Journal: Portfolio Reflection

Module 8

Marta Taylor

CS-405 Secure Coding

Olga Mill

Southern New Hampshire University

April 19, 2021

**JOURNAL: PORTFOLIO REFLECTION**

**SECURITY FROM THE START**

Security from the start refers to incorporating security principles and practices into the planning process as well as the day-to-day operations. It means being proactive by adopting secure coding standards and complying with those standards right out of the gate – before code is even written. Considerations include the identification of the different kinds of information collected (personal, financial, etc.), data storage management (how long is data retained), what permissions should be granted to individuals or groups (read, write, execute) and tracking of how and when data are accessed (used or transferred) (FTC, 2015). If an organization considers what type of data is to be collected, maintained, used and/or stored at every step in the operation, and all this information is contained within a working, living security policy, the risk of a breach is reduced because the prevention facet of Defense in Depth has been implemented.

**MITIGATION OF RISK**

Threats can be internal as well as external, and risk should be the driving force in data management decision making. There will always be every degree of threat present, and the threats of today will not necessarily be the same as the threats of tomorrow. The consequences of not addressing every potential threat are dire as compared to the cost benefit of mitigating the risks associated with each threat. By employing simple coding techniques, testing early and testing often (ie: daily), performing static code analyses after every code amendment, consistent and automated unit testing, third party penetration testing – these are all tools available to developers to mitigate risk. Most of these tools are inexpensive to employ, especially when compared to the severe consequences of a breach.

**ZERO TRUST**

Traditional authentication methods are failing and are in need of bolstering. By assuming zero trust, 100% of the time, verification is performed for every user, device, network, data access and application (Kueh, 2020). This is the “never trust/always verify” model. For user verification, biometrics, certificates and multi-factor authentication are replacing usernames and passwords. Devices are being catalogued, monitored, managed and controlled. The security policy dictates whether a device can be trusted and if it is compliant.

Network or session trust is based on the least privilege access principle, such that a user or system only has access to the absolute minimum number of resources needed to perform a function. Applications are being modernized to allow the use of single sign-on (SSO) access. And data trust, which is the whole reason for the need for secure coding, can be achieved through data loss prevention (DLP) technologies.

**SECURITY POLICIES**

The importance of creating, maintaining, and reviewing a living security policy is no longer an option if code and / or data are to remain safe. The security policy contains the core principles, coding standards and best practices required to be followed and incorporated into every development project. Software development requires consistent implementation of secure principles for all projects crafted by all developers. Consistent approaches and methodologies must be maintained through all policies that are uniformly defined, implemented, governed and maintained over time.

**BEST PRACTICES**

There are several lessons learned recommended by the FTC based on past law enforcement actions (FTC, 2015). These suggestions speak to common vulnerabilities that affect many organizations. Some of these lessons, in addition to starting with security include controlling access to data; requiring authentication; storing and transmitting data securely; maintaining a log of activities; securing the physical devices that the data is stored on; and employing a living security policy to keep security practices, policies and procedures current. Finally, third party software as well as network providers should also be implementing security measures.

Additional practices should include keeping software up to date and ensuring that all patches are applied promptly; performing unit testing using range boundaries; requiring passwords are changed periodically and have requirements such as using special characters or numbers and letters together; train employees to be leery of suspect emails or links; never leave devices unattended without logging out; keep security software up to date and install malware as well as a firewall; or use two step authentications for logins. All of these best practices add a layer of Defense in Depth and can help mitigate the likelihood of an attack.

**REFERENCES:**

Federal Trade Commission (FTC). (2015). *Start with Security: A Guide for Business*. Retrieved from <https://www.ftc.gov/tips-advice/business-center/guidance/start-security-guide-business>

Kueh, Tony. (2020). *A Practical Guide to Zero-Trust Security*. Retrieved from <https://threatpost.com/practical-guide-zero-trust-security/151912/>