Portfolio Reflection

8-2 Journal

CS 405

Secure Coding

Southern New Hampshire University

Sawyer Kent

August 20, 2023

A couple key themes strongly stick with me as I think back on this course and the vast amount of information we were exposed to, and they have influenced how I see software development and security.

Adoption of a Secure Coding Standard: One of the core principles we've struggled with is the necessity of integrating security right from the start, as opposed to retrofitting it as an afterthought. In "Secure Coding: A Practical Guide," Murray (2020) succinctly said that it is not just about writing code that works but also about creating code that is resilient to threats. Building a solid foundation for a house is equivalent to the idea of not putting security off until the last minute. An oversight made early on in the coding process can later turn into a serious vulnerability that is more difficult and expensive to fix.

Evaluation and assessment of risk: Any digital undertaking inevitably involves some level of risk. However, what separates a reactionary response from a strategic one is our attitude to risk assessment, comprehension of the potential consequences, and the cost-benefit analysis of mitigation. This sentiment is emphasized in Dunham's 2020 discourse on information security regulations, highlighting the importance of taking a proactive approach. We not only secure our systems but do it in a resource-efficient way by evaluating potential threats and weighing them against the cost, both financial and reputational, of mitigation measures.

Zero Trust: The concept of "Zero Trust" has arisen as a beacon, guiding our security processes, in the ever-evolving world of cyber dangers. The time when foreign threats were the only worry is long gone. The saying "trust but verify" has evolved into "never trust, always verify" in today's world as a result of the expansion of the threat matrix. Every entity, whether a computer or a human, is being scrutinized, whether it's vetting a software component or validating user access, emphasizing the significance of ongoing validation.

Implementation and Recommendations of Security Policies: Creating a security policy is similar to drawing a map for a journey because it identifies the route, potential obstacles, and solutions for avoiding them. But as we just mentioned, these policies frequently have gaps. They may be caused by inadequate coverage, stale threat information, or even a lack of regular reviews. As security experts, it is our responsibility to not only put these rules into practice but to keep improving them so that they are responsive to the always changing threat landscape.

In conclusion, this course has been an enlightening experience that has instilled a deep respect for the complex dance between code and security in me. It is clear that learning languages and platforms isn't enough to practice secure coding; one must also embrace a certain mentality that places equal importance on innovation and functionality as it does on protecting assets, data, and reputation.