

**Practices for Secure Software Report**

**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**

Maxwell Sheehan

* **Algorithm Cipher**

The AES(Advanced Encyption Standard) is the algorithim cipher I'm recomending for Artemis Financial. Its a symmetric block ciper which is deployed in multiple institutions as its increadibly secure for sensitve data. It stores keys up to 256 bits which is increadibly resilient to known attacks, including brute force, as the longer key strings enhance security and increase the time of brute force. AES is an encryption cipher, combining it with SHA-256 ensures data integrity as a hash function. The symmetric key system encrypts and decypts data, to increase the secuirty of the application by removing plain text that can easily be found and exploited.

* **Certificate Generation**

Insert a screenshot below of the CER file.

  


* **Deploy Cipher**

Insert a screenshot below of the checksum verification.



* **Secure Communications**

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

* [
* 
* **Secondary Testing**

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

  


* **Functional Testing**

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



* **Summary**

For security improvenments we added SHA -256 hashing at the hasn endpoint, we also implemented HTTPS via PKCS12 keystore to secure our communcations. Specifically input validation has no sensitve data exposed, data in transit is encrypted via HTTPS which is industry standard. We've implemented industy standard and regulatyory cryptography via strong hashing algorithims and ciphers. We also tested and refactored our code to ensure no vulnerabilites were introduced.

* **Industry Standard Best Practices**

We ensured our code starts cleanly via Spring Boots tests, this allows the code to be more modularized and scaled further in the future if the company grows. We used the industry standard and regulatory recommended algroithims and ciphers for strong standardized cryptography. This is seen with SHA -256 and using HTTPS instead of the unsecure HTTP. By meeting compliance with NIST, and OWASP we also impvoed the companys position and general public trust.