CASE STUDY: QWIET AI vs LEGACY

An anonymized look at a real life bake off at a Fortune 100 technology company

qwiet AI

TABLE OF CONTENTS

Overview

In a recent competitive bake-off, **Qwiet AI's preZero** platform was pitted against a legacy application security tool at a Fortune 100 technology company. The legacy competitor (referred to as "competitor" in this document) had been in place at the customer for multiple years prior to the bake-off. The bake-off consisted of **preZero** and the competitor scanning 10 production applications then comparing the results across multiple categories. The results from the bake-off clearly demonstrated Qwiet AI to have much shorter scan times, significantly more accurate results, better prioritization options, and dramatically fewer false positives, ultimately leading to almost 10,000 hours of improved productivity for the development organization. .

.

What is a “legacy” AppSec tool?

At Qwiet AI, we use the term “legacy” when referring to the previous generation of application security testing tools. There are a few identifiers that are shared across legacy application security testing tools. We’ll briefly cover these to establish a baseline understanding of the differences between modern tools (such as Qwiet AI’s preZero) and legacy tools.

1. Scanning Methodology

Quite possibly the biggest difference between legacy and contemporary tools such as preZero is the method used for scanning. Legacy tools utilize a “localized” scanning method that scans code in separate blocks without taking into account the data flow throughout the application. This method often suffers from very low accuracy, especially when it comes to reachability, due to lack of insight into the data flow in the application.

Qwiet AI uses a patented scanning method based on creating a Code Property Graph (CPG) of the target application. By combining an Abstract Syntax Tree, Control Flow Graph, and a Data Flow Graph, scans become more holistic, taking the entirety of the application into account, including the path data travels through the code. This provides a much more accurate detection of vulnerabilities and their reachability. (For more information on the CPG, please see an overview in this blog post or read the research paper behind the concept.)

2. Feature Adoption Methods

Legacy vendors often add functionality through acquisition. While it’s a very effective way to quickly add additional features, it often leads to performance or operational issues for customers. It’s not uncommon for legacy tools to require multiple scans to accomplish what modern tools do in a single scan. Modern AppSec tools offer many advantages, such as having SAST, SCA, Container, and Secrets detection all in one single scan. Modern tools also tend to have shorter ramp time to add new languages for scans.

3. Implementation of AI

With AI being a hot topic, everyone is looking for ways to implement AI in their toolset. Legacy vendors have been typically taking the approach of adding AI as a way to provide natural language queries for their scan results or to assist with processing results. Modern tools typically have AI integration that enhances speed and accuracy of results, expanding past what legacy AppSec tools are able to provide with traditional detection methods. (For more on Qwiet AI’s implementation, see this article.)

The Bake Off

The customer selected 10 applications across their organization to provide a wide variety of implementations, size, and languages to use for testing. Qwiet AI worked with the customer to get the product deployed, but no other efforts were made to tune the detection policies.

Step 1: Scan Times

With almost 5 million lines of code across 10 applications, there was a wide range of application sizes to provide a good understanding of how Qwiet AI would perform on a daily basis compared to the legacy AppSec tool. The first thing measured was the time it took for both Qwiet AI and the legacy solution to complete each scan. The chart below shows the application line count and the scan times in minutes per vendor per application.

Application	Lines of Code	Qwiet AI Scan TIme	Legacy Scan Time
1	8,950	1	1
2	1,103,924	2	108
3	170,179	3	45
4	11,404	1	Vendor report did not return scan time
5	24,829	1	Vendor report did not return scan time
6	2,176,490	1	23
7	805,272	5	65
8	32,396	4	45
9	531,640	10	90
10	171,066	2	6
Total	**4,968,150**	**28 minutes**	**383 minutes**
		4 minute average	**48 minute average**

Qwiet AI scan times were more than 10x faster than the legacy solution, even given the advantage that the legacy vendor had two scans that did not return times.

Step 2: Findings

The next step was to examine the total findings returned for each scan.

Application	Lines of Code	Qwiet AI Findings	Legacy Findings
1	8,950	23	3
2	1,103,924	64	487
3	170,179	57	210
4	11,404	39	39
5	24,829	50	61
6	2,176,490	9	146
7	805,272	147	204
8	32,396	70	350
9	531,640	43	1351
10	171,066	20	77
Total	**4,968,150**	**522**	**2,928**

Of course, the initial reaction to seeing Qwiet AI return around 1/5 the results found by the legacy solution was a bit concerning to the customer. Given the accuracy of our CPG based scanning method, Qwiet AI was confident in the findings. The customer then went through the results to determine the false positive rate and were pleasantly surprised with the results.

Step 3: False Positives

The next step was to examine the total findings returned for each scan to determine the validity of the results. As you can see in the chart below, **the legacy solution had a 79.88% false positive** rate!

Application	Lines of Code	Qwiet AI Findings	Legacy Findings
1	8,950	1	1
2	1,103,924	8	373
3	170,179	2	95
4	11,404	1	36
5	24,829	15	32
6	2,176,490	0	127
7	805,272	2	97
8	32,396	3	178
9	531,640	4	1338
10	171,066	6	62
Total	**4,968,150**	**28**	**2,339**
False Positive Percentage		**8.05%**	**79.88%**

While some false positives are to be expected when dealing with code, a false positive rate close to 80% goes well beyond tolerable levels. In the AppSec space, each finding needs to be researched and remediated. False positives dramatically increase the time an organization spends on their remediation efforts.

The customer informed us that it takes an average of four hours to research and fix each findings from their AppSec platform. With this sample set of 10 applications, results from the legacy vendor (including false positives) would take approximately 11,712 hours to research and remediate. Qwiet AI's findings would take approximately 2,088 hours to remediate, **saving the customer 9,624 hours** of developer time.

Conclusion

Application security testing requires a delicate balance of speed, accuracy, and prioritization. If a scan takes only a minute or two to complete, then scans are used more frequently and become part of the development process. If a scan takes an hour or more to complete, then they are used less frequently, often causing vulnerabilities to pile up, creating unforeseen and often substantial tech debt.

Accuracy is extremely important: each hour spent investigating a vulnerability is an hour not fixing bugs or developing features, because the people investigating a vulnerability are typically the same developers who are working to get the code out the door. False positives can also erode confidence in the findings sent over to developers from the AppSec team, leading to alert fatigue, and ultimately causing developers to mistrust and potentially ignore future scan results.

Accurate results also take time to remediate, so prioritization becomes extremely important the closer you get to releasing code. By properly prioritizing results, an organization can tackle vulnerabilities that present the biggest risk to security first, then relegate lower priority and unreachable vulnerabilities to be fixed in a later development cycle.

Qwiet AI's patented CPG-based scanning method allows for scans that are dramatically faster and much more accurate than other application security testing platforms on the market. Qwiet AI is helping organizations across many industries to quickly address the most pressing vulnerabilities in their code without a dramatic impact to their tech debt.

See how Qwiet AI can help your organization by taking it for a spin yourself or reaching out for a demo.