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

The adoption of a secure coding standard early in the development process is critical to the success of any product or software. Not leaving security to the end is a mentality, approach, and ultimately a best practice in secure coding. The security mentality needs to be as essential to the development lifecycle as oxygen is to the body. That means, from the initial stages where the software is simply a thought, all the way until the software is in its maintenance stage or has been discontinued. The benefits of early implementation are building and ensuring customer trust and brand respect, mitigating threats, reducing overhead testing, and the protection of internal and external data.

From an article on techspective.net, Tony Bradley writes:

“There are two very important reasons to incorporate security principles and design early and throughout the development lifecycle-cost and efficacy. Security is often not considered until a product is essentially ready for release. At that point, most of the design and engineering decisions are pretty much set in stone and it would be very costly to try and go back and fix security issues that are baked into the core of the product.” (Bradley, 2021)

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

The process of assessment involves identifying threats and vulnerabilities to prioritize which risks to address first based on their severity and cost of mitigation. A cost-benefit analysis should guide decisions on whether the severity and cost of mitigation justifies the investment of a security measure. For example, let’s say a company who handles overly sensitive customer data identifies a breach due to weak encryption. The cost-benefit analysis would reveal that the initial investment into AES (advanced encryption standards) is rather high. However, the potential financial and reputational damage from these types of breaches outweighs the investment cost.

* Zero trust.

The Zero Trust model, or mindset as I like to call it, aims to enhance cybersecurity by minimizing risk and refining outcomes. It is a mindset we must adopt. We must think about potential threats that can occur anywhere internally and externally. For example, this could involve implementations such as continuous verification of user identities and devices rather than a one-time verification. Much like a financial institution that implements multi-factor authentication (MFA) for all users ensuring unauthorized access is prevented even if credentials are compromised. It will also involve a least privilege access implementation where users are only granted access necessary to their role. But beyond these, the incorporation of zero trust must be an on-going mentality and culture that continuously monitors and never relents.

* Implementation and recommendations of security policies.

The implementation and recommendations of security policies involves launching rules and procedures that aim to shield data assets. These policies will cover areas like access control as mentioned above, data protection, incident response, training, etc. A good example of this can be thought of in healthcare organizations. They require complex but comprehensive security policies to protect patient information. This will include strict access controls, strong passwords, frequent/regular password changing, and reoccurring training sessions to educate staff about the ever evolving phishing and social engineering tactics. By implementing these measures, healthcare organizations create a robust security structure to maintain compliance with their regulations and shield their sensitive patient information.

Resources:

Bradley, T. (2021, January 20). *Better Security through the Security Development Lifecycle*. TechSpective. <https://techspective.net/2021/01/19/better-security-through-the-security-development-lifecycle/>