Secure Code Review

Defining a Secure Code Review

Secure code review is a manual or automated process that examines an application’s source code. The goal of this examination is to identify any existing security flaws or vulnerabilities. Code review specifically looks for logic errors, examines spec implementation, and checks style guidelines, among other activities. Automated code review is a process in which a tool automatically reviews the source code of an application, using a predefined set of rules to look for inferior code. Automated review can find issues in source code faster than identifying them manually. Manual code review involves a human looking at source code, line by line, to find vulnerabilities. Manual code review helps to clarify the context of coding decisions. Automated tools are faster but they cannot take the developer’s intentions and general business logic into consideration. Manual review is more strategic and looks at specific issues.

Nowadays, application code vulnerabilities are a lucrative cyber security target in the digital habitat. So, what causes this susceptibility? Indeed, the lack of security observation in the early project development life cycle is the root cause of the security crisis. And here is where the Secure Code Review services benefit in readying your applications.

So, let us first explain the term in detail. A Secure Code Review is a manual or automated technique that examines an application’s code base to discover existing flaws and vulnerabilities. The process also checks for logical errors and inspects spec implementation and style guidelines.

Code review can be of two types – Manual and Automated Code Review. Automated Code Review involves a tool that reviews the application source code using a predefined set of rules. On the other hand, manual review involves a human element inspecting the source code line by line to detect susceptibilities.

Secure Code Review: Focus

Where does a secure code review plan its applicability? There are seven key areas/security mechanisms a Secure Code Review concentrates on. An application that falls short of protection in any of these areas could be easy prey for malicious users. A Secure Code Review helps identify gaps for the development team, illustrating the soundness of applications in these areas. So, what are these areas or security mechanisms?

1. Authorization
2. Authentication
3. Data validation
4. Session management
5. Error handling
6. Logging
7. Encryption

Secure code review: Process

As we already indicated, code review could be of two types – Manual and Automated. The current best practice of organizations involves using Manual and Automated Reviews together. Hence, the approach would be able to catch the most sophisticated threat vectors existing in your application codebase.

When do you require a Secure Code Review? For the best result, applications need code review in the initial stages of the software development lifecycle. It can help developers fast-fix existing flaws in the codebase and help improve application readiness in all ways.

Now, let us consider the situation where a developer is writing the code. A parallel automated review at this time can help incorporate immediate changes to the codebase. An Automated Review enables fast analysis of large codebases using open source or commercial tools. The advanced development team also uses SAST tools to fix vulnerabilities in real-time.

A Manual Review, on the other hand, gets useful in the commit phase of the project. It considers business logic and incorporates developer intentions. The process involves the mindful examination of the source code by a senior code review person. It is time-consuming but effective while looking to catch business logic errors or issues.

So, what is an ideal Secure Code Review Process? Indeed, the combination of both automated and manual approaches. The human element that the manual review involves is vital, and if you combine it with the SAST tool functionalities, it can enhance the overall security of the code. Additionally, it helps minimize the number of flaws or susceptibilities flowing into the production cycle.