Luke Kundinger

CS-405

8-2 Journal

Adopting a secure coding standard can lead to an efficient software development lifecycle with strong security. This standard works best when it is adopted in the initial stages and before development. Leaving security for the end of the SDLC can cause problems as it can be time-consuming and costly. Addressing security in the planning stages can create a guideline with security requirements to follow. This can lead to an efficient development process with little to no backtracking.

Evaluating risks during the planning stage can help prioritize security concerns and create security requirements. Considering the motive for attack can help developers determine what information needs to be protected. For attacks pertaining to identity theft, authentication methods would be top priority. The prioritization of security methods relates to cost benefit mitigation. Higher priority security methods will be worth using the extra resources on to keep the program secure. By evaluating risks developers can determine where their effort and resources should be applied in the SDLC.

The zero-trust policy means nothing can be trusted and verification is always necessary. It works by treating every user as a potential threat until proven otherwise. This policy can be applied by verifying users every step of the way and granting access on a need-to-know basis. The zero-trust policy may be inconvenient to users, but it does add piece of mind. With constant verification methods it can cause the user to put in more effort to get where they need to go. Users may be asked to confirm their identity with multi-factor authentication which would take longer than it usually would. This added security offers a piece of mind to users because attackers will struggle to commit identity theft with the verification methods.

Implementing a security policy serves as a baseline for developers to follow when creating a secure program. The policy chosen can vary for types of programs and what a company is looking to protect. For example, a large company may be looking to protect data and emphasize secure data handling methods with encryption. It is important to determine what security policy works best in the early stage of the SDLC. This can help create security requirements developers can implement to fit the policy.