Nicholas Davis

CS405

Professor Mimi Tam

08/24/2025

**Portfolio Reflection**

Adoption of a secure coding standard and not leaving security to the end are fundamental aspects of secure coding, as such they should be implemented throughout the entire software development lifecycle. By adopting a secure coding standard, we can set and use best practices for all developers to follow and use tools that aid in secure systems. This also enables developers to understand what is expected and how they should implement features like input validation. With the concept of not leaving security to the end we can ensure a quality foundation of security that should reduce cost if implemented correctly, as opposed to waiting until something has happened to implement effective security, or clean up the issue. Additionally, this will boost trust with users and stakeholders, as they are able to see the development team takes security seriously.

Through evaluation and assessment of risk and cost benefits of mitigation we can eliminate many problems before they occur. This involves identifying risks and analyzing what harm they could potentially incur, as well as the likelihood they will occur. This will enable prioritization of risk and concentrate workload on the most critical threats. If security is implemented early enough and is effective there will be great initial cost, however the costs would be greater if this was not the case, as cleanup of vulnerabilities and implementation of new security would be much more costly.

The “no one is safe” (zero trust) concept is a model that makes security a forefront issue by ensuring no device or user is trusted by default. This means that there is always verification of users and devices, even if they are already using the network. With the traditional approach to security, it is assumed that users or devices that are already using the network must be authorized. With the Zero Trust model, it requires verification at every step, and even if the user or device is inside the network. Though it may be redundant for users, this is important because it gives the systems greater security by allowing fewer successful attacks, enabling less cleanup on the back end.

Implementation of security policies should be assessed in the pre-production phase with the goal of preventing attacks, creating a greater defense in depth approach, and low upfront cost compared to later implementation. The utilization of tools needs to be identified so that any given job is given the correct tools. Sources and repositories need to be understood to provide adequate security. Additionally, if vulnerabilities do arise in post-production, the policy should be clear on what actions need to be taken according to the issue. The majority of this can be done with the mindset of three coding principles; Adopting a Secure Coding Standard, Keep it Simple, and Architect and Design for Security Policies.

**References:**

SEI CERT C++ Coding Standard. <https://wiki.sei.cmu.edu/confluence/pages/viewpage.action?pageId=88046682>

Information Security Policies: Why They Are Important To Your Organization. <https://linfordco.com/blog/information-security-policies/>

Understanding the Zero Trust Security Model to Safeguard Digital Infrastructure(2024). https://www.digitalguardian.com/blog/understanding-zero-trust-security-model-safeguard-digital-infrastructure