Application Security Testing Whitepaper

Evaluating the benefits and drawbacks of manual and automated vulnerability testing techniques

seisollc.com

About This Whitepaper

The Application Security Testing whitepaper is free for everyone to use, augment and add to. It has been written to create a basis for anyone who wants to learn more about the best ways to approach automated and manual tested methods from the perspective of both a security practitioner and executive, and for companies to measure the success of their security programs. Feedback and ideas are always welcome, you can contact us at: **info@seisollc.com**

Author

Eric Lansbery

Eric serves as the COO and Principal Security Engineer for Seiso, executing on and supporting projects related to web application penetration testing, application security, product and platform security, governance risk and compliance, and security tooling implementation.

Reviewers

Keith Holland
Joseph Wynn

Automated or Manual Testing?

While automated tools offer efficiency and scalability in scanning codebases for common vulnerabilities, manual testing provides a deeper understanding of application intricacies and uncovers nuanced flaws that automated tools may overlook. When both are used, testers obtain a more complete understanding of the weaknesses in the application being developed.

Topic Summary: Why Do You Need to Choose?

This whitepaper delves into the key differences between automated and manual security testing approaches within the context of penetration testing for custom-developed applications and the APIs they rely on. The differences between automated and manual testing methodologies significantly impact the depth and effectiveness of these security assessments. The insights provided herein are intended to help security practitioners and leaders alike understand these differences as they discern the pros and cons of each option and effectively implement a combined approach when assessing the security of custom-developed applications.

In summary, if you are a decision maker in the process of obtaining your next penetration testing partner, this whitepaper will provide the critical information you need to choose the right one and result in an auditable, professional, and complete penetration test.

After reading this whitepaper, you will be able to effectively:

- Begin developing a testing program that meets the business's risk management needs
- Effectively identify, prioritize, and plan for fixing critical vulnerabilities
- Improve testing efficiency and effectiveness, covering all bases in code review
- Reduce overhead in the tedious evaluation and remediation of vulnerabilities in code
- Obtain clarity on the methods challenging security leaders in making a solid decision when implementing testing methods or investing in the resources to perform the activities

Topic Presumption: Which Method Is Better?

While automated tools offer efficiency and scalability in scanning codebases for common vulnerabilities, manual testing provides a deeper understanding of application intricacies and uncovers nuanced flaws that automated tools may overlook. When both are used, testers obtain a more complete understanding of the weaknesses in the application being developed.

Topic Thesis: Avoid Putting Your Customers at Risk

As organizations increasingly rely on custom-developed applications to drive innovation, meet market demand, and ensure their own competitive advantage, the security of these applications becomes paramount. Further, the reliance on Application Programming Interfaces (APIs), and the

data they consume or process, has amplified the importance of accurately evaluating these components for effective protection mechanisms.

Penetration testing emerges as a critical method for assessing the security posture of custom applications, providing insights into vulnerabilities and weaknesses that malicious actors may exploit. These testing methods in combination with a well-established vulnerability and risk management program will build the capabilities needed to protect your application customer's data and instill the confidence they need to continue partnering with your business.

Topic Exploration: Automated Versus Manual Testing Explained

Automated Security Testing

Automated security testing leverages specialized tools and technologies to assess the security posture of custom-developed applications efficiently. These tools offer several advantages, including speed, scalability, and repeatability, making them invaluable assets for organizations seeking to identify common vulnerabilities in their applications. Common automated security testing techniques encompass static analysis, dynamic analysis, and fuzz testing.

- Static analysis involves examining the application's source code or binary without executing it while still being able to identify potential security vulnerabilities such as code injection, insecure dependencies, or hard coded credentials.
- Dynamic analysis, on the other hand, involves executing the application and observing its behavior in real-time to identify vulnerabilities that may arise during runtime, such as input validation errors, insecure configurations, or authentication bypasses.
- Fuzz testing, also known as fuzzing, involves feeding the application with invalid, unexpected, or random inputs to uncover potential vulnerabilities.

While automated security testing offers notable benefits, it also presents challenges in detecting complex vulnerabilities and distinguishing between genuine threats and false positives or negatives. For example:

- Automated tools may struggle to detect logic flaws, business logic vulnerabilities, or subtle misconfigurations that require human expertise and contextual understanding to identify accurately.
- Automated testing tools may produce false positives, indicating vulnerabilities that do not pose a real risk, or false negatives, failing to detect genuine vulnerabilities, thereby necessitating manual validation and verification.

Thus, while automated security testing provides a valuable foundation for security assessments, organizations must complement it with manual testing approaches to achieve a comprehensive understanding of their application's security posture.

Manual Security Testing

Manual security testing plays a crucial role in complementing automated approaches by providing human expertise and intuition to uncover nuanced security flaws and attack vectors in custom-developed applications. Unlike automated tools, which rely on predefined algorithms and patterns, manual testers possess the ability to think creatively and adapt their strategies to the unique characteristics of each application.

- Manual testing involves exploring the application's functionalities, interfaces, and underlying architecture in depth, allowing testers to identify subtle vulnerabilities that automated tools may overlook.

- Human testers leverage their expertise to delve into the application's logic, identifying logical flaws, business logic vulnerabilities, and misconfigurations that automated scans may not detect. These vulnerabilities often stem from complex or unconventional usage scenarios that require human intuition to uncover.

- Manual testers can tailor testing scenarios to simulate real-world attack scenarios specific to the application's functionalities and usage contexts.

- By mimicking sophisticated attackers and thinking like adversaries, manual testers can uncover vulnerabilities that automated tools may miss, such as privilege escalation, session management flaws, or data leakage vulnerabilities.

- A customized testing approach ensures that the security assessment is aligned with the organization's unique risk profile and provides actionable insights for improving the application's security posture.

Manual security testing serves as a critical component of comprehensive security assessments, enabling organizations to identify and mitigate vulnerabilities effectively and safeguard against potential cyber threats.

It takes many years to become a skilled manual tester and most who are skilled need to constantly keep up with their ongoing development and reflexes. An additional factor to consider regarding manual testing is the time and skillset required to complete a thorough examination of an application's structure to manual testing alone. Without the balance of automated and manual testing activities, the manual testing can still fall to the gaps of the human element.

With that being said, often times the public consensus is that manual testing is very error prone and could lead to instability in the application if the tester is overzealous with their methodology. This is a main concern of organizations who are seeking the benefits of penetration testing primarily, while both automated and manual testing should also be deployed within the software development lifecycle prior to any code reaching production. The added emphasis of testing early and aligning the skillsets of testers to the environment help to reduce this risk and further justifies the need to combine both methods in a well-defined vulnerability management practice.

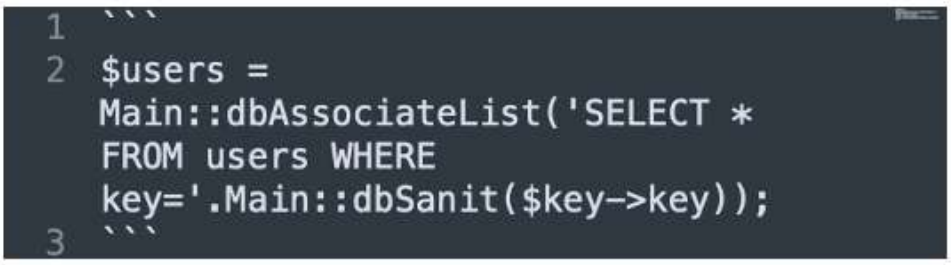
Depth of Analysis Comparison:

When comparing the depth of analysis achieved through automated and manual testing approaches, it becomes apparent that each method offers distinct advantages and limitations. Automated testing tools excel in scanning large codebases efficiently and identifying common vulnerabilities across a broad spectrum of applications.

These tools employ predefined algorithms and heuristics to analyze code and detect known vulnerabilities, providing organizations with a rapid assessment of their security posture. However, while automated tools are proficient at identifying well-documented vulnerabilities, they may struggle to uncover complex interactions, edge cases, and potential attack paths that require human intuition and contextual understanding to discern.

Technical Jargon Ahead

Here's an example line of code from a common automated scanner that was used in a real-world environment, that triggers a high severity finding.



The finding is listed as **php.lang.security.injection.tainted-sql-string** in the SAST scanner, which could allow an attacker to use a SQL injection attack to steal or modify contents of the database. What the scanner does not know is that a customized sanitization function was also implemented in this example finding that prevents the problematic syntax from being exploited. This additional knowledge would lead a manual tester to lower the severity level based on true risk to the organization.

Manual testers possess the capability to explore these nuances in depth, leveraging their expertise to identify subtle vulnerabilities that automated tools may overlook. By immersing themselves in the application's functionalities, manual testers can uncover hidden flaws, logic errors, and unconventional attack vectors that automated scans may miss.

This human-driven approach to testing provides organizations with a comprehensive understanding of their application's security posture, empowering them to address vulnerabilities effectively and mitigate potential risks.

Contextual Understanding:

Manual testers bring a depth of contextual understanding to security testing that automated tools alone cannot match. They possess a comprehensive grasp of the application's business logic, user workflows, and data flows, allowing them to identify security risks that are specific to the application's architecture, design, and functionality.

By immersing themselves in the intricacies of the application, manual testers gain insights into how different components interact, how data is processed and stored, and how users interact with the system such as integration points with third-party systems or external APIs. They can analyze the application's architecture and design to pinpoint potential weak spots and attack surfaces that could be exploited by malicious actors. Additionally, manual testers can simulate user behaviors and usage scenarios to identify security risks that may arise from unconventional usage patterns or edge cases.

This intimate knowledge enables testers to uncover vulnerabilities that are unique to the application's environment, such as misconfigurations, access control issues, or data leakage points that automated tools may overlook.

However, achieving a similar level of contextual understanding through automated testing alone poses significant challenges. Automated tools lack the human intuition and contextual understanding required to identify subtle vulnerabilities or understand the nuances of the application's behavior. While automated scans can identify known vulnerabilities and perform basic checks against predefined criteria, they may struggle to assess the application's security posture in the context of its unique environment

Real-World Attack Simulation:

Simulating real-world attack scenarios and threat actor behaviors is paramount in evaluating the resilience of custom-developed applications against sophisticated cyber threats. Manual testing offers a unique advantage in this regard, as testers can mimic the behaviors and tactics of real attackers, adapting their strategies based on evolving security measures.

Unlike automated tools, which rely on predefined patterns and algorithms, manual testers possess the flexibility and intuition to explore the application's vulnerabilities from the perspective of a determined adversary. By leveraging their expertise and understanding of common attack methodologies, manual testers can craft customized attack scenarios tailored to the specific characteristics of the application and its environment.

This human-driven approach enables testers to uncover vulnerabilities that automated tools may miss, such as logical flaws, privilege escalation paths, or data leakage vectors.

Manual testing can adapt their strategies in real-time, responding to changes in the application's defenses and evolving threat landscapes. In contrast, automated tools are limited in their ability to emulate human-driven attacks and identify compounding vulnerabilities or adapt to dynamic environments.

While automated tools excel in scanning for known vulnerabilities and performing repetitive tasks efficiently, they fall short in simulating complex attack scenarios and understanding the nuances of the application's behavior. Therefore, manual testing remains indispensable in providing organizations with insights into their application's security posture and resilience against real-world threats.

Topic Conclusion: Which Solution is Right for You?

The evidence is clear that both automated and manual security testing play crucial roles in penetration testing for custom-developed applications, each offering distinct advantages and limitations. Automated testing provides efficiency and scalability in scanning for common vulnerabilities, while manual testing offers depth of analysis and contextual understanding. The key differences between these approaches lie in their ability to identify nuanced vulnerabilities, simulate real-world attack scenarios, and adapt to evolving security measures.

While automated testing offers efficiency and scalability for routine validations, manual testing provides a deeper understanding of application intricacies and uncovers nuanced vulnerabilities that automated tools may overlook.

From the perspective of a pentester and a security leader, take great caution when hiring 3rd party testers that only rely on automated tooling. You will find that the depth of the findings and recommendations lack enough detail to truly protect your environment.

It is important to note that not all manual or automated testing solutions available are created equal in effectiveness, depth, or overall analysis of an application. The results of the testing methods are highly dependent on not only the skillsets of the testers themselves, or the development teams creating automated solutions, but the understanding of how businesses identify, categorize, analyze, and remediate vulnerabilities in code based on the true exploitability of the issue.

A well-organized risk management and data privacy practice that enables these testing methods will always be a good starting point for any company to effectively prioritize the mitigation steps required to truly protect their application, their business, and their customers.

Seiso's Penetration Testing Solutions

Web Application Penetration Testing

Seiso ties traditional testing methods with the modern application stacks and provides an in-depth, customer interaction-based engagement that includes code reviews, client-side, and business logic testing. Reporting is designed to support both compliance and internal best-practice requirements, while providing in-depth remediation opportunities and re-testing steps.

Business Risk Management Focused Testing

Seiso prioritizes understanding your critical business processes to focus on preventing security compromises. We employ a risk-based approach to inform our testing methods, ensuring effectiveness. Our certified testers are highly skilled in manual and automated testing, risk management, framework alignment, vulnerability management, wireless testing, and social engineering. Our testing produces tailored, auditable reports compliant with SOC 2, ISO 27001, NIST, CMMC, and other requirements. We provide concise finding summaries and remediation recommendations that seamlessly integrate into your risk and task management solutions.

Enterprise Penetration Testing

Seiso specializes in proactive security assessments, where our authorized cybersecurity professionals simulate real-world attacks. We pinpoint vulnerabilities in IT systems, networks, wireless access, and personnel, empowering organizations to fortify their defenses and preempt potential risks.

Cloud Security Penetration Testing

Seiso's pentesters are experts in all aspects of cloud operations and security. With skills in software development, cloud security engineering, and security assessment, we employ top-tier methods, tools, and frameworks to thoroughly evaluate your cloud environment. Our goal is to identify vulnerabilities early and facilitate easy remediation.

Blogs

[Penetration Testing Differentiators]

Seiso by the Numbers

95%

Customer Retention

100%

Client Certification Success

100%

Focused on making security your advantage