RISE

Attracting and Developing Top Level IT and R&D CS/CE/Cyber Professionals

SAND2018-7731D

Software Security Testbed

Running Security Tools Against the OWASP Benchmark

Ashley Joy Paw, SUNY Polytechnic Institute

Project Mentors: Gary Huang, Org. 9371 and John McCloud, Org. 9366

Abstract:

We gathered metrics from a number of Static Application Security Testing (SAST) and Dynamic Application Security Testing (DAST) tools to quantify their effectiveness at detecting software security flaws. The OWASP Benchmark, which contains code with known security vulnerabilities, was used to provide ground truths for evaluation. Interim results are presented, with the goal of learning about and identifying security tool candidates for inclusion in software development processes and automated build pipelines.

Introduction:

Software made in the government sector has the highest prevalence of highly exploitable vulnerabilities, such as cross-site scripting and SQL injection, according to Veracode's 2017 State of Software Security report.

Sandia develops custom, high-consequence software for a wide range of customers. Using tools to automate finding security flaws in software is important in helping to protect Sandia's reputation and national security.

Procedure:

1. Set up a testing environment on an Ubuntu 18.08 virtual machine.
2. Set proxies to allow the pulling and building of the OWASP Benchmark on the VM.
3. Research commercial and open source security tools for analysis and gathered options.
4. Configure tools to run against the OWASP Benchmark
5. Record results and organize by logistical statistics (e.g. memory usage, time taken to run, etc.)

Results:

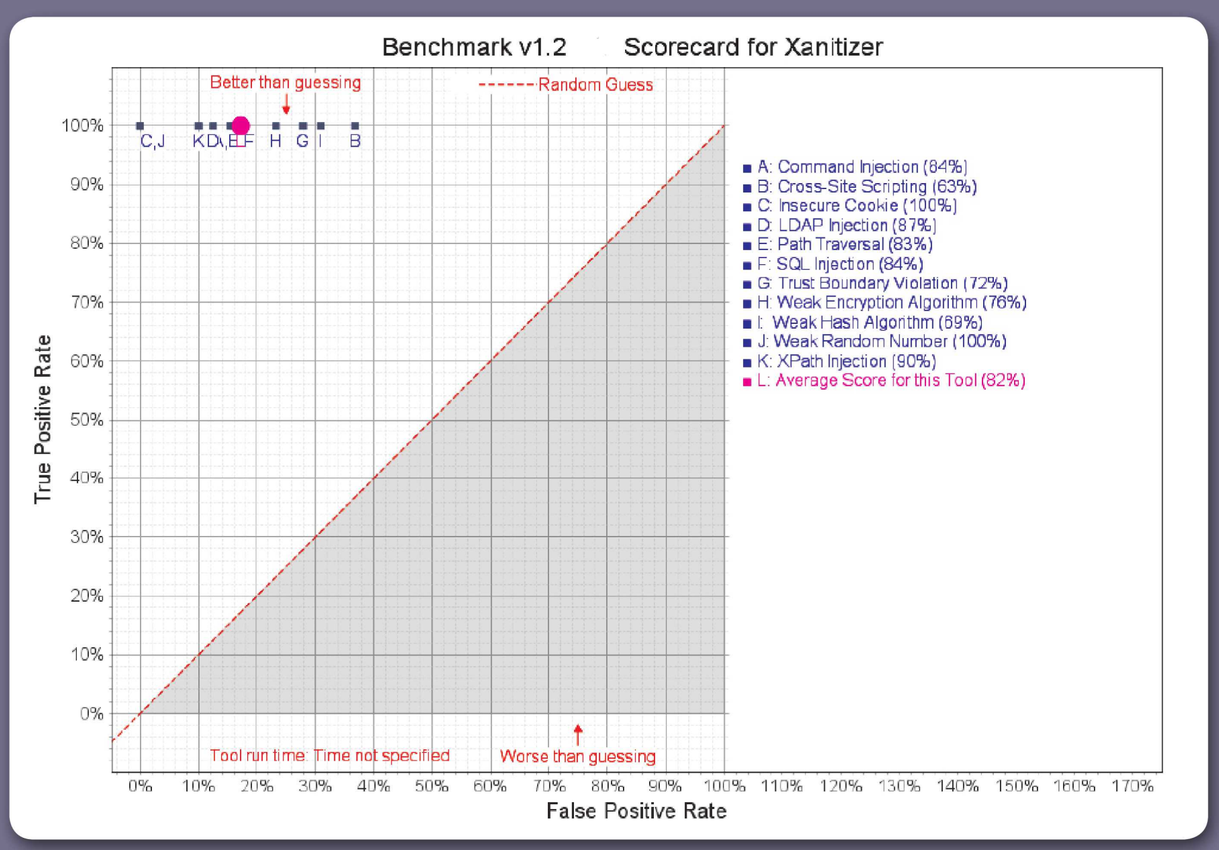

Figure 1: Report for Xanitizer

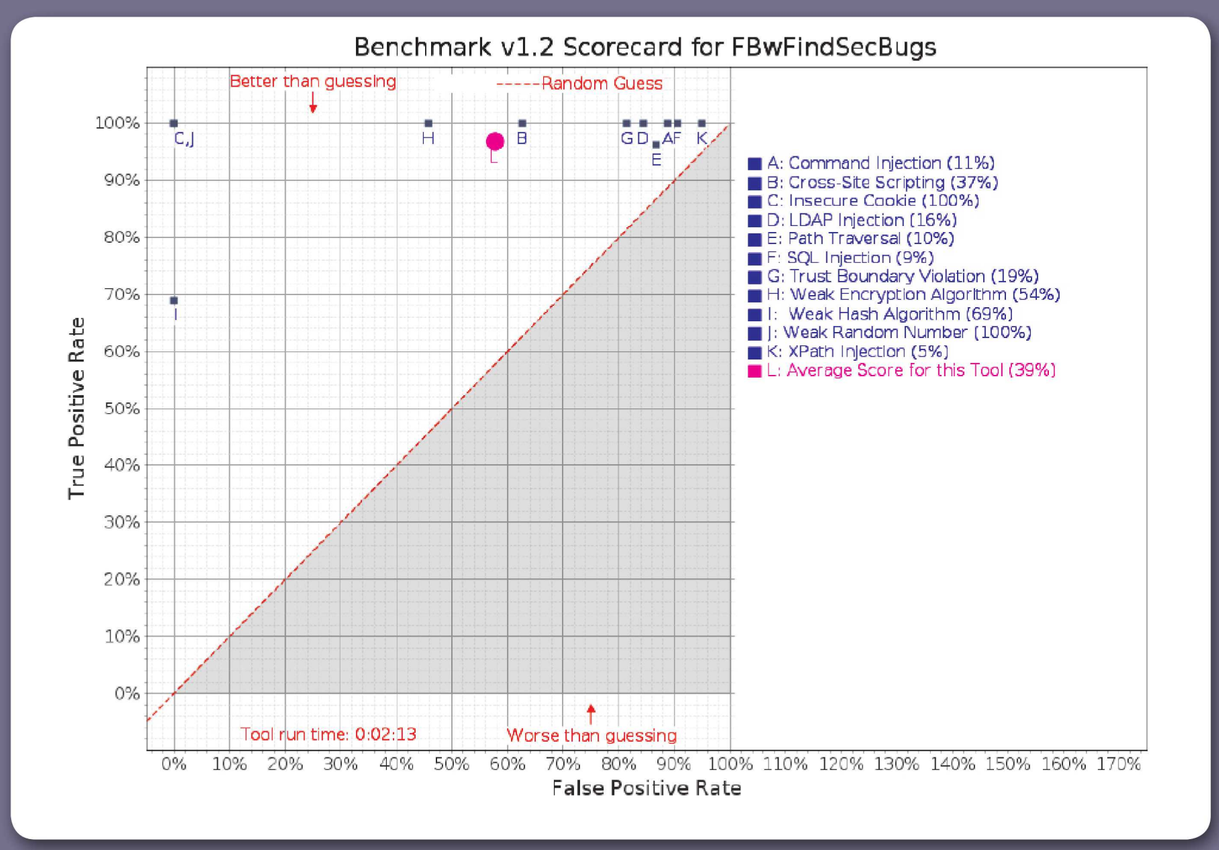

Figure 2: Report for FindSecBugs

Six different SAST tools were run against the OWASP Benchmark. Figure 1 shows the best score, and Figure 2 the runner-up. The vulnerabilities exploited are part of the OWASP Top 10 with an overall score rating the tool averaged.

Future Steps:

Gather metrics from resource-intensive DAST tools against the OWASP Benchmark on Carnac, integrate the best tools into CI/CD pipelines, and establish and maintain secure software development practices.