Digital transaction management company uses risk detection solution to streamline software testing

“We were able to automatically run millions of test cases across multiple virtual machines, entirely automated, with no extra work …. We’re always looking for services that add value, and that scale of automation is a great added value.”

—John Heasman, Senior Director, Software Security, DocuSign

For DocuSign, a global e-signature and digital transaction management (DTM) leader, keeping its customers’ data secure is a top priority—and as a result, the company recently evaluated the Microsoft Security Risk Detection service. DocuSign used the service to perform millions of tests running on multiple virtual machines that it can deploy with minimal effort. Test results are thorough, accurate, and easy to analyze, which strengthens DocuSign security, bolsters customer trust, and saves the company time.
**Strengthening security to earn customer trust**

Founded in 2003, San Francisco–based DocuSign provides e-signature and digital transaction management (DTM) services to more than 300,000 companies and more than 200,000 million users in 188 countries. The company’s solutions operate in the cloud to safely manage the approvals, decisions, and contracts of people globally, without the need for them to exchange paper documents.

Taking paper-driven processes online is a key part of many organizations’ efforts to improve efficiency. For that to work, customers have to trust that their information is safe whenever they use DocuSign. “Our department’s mission is to ensure that the code we ship is secure, whether it’s code that we write ourselves or that we access from third-party libraries,” explains John Heasman, Senior Director of Software Security at DocuSign. “We engage development teams early, we go through design reviews, and we perform a security review before that code is deployed. Security is paramount. It’s at the heart of everything we do.”

Heasman’s team is responsible for threat modeling, code review, and automating a variety of approaches to identify security bugs—including dynamic analysis, static analysis, and fuzzing. ‘Fuzzing’ (or fuzz testing) is a technique for finding errors and security vulnerabilities in software code and networks. A fuzzer supplies the system with malformed data and monitors its behavior. As part of the company’s ongoing mission to improve security testing, Heasman’s team evaluated Microsoft Security Risk Detection—a fuzz testing service—to help ensure that DocuSign code is ready to handle today’s evolving security threats. “DocuSign and Microsoft have worked together in several areas in the past,” says Heasman. “When our counterparts at Microsoft asked if we’d be interested in trying out Security Risk Detection, we were really excited about it. We decided it would be an excellent opportunity to test some of our third-party document parsing libraries.”

**Automating testing to achieve maximum value**

DocuSign workflows begin with a customer uploading a file in a variety of formats, such as a PDF, a Microsoft Word document, or an image. “Our code has to operate on potentially untrusted customer data, and anyone who’s worked in application security knows that can lead to a variety of security issues,” Heasman says. The testing team configured Security Risk Detection to run several test harnesses (collections of software and data that monitor a program as it runs under various conditions). “We built the harnesses to run a set of common operations, such as image rasterization, and then we let the service run. But the key benefit is that we were able to automatically run millions of test cases across multiple virtual machines, entirely automated, with no extra work needed from us beyond the initial setup. We’re always looking for services that add value, and that scale of automation is a great added value.”

The team deliberately used multiple third-party libraries to compare test results from Microsoft Security Risk Detection. Says Heasman, "We had a wide-ranging sample set. We were able to identify that some libraries were indeed better than others. This confirmed our prior suspicions and evidence from our own internal testing, which gave us high confidence. The results were very positive."

Heasman notes that the ability to drop compiled binaries into Security Risk Detection and start testing them almost immediately is a particularly attractive feature. "That means we don’t need to instrument code in any special way, so we can focus our time on triaging the results, rather than having to build out a test infrastructure ourselves.”

—John Heasman, Senior Director, Software Security, DocuSign
on triaging the results, rather than having to build out a test infrastructure ourselves,” he says. “Whether the component was written by a one-person development shop or by an enterprise-level software company, my team has to evaluate whether that component is robust. This is a time-consuming process. Any way that we can automate that process while still getting a deep level of coverage is most welcome.”

Connecting the results with his point about added value, Heasman says that a major challenge with automated tools is the common trade-off between ease of use and the efficacy of the tool’s output. “The results will either be scarce—we don’t really discover anything—or there will be a huge number of false positives. Microsoft Security Risk Detection adds value because it doesn’t make us review a result set that’s largely meaningless.”

Taking advantage of outside security expertise

When designing their security testing regimes, companies like DocuSign have to decide whether to build a particular testing tool in-house or use one from a vendor. “I generally believe it’s better to develop security expertise in-house, but it’s also important to recognize scenarios where that’s not the smart thing to do. This is certainly one of those scenarios,” says Heasman. “We knew that the technology in Security Risk Detection had come out of many years of efforts from Microsoft Research around mathematical constraint solving. A mathematical formalism like that is not something a company like ours would invest its time in. It just made sense to take advantage of Microsoft expertise.”

Referring specifically to fuzz testing, Heasman adds that his team does have experience creating fuzz tools for PDF files. However, “Building new fuzzers for additional data formats becomes very expensive very quickly,” he says. “Going forward, we see Security Risk Detection as a cost-effective alternative. We gain fast validation on a particular library, parsing a particular format, without having to invest a lot of time in understanding the format to a level required to write our own fuzzer. And we’ll get more accurate results.”

Summarizing the evaluation of the Microsoft solution, Heasman says, “I would certainly recommend Security Risk Detection to other companies that are looking for cutting-edge, scalable, easy-to-use testing. These were the factors that impressed us the most. Having this type of technology available has been hugely beneficial to our team. We’ve been able to cover more testing ground with deeper coverage—it’s a great tool.”