CYBR2100 Reflection – Week 5

What I Learned

This week I learned how policies, standards, procedures, and guidelines all connect in an organization. Policies set the high-level rules, standards make those rules measurable, procedures explain the exact steps to follow, and guidelines give flexibility when strict rules are not practical. What stood out to me was how this stack makes accountability clearer, because it shows who sets the rules and how they should be applied in daily work. Owners and approvers are important because they decide what is authorized and sign off on changes, while review cadences make sure the documents stay current. Enforcement ensures the policies actually matter, because without consistent enforcement they are just words on paper. The *Ethics in Technology* eBook tied this to professional responsibility and harm prevention, stressing that ethical conduct depends on more than just good intentions—it also requires structure and follow-through. The ACM Code of Ethics connects directly to this by requiring computing professionals to follow organizational policies and respect privacy and integrity. Seeing how policies and codes line up showed me that ethical behavior and policy compliance go hand in hand, and both are needed for safe testing and responsible incident response.

How I'll Apply It

In practice, I think the most essential part of a Rules of Engagement document is the scope and authorization section. Scope makes it clear which systems and accounts can be tested and which ones are completely off-limits. Authorization ensures that no testing happens without formal written approval from the system owner or CIO, so the activity is accountable and legal. Without these two clauses, a penetration test or security assessment could easily go too far, collect sensitive data, or create harm outside of what was intended. By setting boundaries up front, scope and authorization protect both the tester and the organization and give everyone confidence that the work is being done responsibly.

Muddiest Point

I am still unclear about exactly when to trigger escalation during a security test. Should every unexpected error or odd result be escalated right away, or only if the issue clearly affects systems outside the approved scope? I worry that over-escalating could slow down work, but under-escalating could let a problem spread. Finding the right balance between caution and efficiency is still confusing to me.

Portfolio Note

- I will add my Rules of Engagement outline to show that I can define clear ground rules for safe testing.
- I will include a short rationale about why scope and authorization matter most, since they protect both testers and the organization.
- This matters because publishing clear ROE examples demonstrates accountability and professionalism in cybersecurity practice.

AI Use Note

I used ChatGPT as a structured writing partner. I provided my lecture notes and the assignment requirements, then asked the tool to guide me through the reflection step by step with targeted questions. This let me organize my thoughts more effectively while keeping the final writing in my own words. I see AI as a professional support tool — I controlled the content and direction, but used ChatGPT to make the process faster, clearer, and more organized.