CS 305 Software Security  
Static Testing

Sergio Mateos

Southern New Hampshire University

As Senior Software Developer, I’m responsible that the code would be secure. I will assess potential vulnerabilities in the code and create a migration plan for existing vulnerabilities. The Vulnerabilities Assessment Process Flow will help us to find out any type of vulnerability existing in our code.

## Areas of Security

* Input Validation
* APIs
* Cryptography
* Client/Server
* Code Error
* Code Quality
* Encapsulation

## Areas of Security Justification

* Input Validation:
  + Input Validation is an important area to ensure is secure. Malicious users can inject any attack into our system by entering invalid inputs. The validation of the users can secure our system.
* API:
  + Since we need access from outside to our system, that’s why the implementation of RESTful APIs is a proper way to know where unknown outside sources access our system. REST APIs will need to be inspected.
* Cryptography:
  + All data must be always protected this will help to keep a safe connection with our servers. All types of interactions must be encrypted to keep confidence. APIs are going to be used on HTTP which requests information from customers, and this information must be safeguarded by encrypting.
* Client/Server:
  + The connection and interaction between the servers and customers must be close to prevent any bridge which allows malicious users to hack into the data or servers. Ensuring that the information is past safety is prioritized by the customers’ information.
* Code Error:
  + The code must run properly to prevent any crashes and prevent any backdoors for malicious users. Also, the defenses must always run properly.
* Code Quality:
  + The quality of the code will always be written at a high standard, using the proper functions and methods to achieve high-quality code.
* Encapsulation:
  + The data structure will be secured since the code should not allow access to different functions of the class will be accessed by a different class.

## Code Review Summary

Based on the code provided “CS 305 Module Two Written Assignment Code Base” I discover a couple of problems that can be approached by malicious users and hack the system.

1. The Spring Framework is an old version. The old version of Spring Framework allows us to be vulnerable to any attack since they might not patch the old access to malicious users.
2. Text, letter

   Description automatically generatedThe class “GreetingController.java” produce an error while analyzing the code. The code passes a string value as plain text in our template. Also, the code does not limit the number of characters in our string. Allowing these types of problems to pass make the system vulnerable to an attack from malicious users.

## Mitigation Plan

The Mitigation Plan requires some steps to make sure that our system would be secure and efficient.

* The software must be updated to the latest version to prevent any vulnerability access. Updating to version 3.30 (or the latest version) will maintain our system secure.
* On the “GreetingController.java,” we must limit the size of the string being pasted into the template. This can reduce the system from malicious attacks.
* In addition, verification of the users’ input can prevent any malicious data to be placed in the data.