GLOBALRAIN

# Practices for Secure Software Report

# Table of Contents

**Document Revision History**

| Version | Date | Author | Comments |
|---|---|---|---|
| 1.0 | 2/14/2023 | Vincent Snow | |

**Client**

ARTEMIS
FINANCIAL

**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.

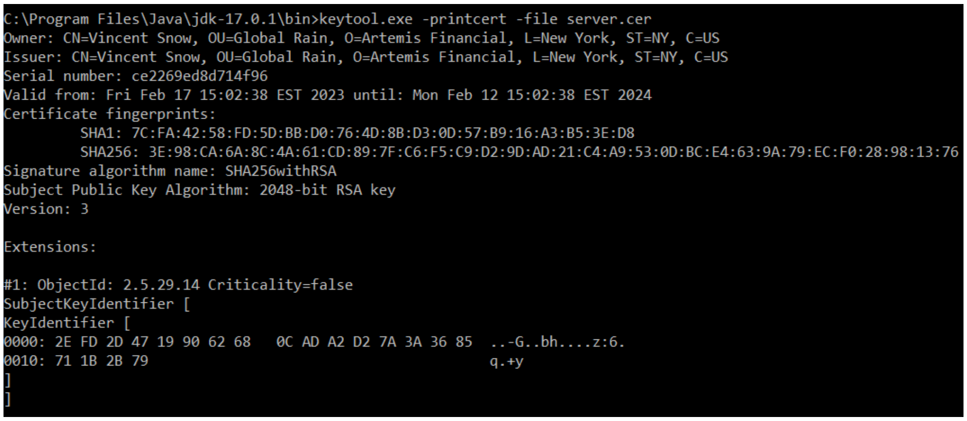
**Developer**
Vincent Snow

## 1. Algorithm Cipher

I recommend that a 256-bit AES cipher be used. This algorithm meets US legal requirements for sensitive data encryption, such as financial account information. The bit depth of 256 is the length of the key used to decrypt data. 256 bits is recommended because computers are unlikely to be able to guess these keys in our lifetimes, making it a long-term high security option. The hash function is the mathematical function used to change information into an encrypted message, and the SHA-256 function is currently standard for sensitive data.
This algorithm is for standard non-symmetrical encryption in which a private key is used to decrypt the message that is different from the public key used to encrypt it. Symmetrical encryption is not safe in most situations because it uses a shared key to encrypt and decrypt. It would be difficult to keep that secret key known only by the client and server. Random number generating algorithms are used in this cipher to generate unique keys that are difficult to guess. Cryptographic algorithms follow the same concepts as encrypted messages used in ancient civilizations, but on a much more complex level. As computers evolved to solve problems and process commands faster, the need for higher bit depths for keys grew. From the 1970s to the 1990s, the DES (Data encryption standard) used a 56-bit cipher. In 2001, it was replaced by AES as the standard encryption due to how easily it could be broken through by brute-force hacking. AES uses at least 128 bits, which is considered secure for short-term data encryption. (GeeksforGeeks, 2022)

## 2. Certificate Generation

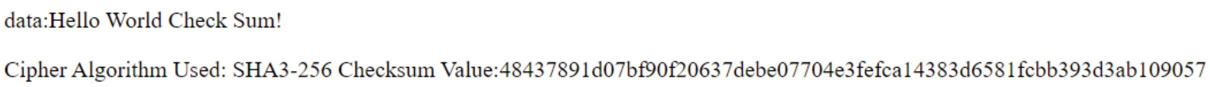
Insert a screenshot below of the CER file.



```
C:\Program Files\Java\jdk-17.0.1\bin>keytool.exe -printcert -file server.cer
Owner: CN=Vincent Snow, OU=Global Rain, O=Artemis Financial, L=New York, ST=NY, C=US
Issuer: CN=Vincent Snow, OU=Global Rain, O=Artemis Financial, L=New York, ST=NY, C=US
Serial number: ce2269ed8d714f96
Valid from: Fri Feb 17 15:02:38 EST 2023 until: Mon Feb 12 15:02:38 EST 2024
Certificate fingerprints:
         SHA1: 7C:FA:42:58:FD:5D:BB:D0:76:4D:8B:D3:0D:57:B9:16:A3:B5:3E:D8
         SHA256: 3E:98:CA:6A:8C:4A:61:CD:89:7F:C6:F5:C9:D2:9D:AD:21:C4:A9:53:0D:BC:E4:63:9A:79:EC:F0:28:98:13:76
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 2E FD 2D 47 19 90 62 68   0C AD A2 D2 7A 3A 36 85  ..-G..bh....z:6.
0010: 71 1B 2B 79                                        q.+y
]
]
```

## 3. Deploy Cipher

Insert a screenshot below of the checksum verification.

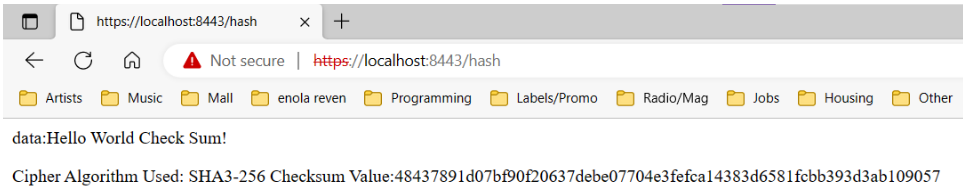


data:Hello World Check Sum!

Cipher Algorithm Used: SHA3-256 Checksum Value:48437891d07bf90f20637debe07704e3fefca14383d6581fcbb393d3ab109057

## 4. Secure Communications

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



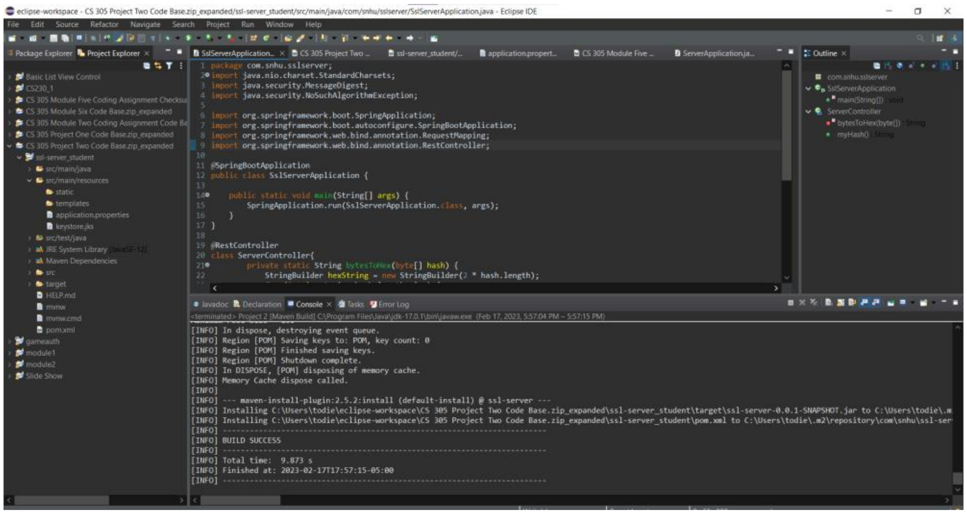
data:Hello World Check Sum!

Cipher Algorithm Used: SHA3-256 Checksum Value:48437891d07bf90f20637debe07704e3fefca14383d6581fcbb393d3ab109057

By default, Edge, Chrome and Firefox will not trust self-signed certificates for security reasons. There are different ways to set up a browser on the client side to accept it depending on your browser, version and OS. (Gu, 2022)

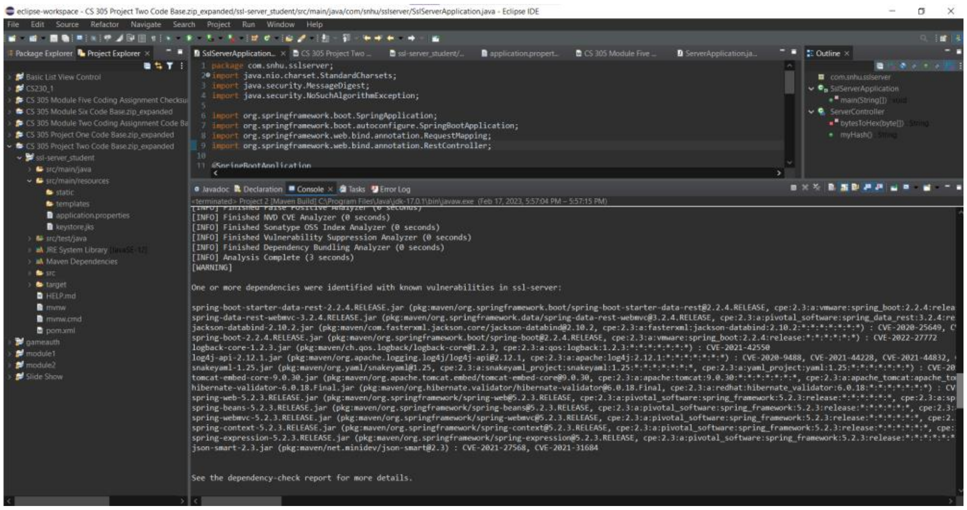
## 5. Secondary Testing

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



```java
package com.snhu.sslserver;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@SpringBootApplication
public class SslServerApplication {

    public static void main(String[] args) {
        SpringApplication.run(SslServerApplication.class, args);
    }
}

@RestController
class ServerController{
        private static String bytesToHex(byte[] hash) {
            StringBuilder hexString = new StringBuilder(2 * hash.length);
```

```
<terminated> Project 2 [Maven Build] C:\Program Files\Java\jdk-17.0.1\bin\javaw.exe (Feb 17, 2023, 5:57:04 PM – 5:57:15 PM)
[INFO] In dispose, destroying event queue.
[INFO] Region [POM] Saving keys to: POM, key count: 0
[INFO] Region [POM] Finished saving keys.
[INFO] Region [POM] Shutdown complete.
[INFO] In DISPOSE, [POM] disposing of memory cache.
[INFO] Memory Cache dispose called.
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ ssl-server ---
[INFO] Installing C:\Users\todie\eclipse-workspace\CS 305 Project Two Code Base.zip_expanded\ssl-server_student\target\ssl-server-0.0.1-SNAPSHOT.jar to C:\Users\todie\.m
[INFO] Installing C:\Users\todie\eclipse-workspace\CS 305 Project Two Code Base.zip_expanded\ssl-server_student\pom.xml to C:\Users\todie\.m2\repository\com\snhu\ssl-ser
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  9.873 s
[INFO] Finished at: 2023-02-17T17:57:15-05:00
[INFO] ------------------------------------------------------------------------
```



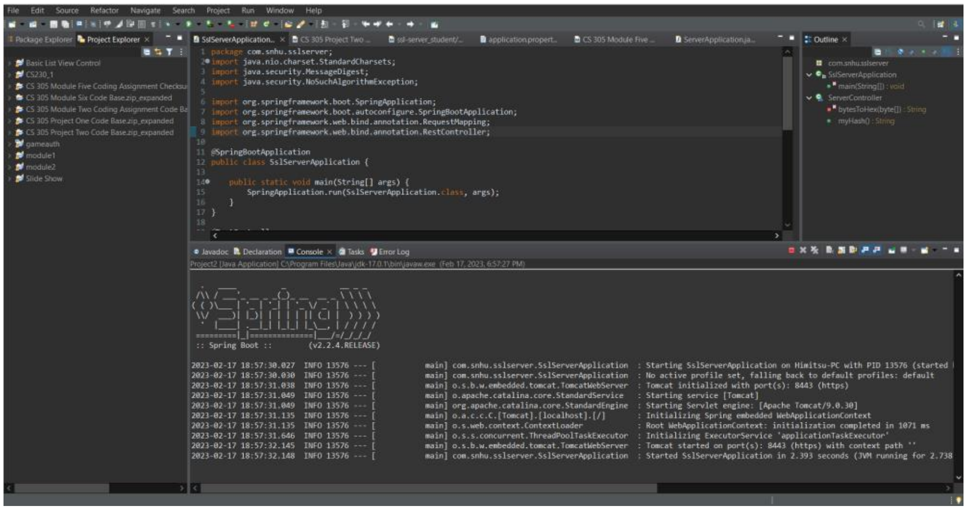
```java
package com.snhu.sslserver;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@SpringBootApplication
```

<terminated> Project 2 [Maven Build] C:\Program Files\Java\jdk-17.0.1\bin\javaw.exe (Feb 17, 2023, 5:57:04 PM – 5:57:15 PM)

```
[INFO] Finished False Positive Analyzer (0 seconds)
[INFO] Finished NVD CVE Analyzer (0 seconds)
[INFO] Finished Sonatype OSS Index Analyzer (0 seconds)
[INFO] Finished Vulnerability Suppression Analyzer (0 seconds)
[INFO] Finished Dependency Bundling Analyzer (0 seconds)
[INFO] Analysis Complete (3 seconds)
[WARNING]

One or more dependencies were identified with known vulnerabilities in ssl-server:

spring-boot-starter-data-rest-2.2.4.RELEASE.jar (pkg:maven/org.springframework.boot/spring-boot-starter-data-rest@2.2.4.RELEASE, cpe:2.3:a:vmware:spring_boot:2.2.4:relea
spring-data-rest-webmvc-3.2.4.RELEASE.jar (pkg:maven/org.springframework.data/spring-data-rest-webmvc@3.2.4.RELEASE, cpe:2.3:a:pivotal_software:spring_data_rest:3.2.4:re
jackson-databind-2.10.2.jar (pkg:maven/com.fasterxml.jackson.core/jackson-databind@2.10.2, cpe:2.3:a:fasterxml:jackson-databind:2.10.2:*:*:*:*:*:*:*) : CVE-2020-25649, C
spring-boot-2.2.4.RELEASE.jar (pkg:maven/org.springframework.boot/spring-boot@2.2.4.RELEASE, cpe:2.3:a:vmware:spring_boot:2.2.4:release:*:*:*:*:*:*) : CVE-2022-27772
logback-core-1.2.3.jar (pkg:maven/ch.qos.logback/logback-core@1.2.3, cpe:2.3:a:qos:logback:1.2.3:*:*:*:*:*:*:*) : CVE-2021-42550
log4j-api-2.12.1.jar (pkg:maven/org.apache.logging.log4j/log4j-api@2.12.1, cpe:2.3:a:apache:log4j:2.12.1:*:*:*:*:*:*:*) : CVE-2020-9488, CVE-2021-44228, CVE-2021-44832,
snakeyaml-1.25.jar (pkg:maven/org.yaml/snakeyaml@1.25, cpe:2.3:a:snakeyaml_project:snakeyaml:1.25:*:*:*:*:*:*:*, cpe:2.3:a:yaml_project:yaml:1.25:*:*:*:*:*:*:*) : CVE-20
tomcat-embed-core-9.0.30.jar (pkg:maven/org.apache.tomcat.embed/tomcat-embed-core@9.0.30, cpe:2.3:a:apache:tomcat:9.0.30:*:*:*:*:*:*:*, cpe:2.3:a:apache_tomcat:apache_to
hibernate-validator-6.0.18.Final.jar (pkg:maven/org.hibernate.validator/hibernate-validator@6.0.18.Final, cpe:2.3:a:redhat:hibernate_validator:6.0.18:*:*:*:*:*:*:*) : CV
spring-web-5.2.3.RELEASE.jar (pkg:maven/org.springframework/spring-web@5.2.3.RELEASE, cpe:2.3:a:pivotal_software:spring_framework:5.2.3:release:*:*:*:*:*, cpe:2.3:a:sp
spring-beans-5.2.3.RELEASE.jar (pkg:maven/org.springframework/spring-beans@5.2.3.RELEASE, cpe:2.3:a:pivotal_software:spring_framework:5.2.3:release:*:*:*:*:*, cpe:2.3:
spring-webmvc-5.2.3.RELEASE.jar (pkg:maven/org.springframework/spring-webmvc@5.2.3.RELEASE, cpe:2.3:a:pivotal_software:spring_framework:5.2.3:release:*:*:*:*:*, cpe:2.
spring-context-5.2.3.RELEASE.jar (pkg:maven/org.springframework/spring-context@5.2.3.RELEASE, cpe:2.3:a:pivotal_software:spring_framework:5.2.3:release:*:*:*:*:*, cpe:
spring-expression-5.2.3.RELEASE.jar (pkg:maven/org.springframework/spring-expression@5.2.3.RELEASE, cpe:2.3:a:pivotal_software:spring_framework:5.2.3:release:*:*:*:*:*
json-smart-2.3.jar (pkg:maven/net.minidev/json-smart@2.3) : CVE-2021-27568, CVE-2021-31684


See the dependency-check report for more details.
```

## 6. Functional Testing

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



```java
package com.snhu.sslserver;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@SpringBootApplication
public class SslServerApplication {

    public static void main(String[] args) {
        SpringApplication.run(SslServerApplication.class, args);
    }
}
```

Project2 [Java Application] C:\Program Files\Java\jdk-17.0.1\bin\javaw.exe (Feb 17, 2023, 6:57:27 PM)

```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::        (v2.2.4.RELEASE)

2023-02-17 18:57:30.027  INFO 13576 --- [           main] com.snhu.sslserver.SslServerApplication  : Starting SslServerApplication on Himitsu-PC with PID 13576 (started
2023-02-17 18:57:30.030  INFO 13576 --- [           main] com.snhu.sslserver.SslServerApplication  : No active profile set, falling back to default profiles: default
2023-02-17 18:57:31.038  INFO 13576 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 8443 (https)
2023-02-17 18:57:31.049  INFO 13576 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2023-02-17 18:57:31.049  INFO 13576 --- [           main] org.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tomcat/9.0.30]
2023-02-17 18:57:31.135  INFO 13576 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2023-02-17 18:57:31.135  INFO 13576 --- [           main] o.s.web.context.ContextLoader            : Root WebApplicationContext: initialization completed in 1071 ms
2023-02-17 18:57:31.646  INFO 13576 --- [           main] o.s.s.concurrent.ThreadPoolTaskExecutor  : Initializing ExecutorService 'applicationTaskExecutor'
2023-02-17 18:57:32.145  INFO 13576 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port(s): 8443 (https) with context path ''
2023-02-17 18:57:32.148  INFO 13576 --- [           main] com.snhu.sslserver.SslServerApplication  : Started SslServerApplication in 2.393 seconds (JVM running for 2.738
```

## 7. Summary

The code attached has been refactored to address the security areas of cryptography, secure API interaction, and client-server interaction. HTTPS protocol routes were added that respond to a specific request. A certificate is used to verify that the view sent from the server is authentic when the client receives it, and checksums verify that the content is unchanged from the original. The code was tested with a self-signed certificate and printed checksum to verify that the processes are working correctly. This will protect the application from cross-site scripting attacks that can attempt to steal customer information by convincing them to enter it into a false website or returning deceptive text or elements on a webpage.

## 8. Industry Standard Best Practices

A highly secure, industry-standard cipher algorithm with HTTPS protocol was used in the application. Not only does this comply with legal requirements, but it also keeps sensitive data very secure and encrypts data in a way that should be safe for long-term storage. Java’s versions and dependencies were tested for vulnerabilities so that the application can be kept up to date from known weaknesses, errors, or successful attack patterns.

# References

GeeksforGeeks. (2022, October 6). *History of Cryptography*. https://www.geeksforgeeks.org/history-of-cryptography/

Gu, D. (2022, January 5). *Working With Self-Signed Certificates in Chrome (Walkthrough Edition)*. Medium. https://dgu2000.medium.com/working-with-self-signed-certificates-in-chrome-walkthrough-edition-a238486e6858