Reflection

Giovanni V Oddo

SNHU

CS 405: Secure Coding

Prof. Ramsey Kraya

April 27, 2025

Reflection

After completing this course, my mindset for security has changed since completing other courses that touch on security topics. Starting with creating a secure coding standard, this is much like writing an outline for a paper or sctructuring main topics to discuss. Adopting a coding standard helps a team understand common vulnerabilities with the project they are to be working on and acts as a guide on how to implement code across the system to avoid those vulnerabilities. This also aids with maintaining the code once the project is done, with the idea being that all developers abide by the given standards and create consistent and secure functions. Another change to my mindset is not waiting until the end of development to address security issues. Since time and money are the major constraints to any project, the more time saved can equate to more money saved. Another positive outcome to address security early is that more vulnerablilities can be mitigated when more minds and experience are working to identify common issues with a specific project.

Evaluation and assessment of risk and cost benenfit of mitigation not only shows the team how severe and probable vulnerabilities are, it also communicates to stakeholders the additional cost should a security or data breach happen if projects or features are pushed to prodution too hastily. Stakeholders may not always understand technical speak, but everyone understands increase to costs or the benefit of saving money. Zero Trust seems to be the standard with any application that allows users to have any sort of input or permissions within a system. A standard that sticks with me from this course is validate all inputs, this is consistent with the Zero Trust mindset.

Implementation and recommendation of security policies need to be both generic and specific to a project. When I say specific to the project, I mean information such as which platform is hosting the application which has specific vulnerabilities such as cross site scripting or an API leaking credentials on a cloud service.