Jael Andre

CS 405

Module 8 Reflection Journal

Professor Andujar

04/24/2025

Looking back on everything we’ve covered in this course, one thing that really stands out to me is how essential it is to treat security as a core part of the development process—not something to tack on at the end. Adopting a secure coding standard early on helps set the tone for everything that follows. When we follow established guidelines, like those in the SEI CERT C++ Coding Standard or OWASP's best practices, we give ourselves a strong foundation to build safer software. It’s not just about checking boxes; it’s about writing code with security in mind from the start so we can avoid problems that are much harder (and more expensive) to fix later.

Another key takeaway is the value of evaluating risk and really thinking through the cost versus benefit of different mitigation strategies. Not every risk needs a high-cost solution, but ignoring risk altogether isn’t an option either. What stood out to me is that it’s all about balance—knowing which vulnerabilities are most critical, how likely they are to be exploited, and what level of effort makes sense to address them. Structured approaches to risk assessment, like using NIST frameworks or even basic threat modeling, give teams a much clearer picture of what’s worth addressing and why.

The idea of zero trust also shifted how I think about protecting systems. It emphasizes that trust shouldn't be automatic—not even within our own networks. Verifying every user, device, and connection helps create layers of defense that make it harder for attackers to move around if they do get in. This concept aligns really well with the principle of least privilege, and though it can take more work to implement, the payoff in terms of security is huge.

Lastly, none of this really works without solid security policies to support it all. Clear policies help ensure that everyone—from developers to IT to management—is on the same page about expectations and responsibilities. They also provide structure when responding to incidents or changes, which helps reduce confusion and improve response times. What I’ve learned is that security isn’t just about code—it’s about people, processes, and culture too.

All in all, this course helped reinforce that good security isn’t something that happens by accident. It’s something you plan for, build into every step, and constantly review and improve. And honestly, when it’s done right, it supports innovation rather than getting in the way.