CODE SECURITY REVIEW

A Higher Level of Software Security

Many organizations run source code checking tools before deploying into a production environment. However, automated checking tools are only equipped to identify common security bugs and anti-patterns. For a truly secure and resilient application, a manual analysis is required.

A Cursory Check Isn't Enough

Code security reviews, also known as static program analysis or static source code analysis involves analyzing software without actually executing software. This type of analysis is increasingly used in safety-critical computer systems, including but not limited to software in the medical, power, and aviation industries. Code security reviews are a must-have for any robust Software Development Lifecycle (SDL), and are a critical component to any legacy code re-write or overhaul. According to the Open Web Application Security Project (OWASP), code security reviews are the single most effective technique for identifying security flaws.

https://www.sans.org/reading-room/whitepapers/cloud/cyber-security-trends-aiming-target-increase-security-2017-37702

Often, software engineering organizations consider their code secure because they run automated code checking tools before security. The truth is, static code analysis tools provide a cursory check at best. After all, new vulnerabilities are discovered every single day. We will carefully review your software and identify critical security vulnerabilities as well as violations of best practices, security design issues, and much more. Our dual approach to source code review provides true assurance.

While attacks that exploit zero-day vulnerabilities tend to get the most press coverage, data continues to show that attacks that exploit well-known vulnerabilities will cause the vast majority of damage.

Vulnerabilities can be avoided by injecting sound security review practices into software development process. The least risky vulnerability to your organization is one that never makes it onto a product system, application or process. Extending vulnerability awareness and mitigations into software development processes and into the evaluation and acceptance criteria for developed or procured software not only efficiently reduces attack surfaces, but it has been proven to reduce time to market for secure business services.

According to SANS and Verizon's Annual Data Breach reports, it is estimated that over 80 percent of cyber security incidents exploit known vulnerabilities. More startling is Gartner's estimate that "through 2020, 99 percent of vulnerabilities exploited will continue to be ones known by security and IT professionals for at least one year." Unfortunately due to business technology trends, the quantity of zero-day vulnerabilities will increase simply because of the increasing number of products and operating systems that will be in use. However, your highest risks will remain and come from well-known and well-understood vulnerabilities. The key to reducing business damage is faster detection of vulnerabilities through sound code review along with more rapid and accurate mitigation AND that's where RevBits comes in!

A Deeper Review

Since every programming language is unique, our engineers will conduct a deep review of your software. We specialize in finding weaknesses in source code, and are are knowledgeable in all major development languages, including but not limited to: C/C++, Java, .NET, and VB.NET. We know where to look to find vulnerabilities lurking in your code before someone else exploits them.

A proper dual source code review will reveal bugs in your software that can threaten your security. We look deeper to prevent attacks such as XSS, CSRF, and SQL injections. We run both an automated analysis and a manual analysis for a truly deep review. This ensures that we don't miss anything that could jeopardize your security and core business.

Automated Tools

provide analysis without actually executing, or running, the software or applications in a non-runtime environment. This method of testing has distinct advantages in that it can evaluate both web and non-web applications. Through our advanced tools, we can detect input and output flaws that cannot be seen through dynamic web scanning alone.

Final In-Depth Review

is our final investigation that is based on the programming language's unique architecture in mind. Our engineers are constantly learning about new threat vectors and security-specific vulnerabilities that may pose a risk to your company.

Manual Code Review

Review is a more efficient review of code where we usually discover a number of common vulnerabilities such as access controls, poor encryption implementation, lack of data protection, proper logging, and back-end communications.

A RevBits code security review examines applications for over 9 code flaw categories. Examples of our in-depth testing are shown below

- Input validation - cross site scripting, SQL injection, XPATH injection, LDAP injection, cross-site request forgery, buffer overflow, format bug
- Source code design - insecure field scope, method scope, and class modifiers, as well as unused external references, redundant code
- Direct object reference - direct references to database systems, filesystems, and memory
- Application Programming Interface (API) usage - insecure database calls, random number creation, memory management calls, HTTP session handling, and strings manipulation
- Information leakage and improper error handling - unhandled exception, routine return value usage, NULL pointer referencing, insecure logging
- Resource usage - insecure file creation, modification and deletion, as well as race conditions, memory leaks, and unsafe processes creation
- Best practices violation - insecure memory pointer usage, NULL pointer dereferencing, pointer arithmetic, variable aliasing, unsafe variable initialization, missing comments and source code documentation
- Using HTTP GET query strings - passing sensitive data through a URL query string
- Weak session management - not invalidating sessions when errors occur, not checking for valid sessions upon an HTTP request, not issuing a new session upon successful authentication, passing cookies over insecure connections.

CONFIDENCE ONLY PAY FOR IMPROVED SECURITY

We've had experience in providing analysis for the finance, commercial, and government sectors.
And look forward to assisting you. We're so confident in our team's ability to perform source code reviews that if we don't find any critical vulnerabilities in your source code, we'll provide your report for free.