Journal: Portfolio Reflection

Joshua Pardue

CS-405 Secure Coding

Professor Aaron Demory

February 24, 2023

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

The process of development should be constructed considering security issues and vulnerabilities. If the program is designed with security in mind, these issues and vulnerabilities are easier spotted early in the process. With professionalism, quality of code, and best practices in mind, the code looks and performs more efficiently as well because patches and fixes do not have to be implemented after the completed product is rolled out. The security is built into the code and can be adjusted and tuned accordingly and not just an inefficient after thought. From a cost standpoint this plan makes sense also as it is easier for developers to address an issue early on in development versus spending time and money to backtrack and re-develop a certain portion.

*Evaluation and assessment of risk and cost benefit of mitigation*

Ideally, with a multi-layered DiD, if one system fails to negate an attack, another system steps in and takes care of business. These varying layers of security come with a specific order and price tag to consider. As stated in the previous paragraph, with an established equilibrium, DiD requires analysis in determination of the necessary layers an organization or individual requires for operation. While no system is ever protected with only one layer, time and money could be an issue as it takes a considerable amount of both to establish a reliable DiD. A diverse layering system equipped with network security, antivirus software, behavioral analysis, and many other protections might not be economically and financially feasible for a small organization only catering to a small amount of traffic versus a huge conglomerate supplying millions of users every hour.

*Zero trust*

As more business operations and even individuals move online, cyber issues are becoming more frequent. Attacks can result in data breaches, ransomware, and other types of cybercrime that can have serious financial and reputational consequences. Zero trust is a security model that is based on the principle of never trust and always verify. The specifics of this model can be incorporated to address the limitations of traditional perimeter-based security models that rely on boundaries to keep attackers out

*Implementation and recommendations of security policies*

There are a few steps and mechanisms that can be utilized to deter threats and malicious individuals from compromising the software or network. Formulating defense in depth strategies and implementing multiple layers with a combination of technical and non-technical security measures are crucial for prevention. Regular security assessments along with backing up data ensures processes are secure. Monitoring network activity and usage of firewalls will dissuade most individuals and usage of encryption for sensitive information will ensure data is protected. Another key factor is having a plan for when things do go wrong so further threats and attacks are mitigated. An established plan will ensure preparation and quick response to effectively put the fire out. While there are many tools that would ensure efficient security measures are taken, the use of automated security testing in general provides an avenue to test an application faster than manual testing. It allows for testing potential vulnerabilities quickly and efficiently. It also allows for better coverage as tests are executed over many test cases and scenarios.

References

Malik, K. (2022, August 11). *All You Need to Know About Automated Secuirty Testing Tools*. Retrieved from Astra: https://www.getastra.com/blog/security-audit/automated-security-testing-tools/#:~:text=Automated%20Security%20Testing%20is%20the,from%20being%20exploited%20by%20hackers.

*What is Defense in Depth?* (2023). Retrieved from Forcepoint: https://www.forcepoint.com/cyber-edu/defense-depth