Mod 8 Journal

CS-405-T5550 Secure Coding 23EW5

Valerie J. Smith

June 21, 2023

Adopting a secure coding standard is a prevalent task for today’s applications. There are a lot of differences in today’s cloud hosted applications as compared to applications that were created years ago, and the security threats keep increasing. Businesses used to follow a “throw the work over the wall” approach to application security, where they coded the complete application and left the security work to another team or did not implement any at all. This type of waterfall development has caused many a lot of trouble and cost due to hackers and security vulnerabilities. The standard is to now implement security as a part of the software development lifecycle, and to move to a DevSecOps lifecycle. The DevSecOps development lifecycle will ensure that security is planned at every step, including coding standards and tools for the developers while they are coding.

Evaluation and assessment of risk mitigation is a benefit to a business to take a deep look at the tech stack and framework components of an application and thoroughly examine the architecture for any possible security vulnerabilities before they happen. It is important to look at existing security vulnerabilities and see if there is a potential for them to occur in whatever architecture the business has chosen by using various OWASP and other tools. For example, third party packages that are imported into an architecture may bring security vulnerabilities with them. By comprising a list and cost of these potential security vulnerabilities and planning to ensure the correct steps are taken to protect from them, the risks will go down. The money spent on good security measures will not add up to the cost of a security breach.

Zero trust is a shift in security standards that moves the security focus from the perimeter of the system to the users that are accessing the system. The motto is trust no one and verify everyone, no matter where they are coming from. A malicious character could be in a system for weeks without anyone knowing, and that is the point of zero trust. In the Zero Trust model, the security is now monitoring and validating the users, devices, and applications that are accessing a network. A focus is also put on access control in Zero Trust, which only allows the smallest access necessary for a user to get their work done to ensure that the correct people have the correct resources and nothing more.

The implementation of a security policy is very important to ensure that all stakeholders for an application are informed and working by the same standards. Recommendations in a security policy are based upon researching potential security vulnerabilities and setting coding standards and security principles that are enforced. Security policies should be updated to reflect the latest security vulnerabilities that may put an application or system at risk for them to be remediated quickly. The security policy outlines what technology stack the development team is using and points to specific security vulnerabilities that could occur and lists steps to remediate them, as well as the standards for automation tools for unit testing and security scans. The security policy outlines the DevSecOps pipeline and documents where and how security will be implemented during each step of the application development, deployment, and production monitoring. The security policy also sets recommendations for future use. The security policy outlines encryption and Triple A standards and definitions, as well as provides a Risk Assessment Summary. Enforcement and Exception processes are also listed in order to ensure that the entire team is aware. The security policy outlines the Defense in Depth process that will provide the stakeholders a concise view of the layers of security that are implemented in order to protect the application and system for the business.