**8-2 Journal: Portfolio Reflection**

Chris Bridges

Southern New Hampshire University

CS 405: Secure Coding

Professor Mill

19 October 2025

This course has taught me valuable information about secure coding and how to implement it. Security should be implemented throughout the development cycle and not something added at the end. Security is about creating products that are more robust and reliable, in addition to being more resistant to attacks. Integrating security from the beginning saves time and money by creating the code once and not having to redo entire sections at the end to meet security guidelines.

Secure code is more reliable and more resilient to both human error and attacks. The downside of creating secure code is the additional time and resources spent throughout the process, but this is small in comparison to redoing a code base to meet security after the fact. Unsecure code creates the risk of attacks, which can incur a financial cost to the customer, a loss for the company in terms of doing the work again, and the loss of confidence for all parties involved.

Zero-trust policies are a great way to maintain a more secure environment. Multifactor authentication helps secure data from all access points, whether inside the network or outside. Physical security can be circumvented and your data stolen. Zero trust policies require employees use authentication to log in, even inside the company building, which improves security internally and externally. This is just another layer in the defense in depth strategy, making data more secure.

I will use secure coding techniques like input validation and range validation to ensure my code is as strong as possible. Integrating secure coding policies with the code I create ensures what I create is durable and resilient. Secure code and DevSecOps policies will ensure I maintain industry standards and meet the expectations of my employers.