|  |
| --- |
| CS 305 Software Security |
| 2-1 Written Assignment: Code Review and Mitigation Plan |
| Emily Wood |

|  |
| --- |
| emily.wood7@snhu.edu  3-12-2023 |

# CS 305 Module Two Written Assignment

## Areas of Security

* We are implementing an expressive command input function for a complex web application. We are currently using Version 2.6.5 of the spring-data-rest-webmvc in the Spring framework, and we want to use the Spring Expression Language to accomplish the task.
* Input Validation
* APIs
* Cryptography
* Client/Server
* Code Quality
* Encapsulation

## Areas of Security Justification

* First, we need to ensure input validation is performed. This is to prevent malfunctions of various downstream components (OWASP Cheat Sheet Series, n.d.).
* Since we are using a web application, secure API interactions are essential. Since APIs are very commonly used and they enable access to sensitive software functions and data, they are becoming a primary target for attackers (Bright Security, 2022).
* Cryptography should be used for data to ensure attackers cannot make use of it (Bright Security, 2022).
* Client/Server is necessary because we are using an API. Correct certificates are necessary for any data transfers.
* Code Quality is essential because basic secure coding practices and patterns impact how the program behaves. Static testing can also be used to identify areas where code quality is lacking.
* Encapsulation is important to ensure that boundaries are maintained, and attackers cannot gain unauthorized access.

## Code Review Summary

* In GreetingController lines 24 throuh 34, the input is not validated.  
  Text

  Description automatically generated
* Also, in lines 24 through 34 if this data being passed is protected there should be some sort of encryption.
* In the entire program, there are no try/catch functions for secure error handling. Errors given in full to the user may reveal information about the code that can be exploited by attackers.
* The code itself needs to be cleaned up for spacing and indentations. Lines 25-26 and lines 36-38.  
  Text

  Description automatically generated

## Mitigation Plan

* Ensure all versions are current.
* Include code to validate any input that is passed through, ie. “name” and “id”.
* Include Try/Catch functions with basic output as to not reveal the nature of any exceptions.
* Remove excess spacing and fix indentation to create cleaner more readable code.

**References:**

Input validation cheat sheet¶. Input Validation - OWASP Cheat Sheet Series. (n.d.). Retrieved March 10, 2023, from https://cheatsheetseries.owasp.org/cheatsheets/Input\_Validation\_Cheat\_Sheet.html

API security: The Complete Guide to Threats, methods & tools. Bright Security. (2022, April 21). Retrieved March 10, 2023, from https://brightsec.com/blog/api-security/#rest-api-vs-soap-security