Trevor Leon

CS-405

Portfolio Reflection

It seems clear that implementing security throughout the development lifecycle is better than leaving it to the end. If a vulnerability isn’t found until the end, then the vulnerable code won’t be changed to the end. If another part of the system uses the updated code, it might also need to be updated; leading to a chain of refactoring and plenty of wasted time.

Using automation tools proved to be very helpful. These tools can detect vulnerabilities in code for a developer and explain what needs to be done to fix the issue. Some vulnerabilities take more resources to fix and aren’t as worthwhile, while others should be easy to fix and come with a low remediation cost. These are graded by the *SEI CERT C/C++ Coding Standards* from L1-L3 with L1 being the most severe and least cost to remediate.

Zero trust is another important concept. It says that you shouldn’t trust the user unless they are properly authenticated. Even if you can trust a user, they should only be given the level of trust necessary to accomplish their goals and nothing more. For example, if someone leaves their company laptop somewhere, it can go to sleep and require a password to unlock it again. Just because someone has access to a physical device doesn’t mean it’s the owner/user.

Security policies should be implemented by any company. If a company is not properly secure through any of the various methods to penetrate applications or cybersecurity measures, it could lead to a data breach, denial of service, manipulation of data, or anything else a malicious user could imagine. If a company is known to not be secure, then customers will go elsewhere, and the reputation of the company will be on the line. Automation tools and the *SEI CERT C/C++ Coding Standards* make cybersecurity much easier for a company to implement.