Michelle Lewis

CS405-R4887

**Module 8 Journal: Reflection**

This class has been an interesting dive into cybersecurity. I have learned the importance of creating coding standards that serve as a guide for developers on how to implement secure coding practices. Coding standards are set based on the requirements of a project and the goals of an organization, ensuring all developers follow the same set of security guidelines instead of leaving it up to each developer to determine which coding standards, if any, they will use (Seacord, 2013). Also, it fosters a culture of prioritizing security so that developers implement these policies in all the work that they do from the start of the software development lifecycle instead of implementing security practices at the end. It is easier to test for security flaws and to fix them as the development process progresses on a project than to catch these flaws near the end and attempt to fix them. Also, it’s more cost effective to fix these flaws in the early stages of development.

When prioritizing which violations of the coding standards to fix first, it’s important to use the risk assessment associated with the violations of the coding standards. The risk assessment indicates how severe the consequences are if a vulnerability is not fixed. If several vulnerabilities are found in a program, developers will know that the most critical vulnerabilities are repaired first.

The zero trust policy helps to prevent security breaches and mitigate the impact of a breach if it occurs. It states that no user or device can be trusted until it’s authenticated. One of these authentication techniques is multi-factor authentication. Even after authentication is verified, there is a principle of least privilege that is applied to an authenticated user or device where they are authorized to the minimum level of access to a system required to complete their job function (Kueh, 2020). This is useful in the event that if the user’s account is compromised, an attacker would have only partial access to a system and its data. This could provide time for the cyberattack to be detected during accounting. The zero trust policy focuses on protecting data even when security has been breached by adding more layers of security to prevent a data breach.

In addition to adopting a zero trust model, I’ve learned the importance of continuously updating and implementing security policies and tools as new threats emerge. Developers require support through training sessions and resources to effectively implement zero trust principles and adhere to secure coding standards. This includes training on automated tools that can proactively detect vulnerabilities and cyberattacks. Moreover, providing developers with regular feedback on their adherence to security standards will facilitate their transition to a culture rooted in security awareness.

**References**

Kueh, T. (2020, January 17). A practical guide to Zero-Trust security. *Threatpost*. https://threatpost.com/practical-guide-zero-trust-security/151912/

Seacord, R. C. (2013). Secure Coding in C and C++ (2nd ed.). Pearson Technology Group. https://mbsdirect.vitalsource.com/books/9780132981972