CS 405 Secure Coding

Module 8 Journal: Portfolio Reflection

Joseph Caron

4/26/25

When it comes to cyber security in general, and specifically with writing code, it is far easier to ensure everything is secure by adopting a standard. If you have a standard set of rules to follow to keep things secure in your code writing, it not only becomes easier to write that security into all of your code but also gives peace of mind that as long as everything follows the standards, it is secure, and you needn’t worry about it as much. When you don’t make sure to code with security in mind, and instead leave it to the end, it often needs to be “shoe-horned” in, forcing it to fit where it wasn’t meant to in the first place. This can lead to a lot of rework, either refactoring the entire code so adding security doesn’t break it or tweaking the security methods so that they don’t mess up the functionality but then ends up being less secure as a consequence.

One of the larger things to consider is cost/benefit of secure practices. Some risks are just too small and too unlikely to be worth spending valuable time and effort, not to mention money, on. Others, while they may be very expensive in all ways to fix, are just too risky and too likely to be an issue to ignore or skimp on. It’s for this reason that risk assessments and cost/benefit analysis of mitigation should be done with any security policy. Tools like a threat matrix can help in this regard, categorizing different threats based on how likely or damaging they can be, to more easily decide what can and should be taken care of.

Another key point when it comes to security is the Zero Trust model. In today’s world of network and remote access, such as people working from home or elsewhere, network security is more crucial and finicky than ever before. You can’t just block out everything that isn’t physically within reach of the network access point, but you can’t just leave everything wide open either. Instead, you much allow access, though highly authenticated and based on verification that the person trying to access the network is who they say they are and that they have authorization to access it in the first place. It’s called Zero Trust for a reason, trust no one, make them prove they are trust worthy every step of the way.

All this put together, secure coding standards, not leaving security until the end, risk assessment and cost/benefit analysis, and zero trust methods, all bring me to the final point. Implementation and recommendations of security policies. Zero Trust is a policy. Having coding standards is a policy. Knowing what risks need to be handled and how, and which are so minimal and unlikely that it is more worth it to just ignore them, is a policy. The last point is also key to expanding on the policies and setting new ones. Security policies shouldn’t be set in stone, but should be able to grow and change as time marches on and new technologies are developed. As technology moves forward, the way security is looked at needs to evolve with it to keep everyone and their personal information safe and secure.