8 – 2 Journal: Portfolio Reflection

CS – 405

Southern New Hampshire University

8/20/2023

Dan Taylor

The statement “Don’t leave security to the end” can be explained simply by acknowledging that security should be a priority starting in the earliest stages of the SDLC (Software Development Lifecycle), planning and analysis, and continue to be a priority throughout the design, implementation, and maintenance of any project or infrastructure. As a developer the goal is to always keep security in mind when writing code, and one should consistently test code for vulnerabilities and fix any that are found.

Throughout the course we have learned and implemented many of the steps a developer can take to prevent threats. It starts from the beginning during planning and analysis there are many security variables that should be considered and resolved before any code is written. It is important to plan multiple layers of security, and to plan how data will be collected, used, and discarded as necessary, and it is very important to plan who will have access to data within the system. The previous examples are only a few of the many points of security that need to be considered. Some useful steps a developer can implement during the development phase include unit tests, encryption, and static testing. These steps will help ensure code is secure against many potential threats when implemented as part of a wholistic security protocol.

Security threats can come from all areas, whether it be internal or external. Risk associated with daily decisions should be the driving force in any decision-making processes. We are often reminded that the threats of today are not the threats of tomorrow and there are huge consequences for not addressing any and every potential threat. The cost of addressing the threats is small compared to the cost of mitigating the risks associated with each threat. Implementing very simple coding techniques, testing early and often, and performing static code analyses after every code amendment are a must. We also have access to automated unit testing and penetration testing that need to be done on a consistent basis.

Zero trust is concept in that the organizations should not trust anything whether it is inside or outside and instead, they should verify everything which they are trying to connect to its system before granting access. The zero trust concept is based on authentication, authorization and validation of security configurations and posture before being granted to data or applications. The zero trust policy forces me to think about incorporating procedures like least privilege, default deny, AAA, encryption and so on.

Security policies give us a guideline to follow to ensure that code and data remain safe. Creating, maintaining, reviewing, and following security policies are a must for every company. Security policies contain core principles, coding standards, and best practices that need to be incorporated into every development project. This will allow for consistent implementation for safety when security policies are followed.