Module 8 Journal: Portfolio Reflection

Adoption of a secure coding standard, and not leaving security to the end

I learned a lot about coding standards in this course. Before I started this degree, I always wondered about the error messages I would get while coding, and how they work. I now understand how it is that code can be so vulnerable, and how to better protect against those errors. These coding standards gave me a better insight into how the problem works and how to correctly perform common actions. The adoption of a secure coding standard is paramount to any development team and should be implemented often.

NEVER leave security to the end. Even before I fully understood how security works in software, I knew that it was not something to be left until the end. There are so many points of entry to a program that have many vulnerabilities and it’s so important to keep these points secure. Front end, back end, APIs, and caching are all examples of areas we must implement security to ensure our software is safe.

Evaluation and assessment of risk and cost-benefit of mitigation

Evaluating risk should be an important part of your development process. All parties play a part in the security of the software being produced, and evaluating any risks that may or may not exist can be paramount in security. It makes sense to strengthen the areas with increased risk for attack. At the same time, it’s also important to have processes in place to mitigate any damage caused if an issue does arise. For example, if a 3rd-parties database has been breached, having a process in place to shut down that connection and keep any new data from entering the breached servers is a good idea. We can’t always predict when an event take place, but just like a fire drill, having processes to help prevent further damage is a smart move.

Zero trust

Zero trust is the art of trusting no one and incorporating authorizations through every aspect of an application. Keeping the data and session secure is paramount to maintaining security and validity in an application. This ensures that your application and its data stay as safe as possible during use. From the front end to the servers, every stop along the way is authorized.

Implementation and recommendations of security policies

As mentioned earlier, a security policy should not be optional for any development team. Implementation should be mandatory, especially when dealing with user data. I think the security policy we made in class was a great start to a real one. While there are many benefits to implementation, the number one priority should always be protecting your user/company assets. Keeping this fact in mind, it shouldn’t be an issue to recommend a security policy to a team. Implementation, however, would be more difficult. People get stuck in their ways and refuse to change sometimes. This is especially true for freelancers that have been doing it their way for a long time. However, protecting assets is of upmost importance and should be valued and protected using the correct security practices.