****

# 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 6](#_Toc102040759)

[3. Deploy Cipher 6](#_Toc102040760)

[4. Secure Communications 6](#_Toc102040761)

[5. Secondary Testing 7](#_Toc102040762)

[6. Functional Testing 8](#_Toc102040763)

[7. Summary 9](#_Toc102040764)

[8. Industry Standard Best Practices 9](#_Toc102040765)

9. References…………………………………………………………………………………………………………………………………………**10**

## Document Revision History

| **Version** | **Date** | **Author** | **Comments** |
| --- | --- | --- | --- |
| **1.1** | **2/22/24** | **Rodney Adkins** | **Added Content** |

## Client



## Developer

Rodney Adkins

## Algorithm Cipher

After reviewing the security requirements of Artemis Financial we recommend that AES encryption is used for all archived data. Due to the sensitive nature of the data managed by Artemis, the data must be secured to a high level to prevent a security breach from damaging the customers of Artemis. If a threat actor were to obtain access to this information many types of attacks could manifest including but not limited to the following. The data could be held for ransom. The data could be used for financial gain through simple theft. Customer’s lives could be destroyed by identity theft. All of these scenarios could both harm the customers and the reputation of Artemis Financial. This provides the justification for an elevated level of encryption.

The Gramm-Leach-Bliley Act of 1999 makes the protection of financial information a requirement. (Liu) Although the act does not specify the level of encryption, it does require the data to be protected against anticipated threats. Our interpretation of this is that a method that is currently respected as an industry standard should be utilized. AES is widely used and has proven to be extremely effective and difficult to circumvent. An alternative is RSA encryption. Although it is more complex and can be even more difficult to crack, it creates much larger encrypted file sizes that would make it impractical to be used for large amounts of data. (Manico & Detlefsen) If this method were to be used the system and storage requirements would increase exponentially.

We recommend that the implementation includes all historical data and all newly created data. All data contained in the database will be encrypted with AES encryption. The AES encryption keys should also be encrypted and hosted on a separate application server that is protected via network security measures such as firewalls to isolate it from the internet. A plan to periodically change the encryption keys should be put into place. We also recommended that packet inspection be performed to detect any breach attempts. Any detected attack attempts should be investigated, and the performance of the protection should be evaluated.

In this case we need the encryption to be reversible so the data can be used by those who legitimately need access to it. This must be handled differently from data such as passwords that do not need to be retrieved in plain text format. In these cases, we can use hashing to scramble the data. When the encryption needs to be reversable AES and other methods utilize keys to be to decrypt the data back to a usable format. To provide greater security the bit level or length of the encryption keys can be increased which results in a more complex set of encrypted data. AES can use 128, 192, or 256 bit encryption making it both secure and flexible. As the bit level increases, so does the system requirements and storage size.

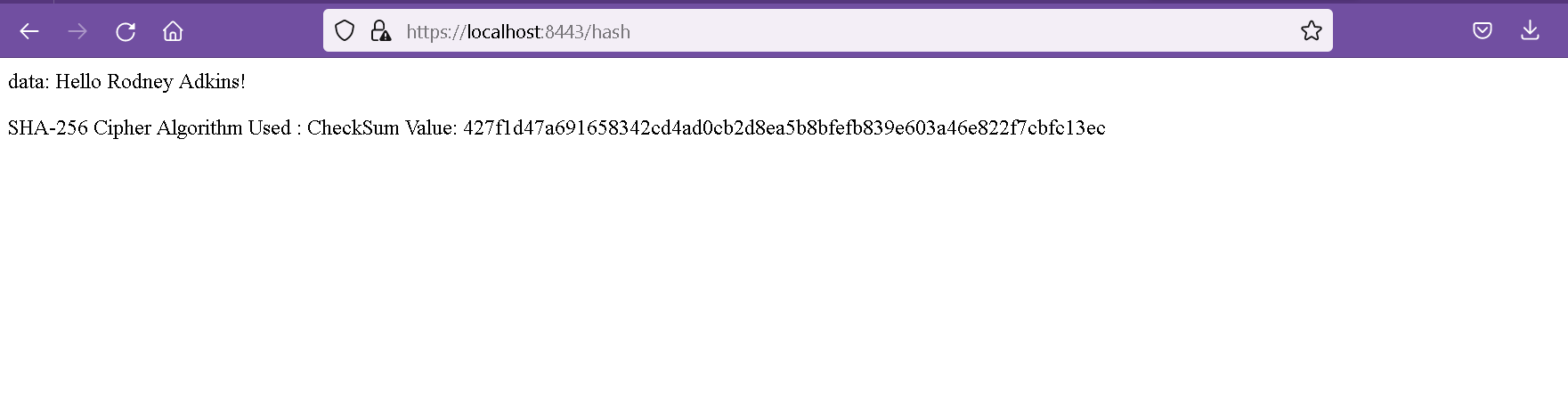
Cryptography is not a new concept and has been used dating all the way back to Julius Caesar in 60 BC. (2023) In the years leading up to and including the world wars the enigma machine played a key role. It was invented by a German engineer named Arthur Scherbius. The ability to crack the code played a big part in the allies’ success in World War 2. Although cryptography has come a long way in the modern world, the concepts remain the same. In the 1970’s IBM invented DES encryption which worked well for many years. It was replaced by AES around the turn of the 21st century. Although AES remains the most widely used encryption algorithm, work continues as more advanced algorithms such as RSA and ECC are being developed and used.

## Certificate Generation

A computer screen with white text

Description automatically generated

## Deploy Cipher



## Secure Communications

A screenshot of a computer

Description automatically generated

## Secondary Testing

A screenshot of a computer

Description automatically generated

I did not get any additional vulnerabilities in the dependency check. I updated the spring boot version and the Java JRE version. After doing this, I did not have any vulnerabilities in the dependency report.

A screenshot of a computer

Description automatically generated

## Functional Testing

## The manual review did not reveal any additional vulnerabilities. We could add some input validation to prevent an extremely long string from being received by the ServerController class. However, as the code is now, the user data is hard coded. If in future releases the data is from external input, this will need to be done.

A screenshot of a computer program

Description automatically generated

## Summary

The code has been refactored to include a SHA-256 cipher algorithm. The spring boot version has been set to the latest version and all dependency have been updated. At the time of running the latest dependency check, no vulnerabilities were detected. A self-signed certificate was added to enable HTTPS communication to the API.

A manual review did reveal a possible future vulnerability in the ServerController class where a long string could be received. The manual review focused on the areas of API, Cryptography, Code Quality, and Input Validation.

## Industry Standard Best Practices

We followed industry best practices by implemented a modern SHA-256 cipher algorithm and using a self-signed certificate to enable secure communication. When this code is put into production it would be a better practice to utilize the services of a certificate authority. This will provide an extra layer of security and improve the marketability of the application. The use of a well-recognized CA improves customer trust and maintainability.

All dependencies and libraries were brought up to the latest versions and are free from known vulneraries as of the date of this document. It would be in the company’s best interest to periodically update the application dependencies and release future revisions. This is the best way to stay one step ahead of any potential threat actors.

REFERENCES

Liu, K. (n.d.). *Guide to the Gramm–leach–bliley act*. Guide to the Gramm–Leach–Bliley Act. <https://iapp.org/resources/article/guide-to-the-gramm-leach-bliley-act/>

Manico, J., & Detlefsen, A. (n.d.). *Iron-clad Java*. O’Reilly Online Learning. <https://learning.oreilly.com/library/view/iron-clad-java/9780071835886/ch06.html#ch06lev1sec1>

*A brief history of encryption (and cryptography)*. Thales Group. (2023, February 1). https://www.thalesgroup.com/en/markets/digital-identity-and-security/magazine/brief-history-encryption