, the associated cost of the cipher. More secure ciphers may necessitate the use of specialized hardware or software, which can be expensive to build and maintain. In other circumstances, the extra expense may not be warranted, particularly for smaller firms or low-risk applications. Last, the usability of the cipher. More secure ciphers might be more difficult to set up and operate, necessitating greater training and experience. This can be difficult for some users, particularly in consumer-facing applications (Chen, 2017).

The Advanced Encryption Standard (AES) cipher uses hash functions to convert plaintext messages into fixed-length data known as message digests. These message digests can be used to validate data and detect unwanted changes or manipulation. AES can perform hash functions such as SHA-1, SHA-2, and MD5. However, the length and strength of the encryption keys are determined by bit levels in AES. The number of bits employed in the key calculation procedure is referred to as the bit level. AES accepts keys with lengths of 128, 192, and 256 bits. Longer key lengths provide greater security, but they can also be slower and require more resources (Gritzalis & Katsikas, 2017).

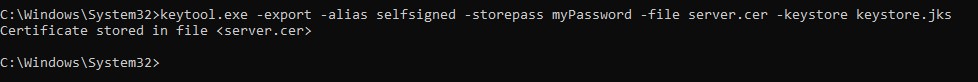
In symmetric encryption techniques, random integers are utilized to create encryption keys and initialization vectors. In key exchange protocols, they are also used to ensure that each side has a unique and unexpected key. The use of random numbers makes it more difficult for attackers to guess or predict the encryption key. But non-symmetric encryption methods utilize different keys for encryption and decryption, whereas symmetric encryption techniques use the same encryption key for both encryption and decryption. In general, symmetric encryption is faster and more efficient, whereas non-symmetric encryption gives more security. The encryption algorithm chosen is determined by the application's specific requirements (Martin, 2017).

Encryption methods have a long and evolving history, with new algorithms and techniques being created on a regular basis to address the changing needs of computer security. The earliest known encryption techniques date back to ancient Greece, when the Spartans encoded messages using a basic substitution cipher. Many more ciphers, including the famous Caesar cipher, were developed over the years. Moreover, Symmetric encryption methods such as DES and AES were created in the mid-twentieth century. These techniques are still frequently used today and use the same key for both encryption and decryption. Furthermore, non-symmetric encryption methods such as RSA were developed in the 1970s. These methods encrypt and decode using different keys and are especially useful for key exchange and digital signatures. Today, encryption techniques are constantly evolving and improving. To combat upcoming dangers such as quantum computing, new algorithms such as elliptic curve cryptography and post-quantum encryption are being developed. To conclude, since the first ciphers, encryption algorithms have come a long way, with new approaches and algorithms always being created to satisfy the changing needs of computer security. (Al-Riyami & Paterson, 2008).

## Certificate Generation

A black screen with white text

Description automatically generated



A computer screen with white text

Description automatically generated

**Figure 1. Shows screenshots of the self-generated certificate in Command Prompt.**

## Deploy Cipher

A screenshot of a computer

Description automatically generated

**Figure 2. Shows web page shot of checksum verification.**

## Secure Communications

A screenshot of a computer

Description automatically generated

**Figure 3. Shows web page shot of checksum verification.**

**HTTPS was not generated due to the certificate being “selfsigned”.**

## Secondary Testing

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

A screenshot of a computer error

Description automatically generated

**Figure 4. Shows successful execution of refactored code.**

[Dependency-Check-Report HTML Link](file:///C:\Users\rayya\Downloads\CS%20305%20Project%20Two%20Code%20Base\ssl-server_student\target\dependency-check-report.html)



## Functional Testing

A close up of text

Description automatically generated

**Figure 5. Shows successful execution of refactored code.**

A screenshot of a computer

Description automatically generated

**Figure 6. Shows successful maven build, clean, and installation of refactored code.**

## Summary

In this project for our client, Artemis Financial, we worked on many facets of security to include Input Validation, Cryptography, Code Quality, Code Error, and Secure API interactions. This was all done using methods like using algorithm ciphers, masking api interactions, handling errors using try-catch blocks, and rearranging and commenting in the code for clarity, and securing data transfers and web usage using https.

## Industry Standard Best Practices

Industry Standard best practices were used to mitigate the risk and solidify the current security of Artemis Financial’s software application. Primarily, encryption as well as compliance with standards and regulations were used. Organizations like Artemis Financial adhere to industry standards and regulatory requirements specific to their domain. These standards, such as the Payment Card Industry Data Security Standard (PCI DSS) or the General Data Protection Regulation (GDPR), provide guidelines and requirements for securing sensitive data and ensuring privacy. Moreover, developers adhered to secure coding practices to minimize vulnerabilities. This included using secure coding languages, validating, and sanitizing input data, implementing proper error handling, and avoiding common coding mistakes that can lead to security flaws. Furthermore, static, dependency, and functional testing was deployed to solidify Artemis software security. Overall, all of the above is vital for the wellbeing of a financial institution like Artemis to have better trust of its clients as well as stay compliant federally. Also, for Artemis, dealing with people funds nationally and internationally, it can be a vulnerable and a chaotic scene if a data leak is skipped or a misappropriation of funds is enabled due to not following industry-specific best practices.

**References**

National Institute of Standards and Technology (NIST). (2001). FIPS PUB 197: Advanced Encryption Standard (AES). Retrieved from <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf>

Gritzalis, S., & Katsikas, S. (2003). Security issues and solutions in electronic payment systems. Journal of Electronic Commerce Research, 4(3), 159-170. Retrieved from <https://www.researchgate.net/publication/228710121_Security_Issues_and_Solutions_in_Electronic_Payment_Systems>

G. Gritzalis and S. Katsikas, "Java Security Standard Algorithm Names," Oracle Corporation, 2017. [Online]. Retrieved from <https://docs.oracle.com/javase/9/docs/specs/security/standard-names.html#cipher-algorithm-names>

L. Chen, "Security of symmetric encryption algorithms," in Proceedings of the 2017 International Conference on Computer Science and Artificial Intelligence, Singapore, 2017, pp. 94-98

K. M. Martin, "Cryptography: An Overview," in Proceedings of the 2017 International Conference on Computer Science and Artificial Intelligence, Singapore, 2017, pp. 67-71

S. S. Al-Riyami and K. Paterson, "Quantum computation and public key cryptography," in Proceedings of the 2008 International Conference on Cryptology and Network Security, Singapore, 2008, pp. 198-207