Kendal Guizado

Portfolio Reflection

**Adoption of a Secure Coding Standard**

Using secure coding standards from the start of development is essential to prevent vulnerabilities. Standards like OWASP or SANS provide guidelines to help developers write code that guards against common threats. Addressing security only at the end can lead to costly fixes and missed vulnerabilities. By adopting these practices early, you ensure that security is built into the code from the beginning, which is more efficient and effective.

**Evaluation and Assessment of Risk and Cost Benefit of Mitigation**

Assessing risks involves identifying potential threats and their impact on your organization. When deciding how to mitigate these risks, you need to weigh the cost of security measures against the potential damage from breaches. This means prioritizing risks based on their potential impact and the resources required for mitigation. A balanced approach ensures that high-priority risks are managed effectively while keeping costs under control.

**Zero Trust**

The Zero Trust model operates on the principle that no user or device should be trusted by default, regardless of their location. It requires continuous verification of every access request and strictly enforces the principle of least privilege. By constantly validating users and devices and not relying solely on perimeter defenses, Zero Trust provides a stronger security posture against both internal and external threats.

**Implementation and Recommendations of Security Policies**

Effective security policies are crucial for guiding and enforcing security practices within an organization. These policies should address key areas such as data protection, user access, and incident response. They need to be clearly defined, regularly updated, and consistently enforced. Regular employee training, periodic audits, and automated compliance tools help ensure that security policies are followed and that the organization remains protected against evolving threats.