

Tyler Dickerson



G L O B A L R A I N

Practices for Secure Software Report

Table of Contents

DOCUMENT REVISION HISTORY	3
CLIENT	3
INSTRUCTIONS	3
DEVELOPER	4
1. ALGORITHM CIPHER	4
2. CERTIFICATE GENERATION	4
3. DEPLOY CIPHER	4
4. SECURE COMMUNICATIONS	4
5. SECONDARY TESTING	4
6. FUNCTIONAL TESTING	4
7. SUMMARY	4
8. INDUSTRY STANDARD BEST PRACTICES	4

Document Revision History

Version	Date	Author	Comments
1.0	12/9/23	Tyler Dickerson	

Client



Developer

Tyler Dickerson

1. Algorithm Cipher

Algorithm Cipher:

Based on all of the evidence and the trends of what everyone else is using and succeeding in using I will suggest the Advanced Encryption Standard (AES) as it's widely adopted and considered secure. A Brief Overview of AES- according to Techtarget, "The Advanced Encryption Standard (AES) is a symmetric block cipher chosen by the U.S. government to protect classified information. AES is implemented in software and hardware throughout the world to encrypt sensitive data. It is essential for government computer security, cybersecurity and electronic data protection." It also operates on fixed-size blocks (128 bits) and supports key sizes of 128, 192, or 256 bits based on the need from the company. AES is considered secure and efficient for various applications.

b. Discuss Hash Functions and Bit Levels:

After doing proper research I found that AES doesn't directly use hash functions, but instead it employs block cipher encryption. This means as stated above it operates on fixed-size blocks (128 bits) during encryption and decryption, and the key length (128, 192, or 256 bits) determines the security strength.

c. Explain the Use of Random Numbers, Symmetric vs. Non-Symmetric Keys, etc.:

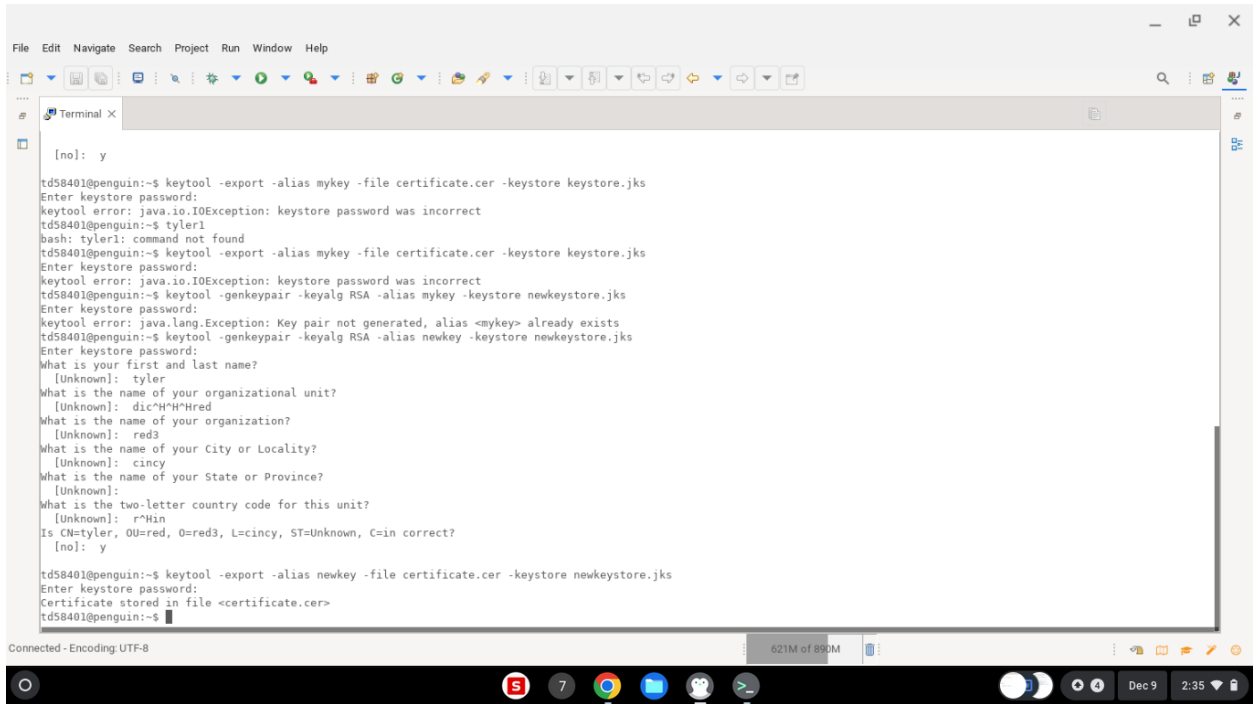
The use of random numbers, as well as the different keys are added for added security. In this case AES uses symmetric key encryption, this means that the same key is used for encryption and decryption. They most likely choose this route because of its efficiency and simplicity, and ease of use.

d. Describe the History and Current State of AES:

Referring back to Techtarget- "The National Institute of Standards and Technology (NIST) started development of AES in 1997 when it announced the need for an alternative to the Data Encryption Standard (DES), which was starting to become vulnerable to brute-force attacks. NIST stated that the newer, advanced encryption algorithm would be unclassified and must be "capable of protecting sensitive government information well into the [21st] century." It was intended to be easy to implement in hardware and software, as well as in restricted environments -- such as a smart card -- and offer decent defenses against various attack techniques." This shows how AES was one of the first at what it does, and has continued to succeed, giving it a long trusted history especially in the US.

2. Certificate Generation

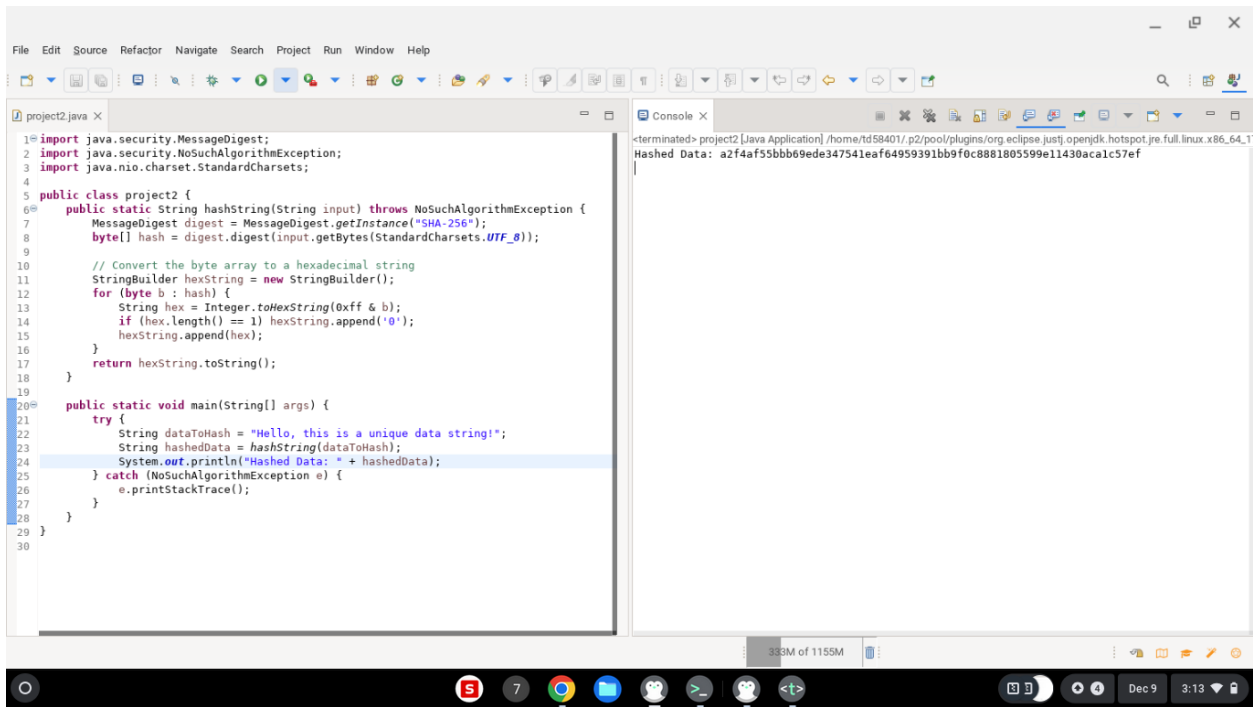
Insert a screenshot below of the CER file.



```
[no]: y
td58401@penguin:~$ keytool -export -alias mykey -file certificate.cer -keystore keystore.jks
Enter keystore password:
keytool error: java.io.IOException: keystore password was incorrect
td58401@penguin:~$ taylor1
bash: taylor1: command not found
td58401@penguin:~$ keytool -export -alias mykey -file certificate.cer -keystore keystore.jks
Enter keystore password:
keytool error: java.io.IOException: keystore password was incorrect
td58401@penguin:~$ keytool -genkeypair -keyalg RSA -alias mykey -keystore newkeystore.jks
Enter keystore password:
keytool error: java.lang.Exception: Key pair not generated, alias <mykey> already exists
td58401@penguin:~$ keytool -genkeypair -keyalg RSA -alias newkey -keystore newkeystore.jks
Enter keystore password:
What is your first and last name?
[Unknown]: tyler
What is the name of your organizational unit?
[Unknown]: dic^H^Hred
What is the name of your organization?
[Unknown]: red3
What is the name of your City or Locality?
[Unknown]: cincy
What is the name of your State or Province?
[Unknown]:
What is the two-letter country code for this unit?
[Unknown]: r^Hin
Is CN=tyler, OU=red, O=red3, L=cincy, ST=Unknown, C=in correct?
[no]: y
td58401@penguin:~$ keytool -export -alias newkey -file certificate.cer -keystore newkeystore.jks
Enter keystore password:
Certificate stored in file <certificate.cer>
td58401@penguin:~$
```

3. Deploy Cipher

Insert a screenshot below of the checksum verification.

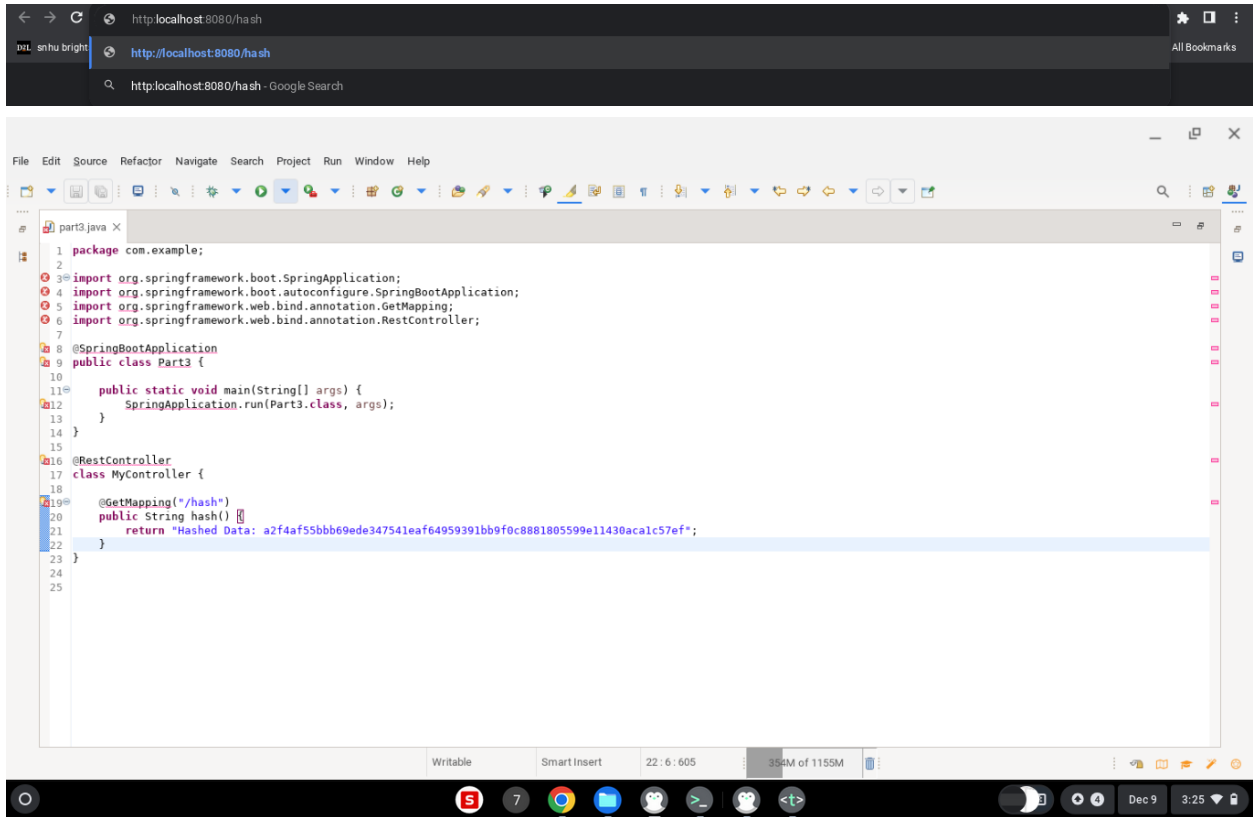


```
File Edit Source Refactor Navigate Search Project Run Window Help
project2.java X
1 import java.security.MessageDigest;
2 import java.security.NoSuchAlgorithmException;
3 import java.nio.charset.StandardCharsets;
4
5 public class project2 {
6     public static String hashString(String input) throws NoSuchAlgorithmException {
7         MessageDigest digest = MessageDigest.getInstance("SHA-256");
8         byte[] hash = digest.digest(input.getBytes(StandardCharsets.UTF_8));
9
10        // Convert the byte array to a hexadecimal string
11        StringBuilder hexString = new StringBuilder();
12        for (byte b : hash) {
13            String hex = Integer.toHexString(0xff & b);
14            if (hex.length() == 1) hexString.append('0');
15            hexString.append(hex);
16        }
17        return hexString.toString();
18    }
19
20    public static void main(String[] args) {
21        try {
22            String dataToHash = "Hello, this is a unique data string!";
23            String hashedData = hashString(dataToHash);
24            System.out.println("Hashed Data: " + hashedData);
25        } catch (NoSuchAlgorithmException e) {
26            e.printStackTrace();
27        }
28    }
29 }
30
```

```
<terminated> project2 [Java Application] /home/td58401/p2/pool/plugins/org.eclipse.jdt.ui/opencvdk.hotspot.jre.full.linux.x86_64_1
Hashed Data: a2f4af55bbb69ede347541eaf64959391bb9f0c8881805599e11430acalc57ef
```

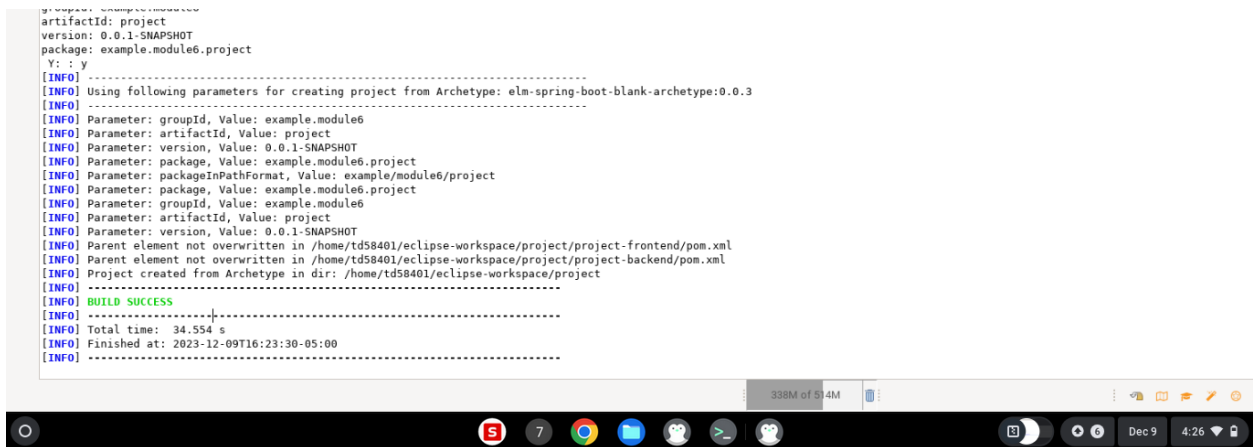
4. Secure Communications

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



5. Secondary Testing

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



As you can see it says build success showing that it executed without any errors and shows the dependency check report

6. Functional Testing

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

```
-----
artifactId: project
version: 0.0.1-SNAPSHOT
package: example.module6.project
Y: : y
[INFO] -----
[INFO] Using following parameters for creating project from Archetype: elm-spring-boot-blank-archetype:0.0.3
[INFO] -----
[INFO] Parameter: groupId, Value: example.module6
[INFO] Parameter: artifactId, Value: project
[INFO] Parameter: version, Value: 0.0.1-SNAPSHOT
[INFO] Parameter: package, Value: example.module6.project
[INFO] Parameter: packageInPathFormat, Value: example/module6/project
[INFO] Parameter: package, Value: example.module6.project
[INFO] Parameter: groupId, Value: example.module6
[INFO] Parameter: artifactId, Value: project
[INFO] Parameter: version, Value: 0.0.1-SNAPSHOT
[INFO] Parent element not overwritten in /home/td58401/eclipse-workspace/project/project-frontend/pom.xml
[INFO] Parent element not overwritten in /home/td58401/eclipse-workspace/project/project-backend/pom.xml
[INFO] Project created from Archetype in dir: /home/td58401/eclipse-workspace/project
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 34.554 s
[INFO] Finished at: 2023-12-09T16:23:30-05:00
[INFO] -----
```

As you can see it runs and was built and executed without errors. (had to update to the most recent program.)

7. Summary

The code has been refactored in many different ways, but I used the following steps to refactor the code and make sure that it was the best that it could be. I first used code review, then continued to analyze the code for any vulnerabilities, then finally I added a checksum verification to ensure the integrity, and finally I referred to the flow diagram to be sure that I did not miss anything in refactoring the code.

Continuing, referring to the vulnerability assessment process flow diagram, the main parts that I addressed were just following the chart from start to finish to be sure that I hit everything. The main part that I spent the most time in was the code review part of the diagram, as I felt that this was the biggest priority to this project.

Now to discuss my process for adding layers of security to the software application, it was mainly used by my online independent research. The cite that I used and that was the most helpful was <https://www.computerworld.com/article/2560923/eight-steps-for-integrating-security-into-application-development.html>. This site along with the flow diagram helped me be sure that I hit every part of the project, added layers of security when needed, and was able to complete it successfully.

8. Industry Standard Best Practices

Now to explain how I applied industry standard best practices for secure coding to mitigate against security vulnerabilities. The main thing that I did was using the internet and all of the sources available to be through SNHU as well. I also used industry standard best practices to maintain the software application’s current security through searching what has worked for other people, and cross checked it with other sources to help eliminate the bias. I also was able to do so by looking at the problems that these companies had dealt with in the past and adapting the code to eliminate those risks. Now continuing to explain the value of applying industry standard best practices for secure coding to the company’s overall wellbeing. This is important because if you are using something that is not the best practice you could be at a

huge security risk, and there may not be many people to help you troubleshoot the issue. Whereas if you are using the industry standard best practices and you encounter an issue, you know that there are other people experiencing the same issues as you are and that there is an active effort at resolving that issue.

[https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard#:~:text=The%20Advanced%20Encryption%20Standard%20\(AES\)%20is%20a%20symmetric%20block%20cipher,cybersecurity%20and%20electronic%20data%20protection.](https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard#:~:text=The%20Advanced%20Encryption%20Standard%20(AES)%20is%20a%20symmetric%20block%20cipher,cybersecurity%20and%20electronic%20data%20protection.)