Britian Holcomb

CS 405

8-2 Journal

**Adoption of a Secure Coding Standard and Not Leaving Security to the End**

A secure coding standard is essential to creating resilient software from the ground up. Following standards like the SEI CERT C++ Coding Guidelines ensures that developers proactively avoid common vulnerabilities such as buffer overflows and improper input validation. The principle of "don’t leave security to the end" reinforces that security must be embedded throughout the software development lifecycle, not as an afterthought. For instance, incorporating secure coding practices during requirements and design phases helps reduce the overall cost and complexity of mitigation later.

**Evaluation and Assessment of Risk and Cost-Benefit of Mitigation**Risk assessment helps prioritize which vulnerabilities to address based on their potential impact and likelihood. For example, mitigating a high-risk vulnerability in an authentication system may justify a greater investment than fixing a low-risk aesthetic bug. This trade-off ensures resources are effectively allocated and supports informed decision-making. Understanding these evaluations also enhances compliance with regulatory standards and minimizes reputational and operational damage in case of a breach.

**Zero Trust**  
The Zero Trust model challenges the traditional "trust but verify" approach by assuming that threats can exist both outside and inside the network. In a Zero Trust architecture, no user or system is automatically trusted, and access is granted based on verification, least privilege, and continuous authentication. Zero Trust is especially vital in today’s hybrid and cloud-based environments, where traditional network perimeters are no longer sufficient.

**Implementation and Recommendations of Security Policies**  
Security policies serve as the foundation for enforcing consistent security practices across an organization. An effective policy outlines roles, acceptable use, incident response, and compliance requirements. Recommendations for strengthening policies include adopting a DevSecOps culture to integrate security into development workflows and updating policies regularly to reflect emerging threats and technologies. These policies help create a security-conscious culture and ensure alignment between technical controls and organizational goals.