****

# 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]** | **Alexander Egelston** |  |

## 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

Alexander

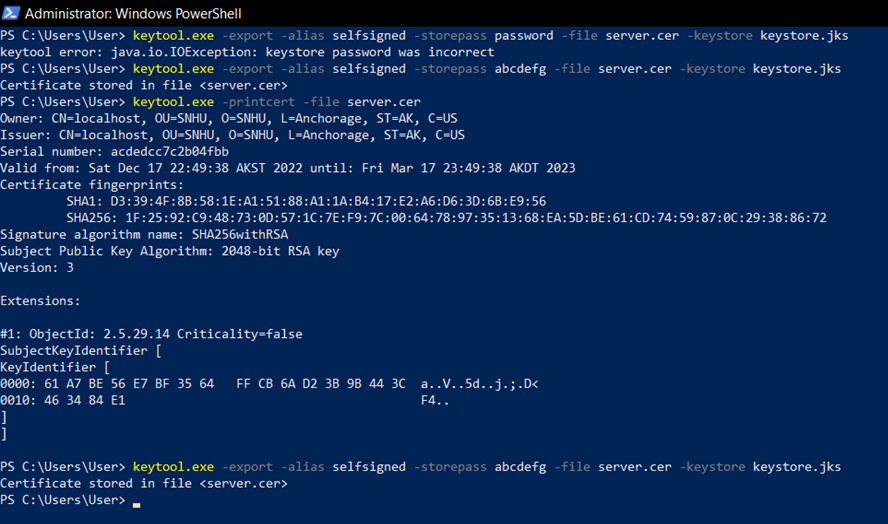
## Algorithm Cipher

**Recommend an appropriate encryption algorithm cipher to deploy, given the security vulnerabilities, and justify your reasoning. Review the scenario and the supporting materials to support your recommendation. In your practices for secure software report, be sure to address the following:**

1. **Provide a brief, high-level overview of the encryption algorithm cipher.**Based on the project and its goals and vulnerabilities, AES (Advanced Encryption Standard) appears to be the best choice for an algorithm cipher. AES has an immense security factor at 128, 192, and 256 bits and has never been cracked. AES is a symmetric block cipher meaning it encrypts chunks of data instead of individual characters. This method encrypts the same data differently every time it goes through the encryption process.
2. **Discuss the hash functions and bit levels of the cipher.**The hash function of AES is .The different bit levels of AES family of algorithms are 128, 192, and 256 bits. The higher the bit level the more secure the encryption is, but the more bandwidth is required for supporting and processing the encryption and decryption of data.
3. **Explain the use of random numbers, symmetric versus non-symmetric keys, and so on.**Random Number Generators are needed for cryptographic key generation. Symmetric encryption uses the same key for encryption and decryption. Asymmetric encryption uses a public key for encryption and a separate private key for decryption.
4. **Describe the history and current state of encryption algorithms.**In the late 1990s the government determined that DES (Data Encryption Standard, the encryption algorithm of choice at the time) was slowly becoming obsolete as modern devices became more advanced and increased in computing power/speed. After a 3-year long international competition, NIST (National Institute of Standards) announced the replacement for DES as Rijndael, an algorithm submitted by two cryptographers from Belgium. Since October 2000 AES (Advanced Encryption Standard, or Rijndael) has been the encryption algorithm of choice around the world as it would take modern computers an estimated years to crack the algorithms' 256 bit key with brute force methods. Even Tianhe-2 (MilkyWay-2), one of the fastest supercomputers in the world, would take millions of years to crack 256-bit AES encryption.

## Certificate Generation

Insert a screenshot below of the CER file.



## Deploy Cipher

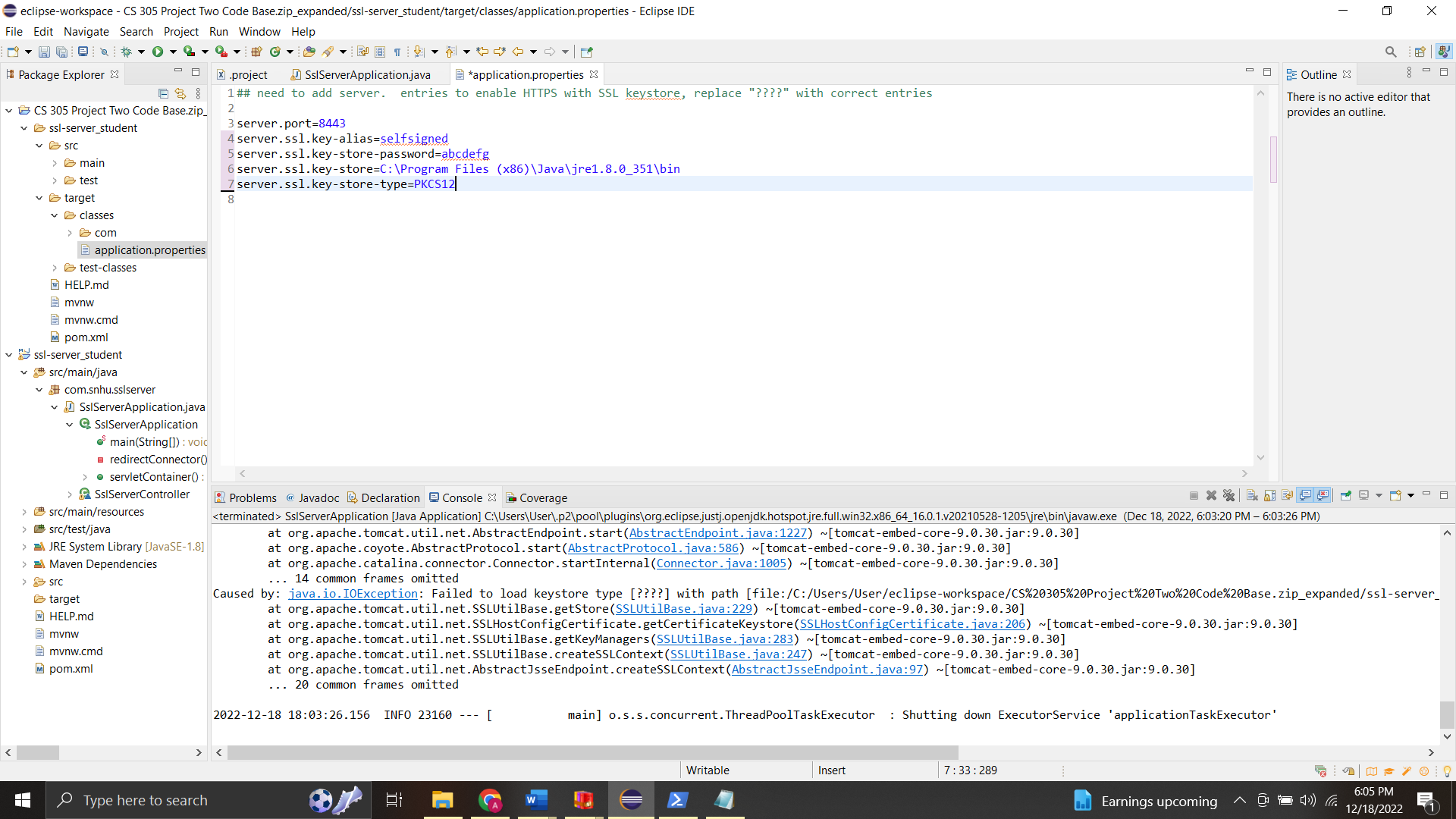
Insert a screenshot below of the checksum verification.

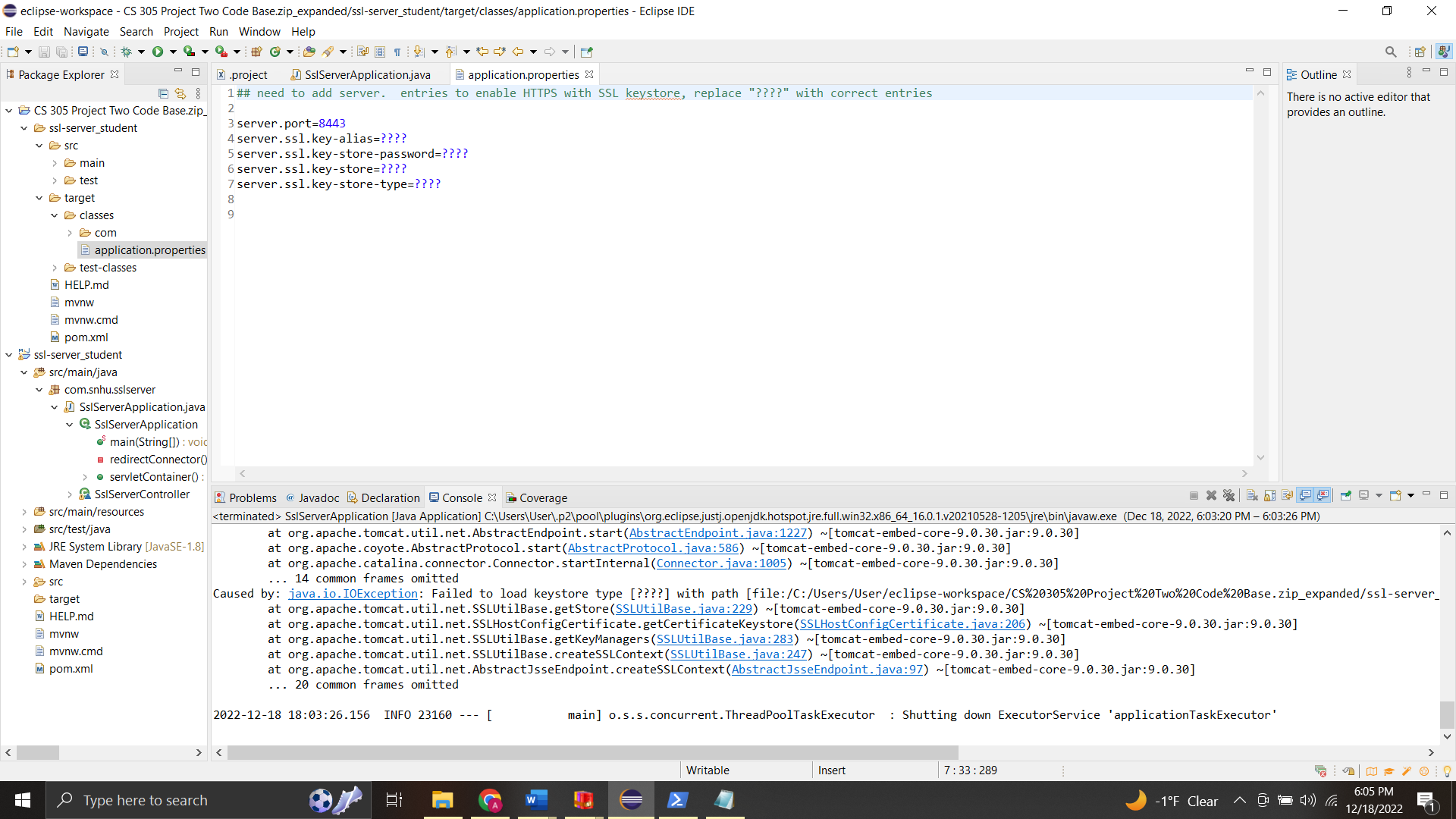
[Insert screenshots here.]

## Secure Communications

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

Attempting to implement an HTTPS localhost I encountered an interesting and very frustrating error. I don’t know what I am doing wrong here but I must be missing something pretty important because every time I save the application.properties file or try to run the code in Eclipse it reverts back to its original state. The first picture is before I save or run the program with all “????” replaced with information from the certificate generation of step 2 and the second picture in this section is after.

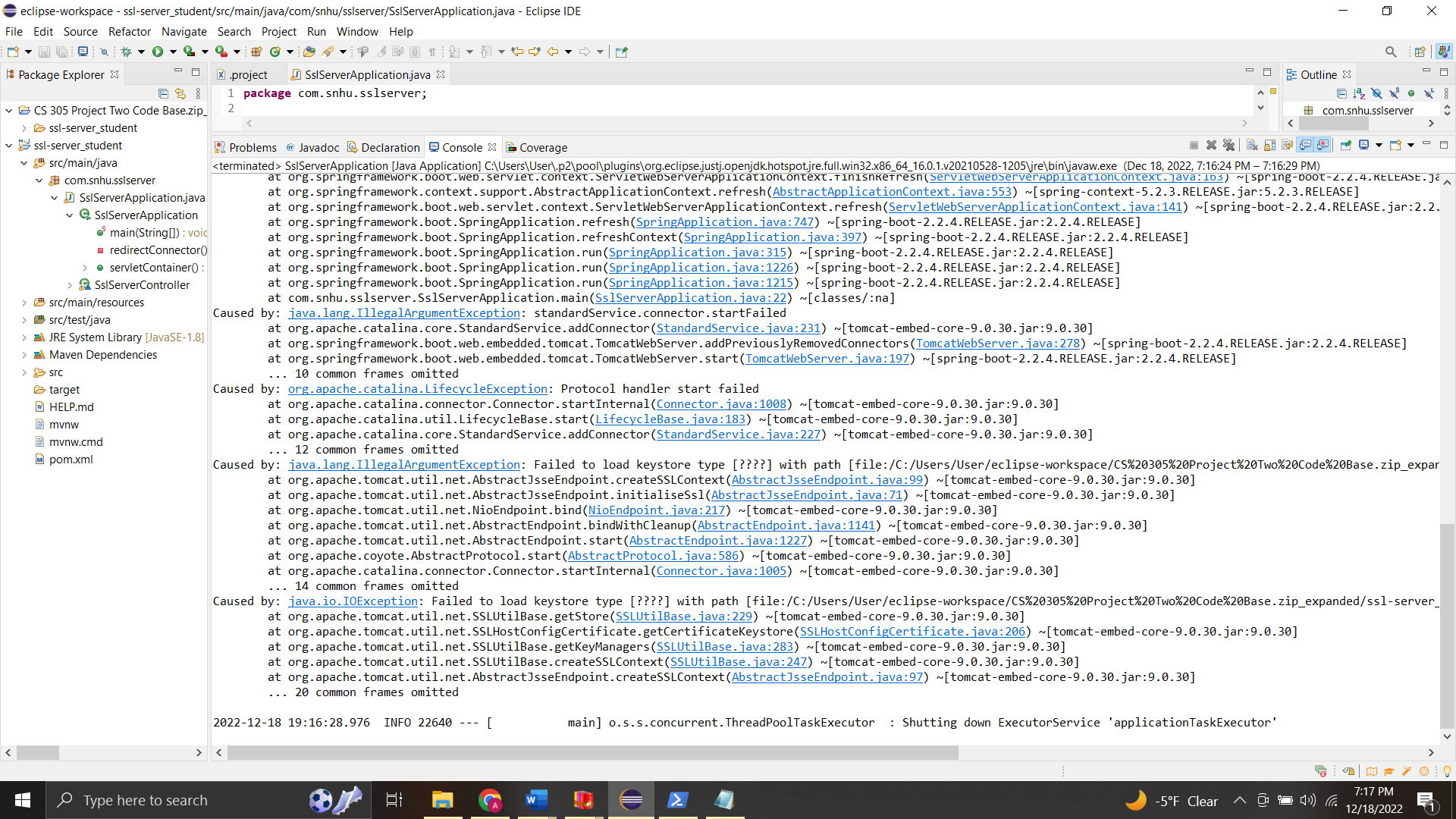




## Secondary Testing

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

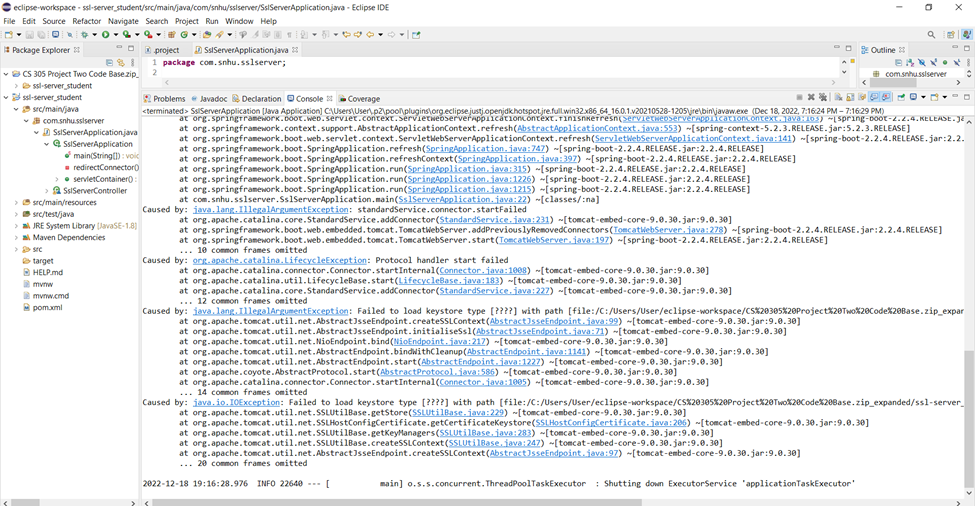
And similar to the step regarding the HTTPS server, I am unable to get the project code to execute at all and run a dependency report due to exceptions with the tomcat server not saving in the application.properties file and instead reverting back to its original state.



## Functional Testing

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

As with step 5 and 4 the application.properties file is reverting back to its original state and clearing any text I change in it (replacing the question marks). Due to this the code is not executing whenever I try to execute the code with changes added.



When I try to execute the code without the changes added to the application.properties file I get an error preventing the build that states “Variable references non-existent resource : ${workspace\_loc:/rest-service}”.

## Summary

Discuss how the code has been refactored and how it complies with security testing protocols. In the summary of your practices for secure software report, be sure to address the following:

1. Refer to the Vulnerability Assessment Process Flow Diagram. Highlight the areas of security that you addressed by refactoring the code.  
   I was unable to refactor the code because it failed to build regardless of what I tried to fix it.
2. Discuss your process for adding layers of security to the software application.  
   I attempted to implement the AES-256 security algorithm for encryption of data in the project.

## Industry Standard Best Practices

Explain how you applied industry standard best practices for secure codingto mitigate against known security vulnerabilities.Be sure to address the following:

1. Explain how you used industry standard best practices to maintain the software application’s current security.  
   I don’t feel industry standard best practices were maintained in the software application because it wasn’t successfully compiled although I did try to write classes and objects as modularly as I could for easy readability.
2. Explain the value of applying industry standard best practices for secure coding to the company’s overall wellbeing.

The value of applying industry standard best practices for secure coding lies in being able to understand what may be a vulnerability in your code as well as easy means of altering it in the future should you find a critical vulnerability that you want to mitigate.

Sources

Loo, A. (2022, November 11). *Hash function*. Corporate Finance Institute. Retrieved December 18, 2022, from https://corporatefinanceinstitute.com/resources/cryptocurrency/hash-function/