Sarah Kerr

CS405 T6652

August 21, 2022

Module Eight Journal

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

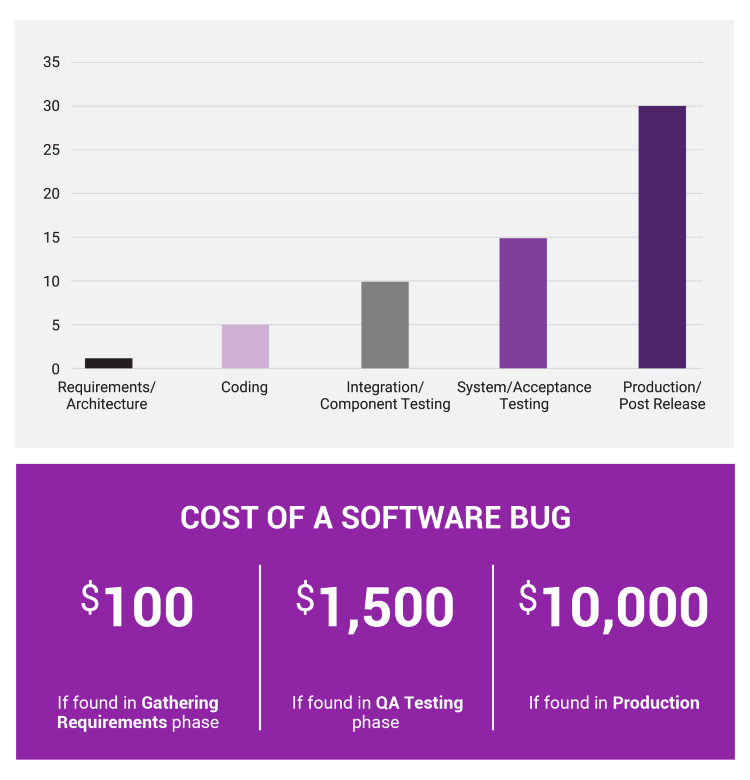
It is important to adopt a secure coding standard that all employees should be familiar with and follow. This policy would serve as a reference to known vulnerabilities and how to mitigate them. Doing so saves time and money in the long run.

Not leaving security to the end when it comes to secure coding means a few different things. First, it means that security should be at the forefront of the team’s mind. It should be a priority rather than an afterthought (Zorz, 2020). Secondly, it means that security should be considered at all stages of the SDLC, not just right at the end before deployment. This makes security integrated into the SDLC rather than something viewed as a separate entity. Finally, it also means that companies should develop security guidelines and update them regularly as well as ensuring all employees adhere to them, as mentioned above.

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

Prior to implementing a security policy, the company should be evaluated and assessed for risks. This is known as the security review. The entire attack surface of the company is scrutinized and all points of entry (areas where data comes in) should be considered vulnerability. The current state of the system should be checked against well-known security standards such as those outlined in the SEI CERT coding standard manual. Once the needed security standards are identified, they can be sorted into a threat matrix by their level of severity. This helps the team understand which security standards are offering the most protection.

As mentioned, the costs of waiting to implement security until the end of production go up exponentially. This chart clearly shows the exponential increase in costs of waiting to implement security(*Poor application security can increase costs by 3,000%*)**.**



**Zero trust**

Zero trust is a security practice by which the user must provide validation each time they access a different area of the system. The assumption flips from “this is a trusted user” once they get in to “this user can’t be trusted until they are again verified”. The past way of focusing on security at the perimeter, and then trusting all users once they have access, is out-dated. With more data being stored in the cloud than ever before, and more employees accessing these systems remotely, perimeter breaches are more likely to occur. If one is taking a Defense in Depth approach, security throughout the system is needed in the event the system is entered by a hacker.

**Implementation and recommendations of security policies**

It is strongly recommended that all companies develop a secure coding policy. As mentioned above, it would need to be started with an in-depth security review so that areas of vulnerabilities could be identified. From there, the company can make their own security standards based on the types of vulnerabilities they frequently encounter. The policy can then be implemented by training employees, starting off with implementing only a few standards (rather than trying to implement everything all at once which might be overwhelming for employees). It is important to also hold everyone accountable to the new security policies by doing frequent follow-ups and updating the policies as needed.

**References**

*Poor application security can increase costs by 3,000%*. dig8ital Cyber Security Services & Consulting Germany UK Australia Global. (n.d.). Retrieved August 19, 2022, from https://www.dig8ital.com/post/poor-application-security-can-increase-costs-by-3000

Zorz, M. (2020, September 9). *Plan for change but don't leave security behind*. Help Net Security. Retrieved August 19, 2022, from https://www.helpnetsecurity.com/2020/09/10/plan-for-change-but-dont-leave-security-behind/