Brittney Miller

SNHU CS 405

Journal 8

When thinking about secure coding and reflecting on everything I’ve learned in this class, adopting a secure coding standard is a must. With a secure coding standard, you have more reliable secure coding throughout your security and DiD strategy. When you follow a specific standard, you lessen the possibilities for unknown vulnerabilities to pop up within your security and help remove errors that make the system behave unexpectedly.

Not leaving security to the end is a way of saying include secutiy throughout your planning phases and processes. Security should be at the forefront of the architecture of the program you are building as it is way easier to plan it throughout rather than find space for it after the project is completed, and it’s an afterthought. Thinking of security from the beginning ensures you prevent risk. In contrast, if you wait until you are finished, the program may require extensive rewriting and testing, which means more costs associated with security that are done at the last minute.

Evaluating and assessing risk and cost benefits helps ensure you know the risks, costs, and benefits of switching to a particular practice or program. When considering how this is showcased, most coding standards will have this information as a table. Included in the table will be the threats, including the level, severity, importance, and cost of implementing the new practice, as well as the priority of the threat to deal with first.

Zero trust coincides with the Triple-A policies and helps ensure a standard for secure coding. Zero trust is the opposite of most coding standards currently, which you can trust, but be sure to verify before allowing access. Zero trust means not to trust automatically at all and to just verify everything. Zero trust is typically a lot more preventative when it comes to breaches or data being accessed by users who are not allowed access to data. Because you distrust everything, you verify and ensure that you are talking to who they truly say they are and giving access to users who are meant to have access to information.

The implementation and recommendations of security policies should be closely followed by their coding standards. DevSecOp is a great standard to follow because it covers both pre-production and production phases to ensure safe, secure coding throughout the process. It also allows everyone on the team to think in a similar format, so there are no differing opinions or how things are created and functioning. This coincides with the zero-trust policy and ensures information and programs are less likely to be vulnerable.

**References**

Foster, S. (2018). *What Are Security Standards? Secure Coding Standards Overview | Perforce Software.* Perforce Software. <https://www.perforce.com/blog/qac/secure-coding-standards>

Pratt, M. K. (2018, January 16). *What is Zero Trust? A Model for More Effective Security* CSO online. <https://www.csoonline.com/article/3247848/what-is-zero-trust-a-model-for-more-effective-security.html#:~:text=Zero%20Trust%20is%20a%20security,to%20don’t%20trust%20anyone>.