Marcus Spruill

CS-405: Secure Coding

August 25, 2024

Journal: Reflection

**Introduction**

The public has greatly increased its reliance on cloud computing services, thanks in part to the proliferation of internet services and portable computing devices. Financial institutions, governments, and healthcare providers offer online services for users to retrieve or update records, request services, or access other features. Those industries store sensitive information on customers; thus, it is paramount that sufficient software security is implemented to prevent damage caused from unauthorized access (Joshi, 2024).

**The Importance of Secure Coding Standards**

Software security should be implemented throughout the development process to ensure that vulnerabilities are detected and there is sufficient time to test solutions. The Development, Security, and Operations practice is one such development process that implements security integration during the software development lifecycle. It avoids the pitfalls of leaving security to the end of the development process, where shortfalls of budget, tight deadlines, or technical difficulties can prevent a planned security audit of the finished product. The secure coding standards are good practices to prevent vulnerabilities for memory, data types, exceptions, and so on, and should be followed to introduce as few issues as possible during development. Finally, defense-in-depth is a strategy to implement security measures in distinct layers, with the goal of detecting and neutralizing as many attacks as possible.

**Risk Assessment and Benefits of Mitigation**

While implementing added security measures during development does incur an additional cost, data breaches can cost the organization a significant penalty. Equifax announced a data breach in September 2017 that exposed personal data of 147 million people. It settled with the Federal Trade Commission, Consumer Financial Protection Bureau, and 50 U.S states and territories for up to $425 million to those affected (Federal Trade Commission, 2024). Security implementation costs can be reduced by utilizing automated tools, such as static code analyzers, to quickly and thoroughly detect vulnerabilities (Foster, 2023).

**Zero Trust and Permissions**

Avenues of unauthorized access do not only arise from unsecure code. Permissions policies dictate which users have access to resources and the level of control available. A “zero trust” security policy describes that actions should be denied by default, with required permissions granted to users and groups. This policy limits the damage caused by malicious actors and negligent users while maintaining productivity. Zero trust also supports the Triple-A framework through the user-permissions relationship.

**Recommendations and Conclusion**

In conclusion, testing strategies such as unit testing, and automated testing tools should be implemented to detect vulnerabilities. The coding standards are good practice to prevent introducing vulnerabilities in the source code. Defense-in-Depth provides a structure for defending against varied and dangerous threats and should be utilized and maintained.

References:

Joshi, A. (April 22, 2024). *These sectors are top targets for cybercrime, and other cybersecurity news to know this month.* World Economic Forum. <https://www.weforum.org/agenda/2024/04/cybercrime-target-sectors-cybersecurity-news/>

United States Federal Trade Commission. (February, 2024). *Equifax Data Breach Settlement.* <https://www.ftc.gov/enforcement/refunds/equifax-data-breach-settlement>

Foster, S. (June 7, 2023). *What Is Static Analysis? Static Analysis Tools + Static Code Analyzers Overview.* Perforce. <https://www.perforce.com/blog/sca/what-static-analysis>