Dominic Clapper

April 21, 2023

CS405

Module 8 Journal – Portfolio/Course Reflection

After working through this course, I feel like I have learned a lot about security in coding and secure coding standards being put to use. There are so many different coding principles and standards regarding security, that it was a little difficult to get through it all, but I have gained a general understanding of the basics. In regard to adopting a secure coding standard, I completely understand how useful and necessary it is in the software development field. There have been many instances in my past where I have created code that is not secure in any way, shape, or form. This course has nailed it into my brain that I need to follow secure coding standards if I want my code to be secure for myself and users, and if I want to save myself time later in the development process. I know for certain, that I will now try my best to practice secure coding standards whenever I develop, as well as use all the tools that are available to help me follow coding standards.

As for the topic of not leaving defense/security until the end, I personally believe that this statement should be followed 100% of the time when developing code. Security is an integral part of the development process, and it is much more beneficial to start working on security for a project as early as possible. Instead of waiting until the end of the development life cycle to work on security, it is better to start working on it in the design phase of the development life cycle. This is because it will save you time in the long run, as well as make it easier to develop code that is more secure. There was also a lot in this course that discussed the DevSecOps process, which basically made sure to make security a priority throughout the entire development life cycle. I feel that the concept of the DevSecOps cycle and making security a priority throughout development is crucial to developing secure code that functions properly. (You should never wait to start implementing security standards)

As for evaluating risks, it is a very tedious and difficult process because of how much it matters to the entire development process. There are a lot of questions related to risks, which makes it very hard to assess them. Thus, in order to assess risks, the best option is to follow a small process that lets you identify the risks, define them, analyze them, and produce an evaluation of them. It is probably best to follow this process as early in the development process as possible. This is because risks can define how we go about developing the security for our systems/projects. The cost benefit of mitigating these risks is substantial. Being able to prevent these risks from occurring, or limiting their occurrence as much as possible is a part of working on the security for a system. The risks for a system help us find out which security vectors we need to think about, and the mitigation efforts are us implementing the security layers.

Regarding the zero-trust policy, I feel that it is a pretty useful policy to implement in most software security scenarios. However, there are some scenarios where a zero-trust policy may not be useful for a piece of software. From my understanding, the zero-trust policy is the practice of not trusting anybody trying to access a system and verifying them every single time. This policy can be very useful because it provides an extra layer of defense that makes it much harder for attackers to enter a system. You can add even more layers of security through the zero-trust policy by including multiple steps for authorization in the system. Having these extra stops forces attackers to have to go through multiple layers before getting to where they want to go in the system. Personally, I like the zero-trust policy because it feels like the policy/practice of not trusting any data coming into a system.

When it comes to implementing security policies, I feel that the best option would be to take it slow and inform employees as much as possible. By this, I mean that it would be best to slowly implement the policies by informing the developers of how things should be done. There can even be test projects where you have the developers train in using the security policies, so that they can put what they learned into practice. Another thing that would help with implementing the policies and assisting the developers would be providing them with adequate training about the security policies and how they should be applied. As for recommendations regarding security policies, I feel that it is important to get feedback from the ones practicing the policies. They are the ones who will be following the policies, so they should have the most weight in the decision regarding the policies. Overall, I feel that the best recommendation regarding security policies is to try to follow them to the best of your abilities, and to use all the resources related to the policies as much as possible.