****

# 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** | **[Date]** | **[Your Name]** |  |

## 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

Heath Davis

## Algorithm Cipher

For this implementation, I selected **SHA-256** as the cryptographic hash algorithm for data integrity verification. SHA-256 provides 256-bit cryptographic security through a one-way hash function that is computationally infeasible to reverse, making it ideal for checksum verification in secure applications.

**Technical Implementation:**

* SHA-256 uses the Merkle-Damgård construction with 512-bit message blocks
* Produces a fixed 256-bit (32-byte) hash output regardless of input size
* Implemented using Java's MessageDigest.getInstance("SHA-256") for industry-standard compliance
* Output converted to hexadecimal representation for human-readable verification

**Security Justification:** SHA-256 is part of the SHA-2 family and is currently recommended by NIST for cryptographic applications. It provides strong collision resistance and pre-image resistance, making it suitable for data integrity verification in financial applications like Artemis Financial's platform. The algorithm's computational complexity ensures that generating fraudulent checksums is practically impossible with current technology.

## Certificate Generation

Insert a screenshot below of the CER file.

A screenshot of a computer

AI-generated content may be incorrect.

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screenshot of a computer program

AI-generated content may be incorrect.

## Secure Communications

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

A screenshot of a computer

AI-generated content may be incorrect.

## Secondary Testing

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

A screenshot of a computer program

AI-generated content may be incorrect.

## Functional Testing

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

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

## Summary

The Artemis Financial SSL server has been successfully refactored to implement comprehensive security measures addressing the client's requirements for data protection and secure communications.

**Security Layers Implemented:**

1. **Cryptographic Hashing**: SHA-256 algorithm for data integrity verification
2. **SSL/TLS Encryption**: HTTPS protocol with TLS 1.2/1.3 for secure communications
3. **Certificate-Based Authentication**: RSA 2048-bit certificates for server identity
4. **Dependency Security**: OWASP scanning for vulnerability management
5. **Secure Coding Practices**: Input validation and proper error handling

**Vulnerability Assessment Process:** Following the security development lifecycle, each component was analyzed for potential vulnerabilities:

* **Input Validation**: Checksum endpoint processes static data, eliminating injection risks
* **Cryptographic Implementation**: Industry-standard algorithms and secure random number generation
* **Communication Security**: Strong encryption protocols prevent eavesdropping and tampering
* **Dependency Management**: Regular vulnerability scanning ensures third-party component security

**Added Security Value:** The implementation provides multiple layers of security that protect Artemis Financial's client data and financial information. The cryptographic checksum ensures data integrity, while SSL/TLS encryption protects data in transit. These security measures demonstrate the company's commitment to protecting sensitive financial information and maintaining client trust.

## Industry Standard Best Practices

The implementation follows established industry standards and security best practices to ensure robust protection for the Artemis Financial application.

**Applied Security Standards:**

1. **NIST Cryptographic Standards**: SHA-256 compliance with FIPS 140-2 approved algorithms
2. **OWASP Security Guidelines**: Secure coding practices and vulnerability management
3. **TLS Best Practices**: Modern protocol versions and secure cipher suites
4. **Certificate Management**: Proper key generation and storage using industry-standard formats

**Secure Development Practices:**

* **Defense in Depth**: Multiple security layers provide redundant protection
* **Principle of Least Privilege**: Minimal required permissions and access controls
* **Input Validation**: Proper handling of user inputs and data processing
* **Error Handling**: Secure exception management without information disclosure
* **Regular Security Testing**: Automated vulnerability scanning and dependency checks

**Organizational Security Value:** Implementing these industry standards provides several benefits to Global Rain and Artemis Financial:

1. **Regulatory Compliance**: Meets financial industry security requirements (PCI DSS, SOX)
2. **Risk Mitigation**: Reduces exposure to data breaches and cyber attacks
3. **Client Trust**: Demonstrates commitment to protecting sensitive financial data
4. **Business Continuity**: Secure systems ensure reliable service delivery
5. **Competitive Advantage**: Security-first approach differentiates from competitors
6. **Cost Avoidance**: Prevents expensive security incidents and regulatory penalties

**Long-term Security Strategy:** The implemented security measures establish a foundation for ongoing security improvement. Regular dependency updates, certificate renewal procedures, and continuous security monitoring ensure the application maintains its security posture over time. This proactive approach to security supports Artemis Financial's mission to provide secure, reliable financial planning services to their clients.

**Future Enhancements:** Recommendations for continued security improvement include implementing certificate pinning, adding request rate limiting, integrating with a Web Application Firewall (WAF), and establishing automated security testing in the CI/CD pipeline.