****

# 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** | **April 25, 2024** | **Larry Shayne McDonald** | **Completion of prokject** |

## 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

Larry Shayne McDonald

## Algorithm Cipher

Encryption dates back thousands of years ago, it became a way of passing orders and conversations in secret and in plain sight knowing that without a key. The information will be secure from others. In WW2, the United States used languages of certain tribes from North AMerica. These people were called wind talkers and could speak in their native tongue and the German Army could not decipher the language as it was never written down and it was passed on verbally throughout the tribe. That did not mean they did not have their own way of using a code to send messages. They invented the enigma machine. This machine allowed someone to type a message that looked like nonsense and send it. Once received, it can decrypt the data and place it back to its original message.

I would choose the Advance Encryption Standard (AES), since this provides the most usual form of security since the internet is involved. It is also secure for known security risks. AES is trusted by larger companies; it does have its downsides. Meaning that all parties involved have a key they use to decrypt the information, this is also known as Symmetric Encryption. Even though the information is encrypted, the key can be stolen and used to decrypt the data. For example, if the key is used when someone is using a public Wi-Fi, the key can be intercepted from your device. This could also be done if you use an unsafe Wi-Fi network.

Since AES is used as the security backbone for many companies and organizations it is regulated by the government. If the regulations were not there it could cause some cost cutting companies to think they are saving money on security, but in turn they are making it easier for hackers to steal information and even money. If the company and their clients are hacked it could cause the local economy to lose millions if not billions depending on the types of business. It is imperative for the companies and customers to have their data secure.

The algorithm cipher is used to help secure a file when it is sent. This could be documents, pictures, etc. The way this helps secure the information is that the recipient will also have the algorithm so they can unscramble the data received and see what the file is. If someone can hack and steal the data, they are not able to read it as they do not have the algorithm key.

After researching the several types, the advanced encryption standard is the best option. This is due to how impervious it is against hackers. However, the one drawback is that it is susceptible to a brute force attack. It could take a while but in the end it can break. But that is as of right now. We have seen how devices have changed over the years along with programming as well. In the future AES could not be the best one to use.

## Certificate Generation

Insert a screenshot below of the CER file.

A computer screen shot of a computer program

Description automatically generated

## Deploy Cipher

Insert a screenshot below of the checksum verification.

A screenshot of a computer

Description automatically generated

## Secure Communications

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

A screenshot of a computer program

Description automatically generated

## Secondary Testing

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

**class** ServerController {

**private** **static** **final** **char**[] HEX\_ARRAY = "0123456789ABCDEF".toCharArray();

**private** String getHash(String input) {

**try** {

MessageDigest messageDigest = MessageDigest.getInstance("SHA-256");

**byte**[] messageDigestMD5 = messageDigest.digest();

**return** bytesToHex(messageDigestMD5);

} **catch** (NoSuchAlgorithmException e) {

e.printStackTrace();

}

**return** input;

}

**public** **static** String bytesToHex(**byte**[] bytes) {

**char**[] hexChars = **new** **char**[bytes.length \* 2];

**for** (**int** j = 0; j < bytes.length; j++) {

**int** v = bytes[j] & 0xFF;

hexChars[j \* 2] = HEX\_ARRAY[v >>> 4];

hexChars[j \* 2 + 1] = HEX\_ARRAY[v & 0x0F];

}

**return** **new** String(hexChars);

}

@RequestMapping("/hash")

**public** String myHash() {

String data = "Ethans Check Sum!!";

String hash = getHash(data);

**return** "<p>data: " + data + "</p><p> Name of Cipher Used: SHA-256 Value: " + hash;

}

}

A screenshot of a computer program

Description automatically generated

## Functional Testing

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

Original Dependency Check

A screenshot of a computer

Description automatically generated

Updated Dependency Check

A screenshot of a computer

Description automatically generated

## Summary

Determine your client’s needs and potential threats and attacks associated with their application and software security requirements. Consider the following regarding how companies protect against external threats based on the scenario information:

• How can secure communications provide value to the company?

• Does the Company produce international transactions?

• To have secure communications are there any restrictions that could be regulated by a government?

• In the immediate future, what external threats may be out there?

• What are the “modernization” requirements such as the role of open-source libraries and evolving web application technologies?

Providing secure communications is important due to how the data is protected when moving from server and clients. Some of the ways we can ensure that is by using encryption and authentication. It is a top priority to ensure that all sensitive information is secure. If it is not thought of as a top priority, it could lead to customers being open to threats of identity theft or fraud. This would not only tarnish the customer but also our company as well. Global Rain can be used to track international transactions, as they specialize in custom software design and development for clients around the world. Since there is no government restriction about how secure communications should be. However financial institutions are regulated such as Global Rain must adhere to.

One of these regulations is the Gramm-Leach-Bliley Act, which requires the company to provide their customers with how their information-sharing practices are implemented. Financial institutions are a major target of hackers for their sensitive information that could leave the customer vulnerable to attacks. If a hacker can interfere with the secure communication, they can retrieve sensitive information like social security numbers, account credentials, and even their home addresses are just a few examples. A financial institution “Reputation” is one way for them to procure more customers and keep existing ones. However, if you have a negative reputation for lacking proper security this will lead to customers taking their business elsewhere. One way to be up to date on security is to have the system run updates to help combat the hackers in the future.]

## Industry Standard Best Practices

First up is the input validation which is something that is mandatory. To be able to reach more customers and provide them with a way to conduct their business through the app, instead of being in person. Due to the software requiring an input function we must ensure that we validate the inputs received to prevent attacks that could jeopardize the system. Having a Secure API interaction because Artemis Financial has a RESTful web interface. It is necessary to ensure adequate cryptography is put in place to encrypt the sensitive data. Secure client / server can use an API access, we must ensure that proper certificate is implemented to confirm security for both sides. Being able to correct code errors while creating the program and trying to make this as secure as possible will be more cost effective.