CS 405 Portfolio Reflection

CS405 23EW6

Jared Smith

8/11/23

In this class we discussed the benefits of adopting a secure coding standard. There are a variety of benefits to adopting a secure coding standard for your project, the primary benefits being a uniform coding style across multiple developers in a project, and the ability to clearly define and mitigate against potential security threats. Adopting this standard also helps a dev team follow the standard of not leaving security to the end. Identifying potential security risks from the beginning of a project and planning on how to mitigate against them gives all the developers a guideline to develop against. This means that no matter how convoluted a project gets, we can always keep in mind the security of our project. When you think about it logically, it just makes sense that developing secure code is much safer than developing only functional code and trying to layer security on top.

The concept of a secure coding standard or security policy also goes hand in hand with the security concepts of Zero Trust development and Defense in Depth. Zero trust is a policy that treats any and all users of a program or application as hostile to that application. This helps us ensure that any breach in authentication doesn’t immediately result in a breach in authorization or access. Defense in depth describes a similar practice of layering defense throughout your program. Just because a user has passed one security check doesn’t mean they should have full access or not have to be authenticated again. These are considerations that are helpful when developing a security policy and secure coding standards.