Journal

Portfolio Reflection

CS-405

Caroline Wilson

April 2021

I believe that my biggest take away from this class was the further cementation of the need to test early and often. From this course, I was able to see why it is so important that this is done throughout the entire development process, but just shining in a different light that I have been used to. With the assignments that have been given throughout the weeks, it became entirely apparent that the adoption of a secure coding standard and not leaving security to the end is fully encompassing best practices that I have learned along the way.

Adopting a secure coding standard, to me, is the representation of a firm understanding of what needs to be done to produce quality work. Setting standards for yourself and expecting nothing that is less than what you are fully able to bring to the table is how you are going to achieve success in this business. Security is all part of that. If you want to produce quality functional work, then you need guideline to adhere to. If you want a good reputation, you need to ensure that the code that is developed is secure and that it functions well. Following a set of standards allows for this. By following the DevSecOps pipeline and adhering to the 10 core principals, you set a framework that is easy to follow and allows for comprehensive testing. Ensuring that the code you are producing is free of errors and vulnerabilities could dictate how far you are able to go with your career and further aspirations.

The easiest way to ensure this success is to not allow the security to be an afterthought. Just as you should test code for functionality as you go, it is important to do the same for security. While an error may not be that of compilation or runtime, there could be vulnerabilities that you overlooked in your stylistic way of handling a task at hand. Something so small as checking a pass code against a string could lead to major vulnerabilities. It is no secret, that especially in DevOps, there is more than one way to skin a cat, and that finesse is something that you learn over time. However, there is still a right and very often a wrong way to handle a scenario. By using automation, it is easy to see things that may not have been noticed initially. Getting into this practice can help ensure quality work in the long run.

Over the course of this class, I learned not only how to check for errors and oversight in security, but how to think critically about how to upend those issues and present secure code. With the tools that we are given, it is possible to think outside the box and look at scenarios critically in order to adapt our solutions for a more secure environment. Having the tools for testing, as well as a framework laid out with steps to follow in order to ensure and demand this of the finished product simplifies the process and gives more room for critical thinking and problem solving.