Jake Steckel

CSC402

Article Review 6

Code Review Tools

This article is all about static analysis tools for automated code review. It includes how they are the most effective, how they work, and why all software developers should be using them. There are four main sections of the article that I will be reviewing. First is the rise of software security, which talks about how software security has grown since it first arrived. The next section explains why code review tools are important and how they impact software security. The article then goes over a brief history of code review tools. Lastly, there is a description of a few code review tools that are being utilized in today’s era.

When computer security first began surfacing, the key idea was to create a barrier between hackers and vulnerable machines. This idea was introduced in the late 1980s and was named a firewall. As computer security progressed, software security became much more prevalent. The idea behind software security is to engineer software that will continue to run smoothly even when the system is under attack by a hacker. Once the rise of software security began, it continued to progress at an incredible rate. The best practices for software security are architectural risk analysis and code review.

Code review tools are essential to software security because they make the lives of analysts much easier. Most security issues are a result of minor bugs in the code that can be fixed rather easily. For an analyst to review the code themselves, it is boring, tedious, and difficult. Unless you want the analyst to hate their job, code review tools should be utilized to locate all the minor issues. An example of a code review tool are the static analysis tools. Static analysis tools automatically search for common coding issues before a program is released. These tools can save analysts from a ton of unnecessary hassle.

The first code review scanner that searched for security issues in code was the ITS4. Since ITS4 was the first one to surface, it was extremely system. It worked by lexically scanning through a file, searching for matches based on a few simple rules that may indicate that there are vulnerabilities in the system. This tool is classified as a basic lexical analysis tool. The issue with lexical analysis tools are that they tend to produce a substantial number of false positives regarding security issues in the code. For the lexical analysis tool to continue advancing, it must learn to utilize more compiler technology.

For modern code review tools, there are a few major vendors that specialize in the building and selling of source code analysis tools. These vendors are Coverity, Fortify, and Ounce Labs. Each vendor takes a similar approach when creating their analysis tools but they are also different in their own ways. Since they are now utilizing compiler technology, their tools have become much more advanced. The rule set for their code review tools are one of the most essential parts of their entire system.

I found this article to be very interesting because I didn’t know how big of a role code review tools played in software security. When I enter the real world I certainly will not take them for granted.