Mark Holden

CS-405 Secure Coding

10/23/2022

Secure Coding Portfolio Reflection

A secure coding standard must be adopted, and software must be architected using that standard from the beginning of the project because that is the best way to avoid vulnerabilities and without putting it in place in the beginning, implementing secure practices later will be much more difficult. If security is left to the end, even if every security flaw that crept into the code over the development phase was caught and remediated, it would take much more time, and risks having to complete major refactoring in late stages of the project. If the coding standard is not fully adopted there could be subsystems that introduce vulnerabilities despite excellent secure coding practices in other areas.

As in everything else in life, the implementation of software security comes with a risk assessment. When deciding which controls to implement immediately and which to postpone, an assessment must be made of each of the risks posed by the project. With the risks identified, the severity and likelihood of each risk should be evaluated. Once the risks have been broken down to this level, controls can be implemented to mitigate each risk with the intent of reducing the severity or likelihood of the risk.

Zero trust is a great new security principle that eliminates many of the security vulnerabilities present in a system where a specific type of access, such as access to an internal network would provide access to other systems. Under that type of architecture, attackers would take advantage of the trust policies by infecting or gaining access to a less secured system that was not the actual target, and then using that access to gain further access, and leapfrog their way to the actual target. Zero trust policies prevent that because even if a user is on the virtual private network, they are still not trusted, and must use credentials or some means of authentication to prove who they are.

In my opinion, the paramount security policies that can be put in place are code reviews and the principle of least access. With code reviews, backed by a solid secure coding standard, all the other important secure coding practices will be followed. The principle of least access is another important aspect of security, but people must understand that the reason they are not given broad access to the whole system is not because they are not trusted as an individual, but because with credentials having such broad access, they become a target. As a target, their credentials could be stolen and used maliciously against the company and because they allowed their credentials to become compromised, they are responsible in part for the breach.