In this reflection, I’ll be discussing the topics of adopting a secure coding standard, not leaving security to the end, evaluation and assessment of risk and cost benefit of mitigation, zero trust, and implementation and recommendations of security policies.

Adopting a secure, renowned coding standard like [SEI CERT C++ Coding Standard](https://wiki.sei.cmu.edu/confluence/pages/viewpage.action?pageId=88046682) lays out the blueprints for developing a framework that minimizes the occurrence of vulnerabilities. Adopting a coding standard early in development allows issues to be addressed in real-time, reduces remediation costs, and builds security with a proactive approach. Another benefit to adopting a coding standard is to have clear instructions and feedback for incorporating methods/functions, and how to respond and address vulnerabilities. In a company setting, this allows all employees to operate with the same understanding.

Throughout this class, one of the most consistent reminders was to incorporate security into the fundamentals and foundation of development rather than incorporating security as an afterthought. The statement, “don’t leave security to the end,” means to approach security as a proactive practice rather than a retroactive practice. Applying this statement and best practice to the development of software minimizes potential threats and prevents weak spots in code from being forgotten.

Evaluating and assessing risk and cost benefit of mitigation refers to balancing risk against cost to make security decisions. Having no protection against risk may make accessing information, data, or a system easy, but it also leaves sensitive data and systems vulnerable. Having too much protection against risk may create a wall of security measures that makes the software feel inaccessible. Thus, to balance the risk and cost, it’s important to look at whether implementing certain security measures are “worth it.” Performing a cost-benefit analysis helps identify which risks are more relevant or severe than others, which highlights which areas need to be prioritized with implementing security measures, and which areas can remain uncluttered from excessive security practices.

Zero trust refers to a cybersecurity model that requires verification of all internal and external access attempts, eliminating trust to prevent breaches (*Understanding the Zero Trust Security Model to Safeguard Digital Infrastructure | Digital Guardian*, n.d.). This model examines security through the statement of, “no one is safe.” Thus, every internal and external system must undergo multiple verification methods to confirm they’re trusted. From a developer perspective, this means integrating different methods of verification like multi-factor authentication (MFA), adhering to the principle of least privilege, and validating input for each process of the system. From a user perspective, this means working through various authentication and verification methods to access the information you desire. On the surface, this seems optimal as it maximizes the security for each process of the system. However, this can be easily overdone and can become an annoyance that dissuades users from interacting with the software and system.

Looking at implementing and recommending security policies is dependent on the project details and the enforceability of the policy. A security policy’s purpose is to establish expectations for secure development practices. This includes addressing layers of security, how to respond to an incident, and compliance with established coding standards. To successfully implement a security policy, proper training and support need to be present. Discussing the policy with the applicable audience is a necessity. Without a discussion, there will be different thought processes or interpretations that may lead to an inaccurate and unsuccessful implementation.

Works Cited:

*SEI CERT C++ Coding Standard—SEI CERT C++ Coding Standard—Confluence*. (n.d.). Retrieved August 20, 2025, from<https://wiki.sei.cmu.edu/confluence/pages/viewpage.action?pageId=88046682>

*Understanding the Zero Trust Security Model to Safeguard Digital Infrastructure | Digital Guardian*. (n.d.). Retrieved August 19, 2025, from<https://www.digitalguardian.com/blog/understanding-zero-trust-security-model-safeguard-digital-infrastructure>