Module Eight Journal - Portfolio Reflection

Daryl Miller

Southern New Hampshire University

CS-405-T2675 Secure Coding 21EW2

December 13, 2021

**Content**

[Secure Coding / Not Leaving Security to the End 3](#_Toc90284343)

[Risk and Cost Benefit of Mitigation 3](#_Toc90284344)

[Zero Trust 4](#_Toc90284345)

[Security Policies 4](#_Toc90284346)

# Secure Coding / Not Leaving Security to the End

This class was truly eye opening in respect to secure coding. Secure coding is extremely vital to any successful program. Implementing things such as Defense in Depth (DiD), security principals and standards, Triple-A framework and other various security protocols will ensure a strong security foundation. With that, security must be at the forefront of the development of the program. Security should be considered and implemented at every stage of the software development life cycle (SDLC) and should not be left to the end of the development. By taking a proactive security approach, the developers can at least establish a solid security foundation and then manage and upgrade as situations dictate.

# Risk and Cost Benefit of Mitigation

Risk associated with anything is inevitable. There is not truly a 100% way to guarantee the security of any program. To provide the securest program possible, a full risk analysis should be done on the program while adhering to a strict risk appetite. The risk appetite is the how much risk the company or developer is willing to accept before the project is ultimately not viable. In addition to risk, a developer must understand the cost of mitigation increases with every phase of the SDLC. Ideally, all bugs and vulnerabilities will be identified and corrected as early as possible. As the program progresses through the different stages of the SDLC, it becomes increasingly more expensive and time consuming to fix the bugs. The worst possible scenario is to find and correct bugs after the program is already in production.

# Zero Trust

The concept of zero trust was something that I used to get frustrated with. Before this class, I did not really enjoy the concept of zero trust or authenticating each time I logged into something. After having taken this course, my opinion of zero trust has dramatically improved and I know understand the reasoning behind zero trust and concepts such as 2-factor authentication. I would rather spend the extra 5-10 seconds verifying myself than leaving me potentially open to hackers. I do believe a policy of zero trust should be implemented in any network and combined with other security principals and standards.

# Security Policies

Through project one and two, it is clear how important it is for a developer or a company to have a clear set of security policies and standards. In this course we adhered to the 10 policies and standards given to us, but there are many more that can be implemented as applicable. Understanding each policy and standard will ensure that everyone is working from the same set of guidelines to help reduce the introduction of errors and vulnerabilities.