SANS

A SANS Whitepaper

Measuring Cybersecurity Controls Effectiveness with Security Validation

Written by **John Hubbard**

December 2020

Sponsored by:

Mandiant

Introduction: The Problem

As anyone reading this paper likely knows, protecting an organization from advanced attacks is no easy task. Year after year, breaches continue to occur with seemingly equal or larger impact than the prior year. Yet every year, we hear about claimed improvements in security controls. What gives? Are we still not spending enough money on cybersecurity? Are the tools not working? Is it our cybersecurity teams' fault? While the answer to this likely lies somewhere in the middle of all these questions and depends on each individual organization's security strategy, some numbers are clear, and the picture isn't exactly pretty. According to Ernst and Young's 2020 paper titled "How does security evolve from bolted on to built-in?"[1]:

- 20% of organizations are extremely confident that the cybersecurity risks and mitigation measures presented to them can protect the organization from major cyberattacks.
- 25% of organizations can quantify in financial terms the effectiveness of their cyber spend.
- 26% of breaches in the past 12 months were detected by the security operations center (SOC).

[1] "How does security evolve from bolted on to built-in?" 2020, https://assets.ey.com/content/dam/ey-sites/ey-com/en_gl/topics/advisory/ey-global-information-security-survey-2020-single-pages.pdf

What do these numbers mean? First, it looks like the average SOC has some serious catch-up work to do, and two, cyber defense teams may be struggling to communicate the volume and effectiveness of their work in keeping their organizations safe.

According to IBM's 2020 report titled "Cost of a Data Breach Report,"[2] data breaches are an incredibly expensive problem. The global average cost of a data breach in 2020 is $3.86 million, with the US having the highest country average cost of $8.64 million and the healthcare industry having the highest industry average cost at $7.13 million.

Given these numbers, are we doomed to fail? Certainly not. But what makes the difference between successful cybersecurity teams and those that routinely miss attacks? One answer to this question is what we will explore in this paper: validation. One of the issues that weighs heavily on every SOC analyst and manager is "How do I know my tools will actually work when a real attack occurs?" The best way we can answer this question is through thorough validation of security controls.

Not all security validation options are created equal. A test is only as good as it is current and representative of the real world. Validation tests performed weeks or months ago may be highly useful at the time but living in the current world of DevOps, digital transformation and cloud migration, your security stance and environmental details can change wildly at any moment. This could leave your data exposed and you unaware because you're basing your knowledge and assumptions on invalid tests.

This paper will explore best practices for getting in front of these issues by measuring cybersecurity control effectiveness. It will explore the field of security validation technologies, what they can do, how they came to be and the key capabilities to consider when choosing a security validation strategy. A solid security validation strategy will help you sleep better at night knowing you've done the best possible job to verify your security stance. It also will help you avoid the much more unfortunate and uncomfortable version of the "How do I know my tools work?" or your organization's leadership asking "How did we get breached? I thought we spent all this money on security tools that were supposed to prevent this!"

A SOC must implement real, meaningful protection, but also clearly communicate the protection plan and how it is a positive return on investment for the organization. It's more important than ever that cyber defenders have tools that help objectively measure defenses. If you are struggling with these problems, continue reading for key information and best practice that can help you overcome these all issues.

[2] "Cost of a Data Breach Report 2020," www.ibm.com/security/digital-assets/cost-data-breach-report/#/

Verification: Past and Present

Throughout the history of cybersecurity, attempts to verify security controls have often consisted of multiple well-meaning, but potentially flawed methods of assessment. Assessment tactics include a combination of periodic penetration tests, vulnerability testing and metrics collection, combined with ad-hoc, single-point functionality checks, or, at worst, burying our head in the sand and hoping our tools work as promised when attackers come knocking.

These methods, however, can be unreliable, which leads to predictable issues. Time after time, we've heard comments such as "That worked before, I swear!" and "I thought tool X should have identified this. What happened?" because security teams are not aware of current solutions that may be more effective. Before jumping into the current solutions, it's important to understand the history of testing to appreciate the current situation.

Traditional Methods of Verification

According to Matt Bromiley's SANS spotlight paper, "What Security Practitioners Really Do When It Comes to Security Testing,"[3] the most used methods for testing effectiveness of security controls are penetration testing, red teaming, simulating attacks with homegrown malware simulations and simulating attacks on a clone of the organization's IT setup (see Figure 1).

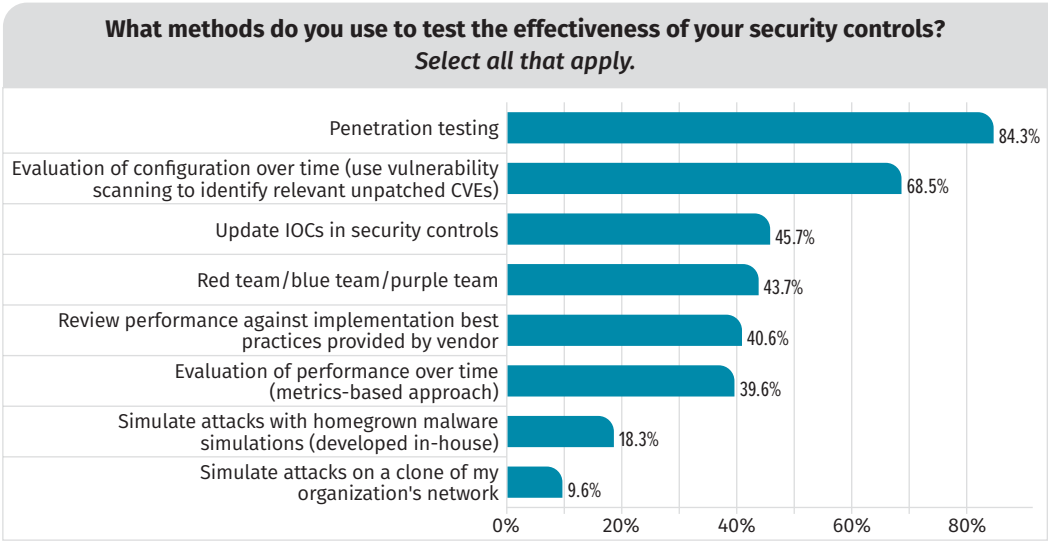

Figure 1. Methods Used to Test Security Control Effectiveness

While these methods are a great start, they are all largely manual approaches, which brings a set of potential issues.

Manual Testing

In the past, security teams looking to verify their security posture outside of penetration testing were forced to approach validation in manual and time-consuming ways. The basic tests consisted of a SOC analyst verifying that when a simulated attack was attempted, the activity was logged and alerted as expected. Of course, this approach comes with several problems.

One problem with a one-off test is that it's limited to verification of that moment in time. What if things change? The average organization's network tools and software are in a constant state of flux, so an analytic strategy that worked one moment may

[3] "What Security Practitioners Really Do When It Comes to Security Testing," October 2019, www.sans.org/reading-room/whitepapers/analyst/security-practitioners-security-testing-39210 [Registration required.]

be affected by a change upstream that has unanticipated effects, leading it to fail to function a moment later. Additionally, log agents and collection software can malfunction, creating unexpected blind spots. Considering these scenarios with the large number of analytics a SOC must implement, you can see how the individual testing approach does not easily scale.

The other big problem is that defenders are often not trained in penetration testing and attack methods and, therefore, struggle to accurately simulate attacks. While many open source attack tools and method lists exist, a cyber defender is generally not expected to know the myriad options for command and control beacons, persistence methods and other post-exploitation tactics, or be able to properly implement them. Collecting various types of command and control beacons, malware and the array of other sources to test can be a full-time job on its own and far too much for the average defender to conduct. Therefore, the threat intelligence required to do a solid test may exclude many teams from even approaching the task in the first place. Consequently, even if the environment is operating as expected, a defender attempting to test things on his or her own may fall far short of what is needed to provide a realistic and representative test.

Security Testing Automation Arrives

To combat these issues, defenders devised solutions. First, defenders individually scripted and automated systems to create scalable testing inside their organizations. While this led to increased scalability, it did not help solve the problem for the teams that lacked coders to write custom automation scripts. Eventually, open source solutions became available which allowed defenders to run these types of tests on their own based on standardized sources of threat intelligence, such as Mitre's ATT&CK™ Matrix. While this started to bring repeatable, automatable testing within reach for many organizations without coders, it still took effort to set up and customize for a given environment, including the downside of unpredictable updates and a lack of official support. Even today, testing often falls by the wayside as a "if we have time" activity. This is likely because many organizations would have to spend an inordinate amount of time to implement and demonstrate a clear return on the investment. Fortunately, as time progressed, additional solutions became available that addressed these issues.

Penetration Testing and Red Teaming

For many years, organizations used the penetration test, red team assessment and other unannounced adversary emulation activities to test defenses in a realistic environment. These tests, while often excellent in terms of coverage and techniques used (assuming a well-trained attack team), have down sides.

One obvious issue related to penetration testing is cost. Employing a team of professionals to plan, execute and report their attack on your organization is inherently a costly activity. While they do a great job at showing what a realistic attacker can achieve, it is not something the average organization is going to run on a continuous basis, maybe not even once a year unless compelled by compliance requirements. This means that

these tests are "point in time" assessments that often don't age well over the months. They also fall prey to some of the same environmental drift and configuration change problems that self-testing has. Bromiley's paper (Figure 2) reveals the issue of frequency quite clearly.

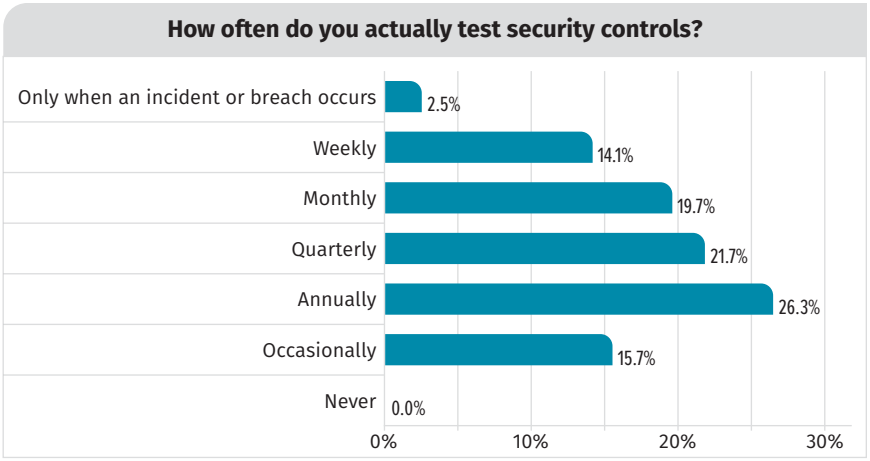

Figure 2. Reported Frequency of Security Tool Testing

The second issue with penetration testing is—let's be frank—they almost *always* succeed. Go ahead and ask a penetration tester how many times he or she has failed to get into an organization and you'll likely receive the answer "Never!" Why is this the case? One reason is that most SOCs fail to complete the self-run analytic verification steps, leading to a complex, expensive test that ultimately has a predictable result: complete bypass of defenses and completion of whatever nefarious mission the attacker set out to simulate. That means not only are penetration tests a single-point assessment, they also are not appropriate because the team clearly wasn't ready!

What if you could get rid of the issues associated with self-run analytic verification while at the same time improving the value gained from unannounced, full-scope penetration tests/red team assessments? New tools in the security validation space can deliver exactly that. Because of their ease of use and improved value, this security product category is gaining momentum.

The Modern Solution

When looking at the barriers to assessing control effectiveness, Bromiley's paper indicates (see Figure 3) that the issues discussed in previous sections affect half of the organizations surveyed.

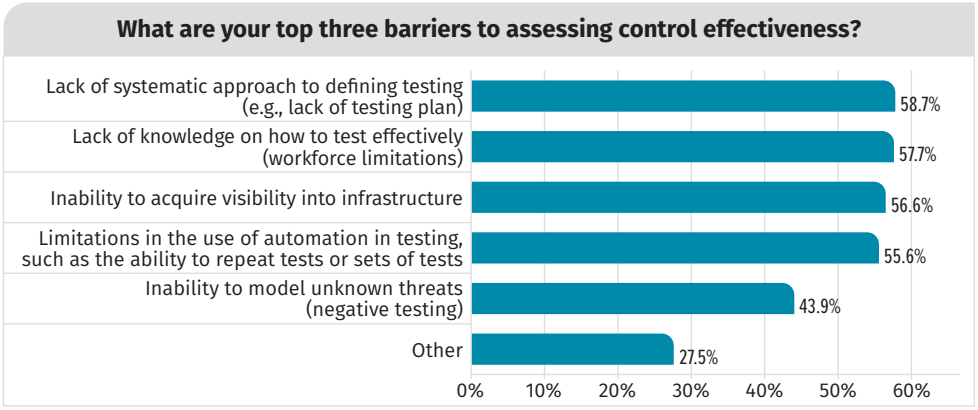

Figure 3. Reported Barriers to Assessing Security Control Effectiveness

Those barriers are:

- **A lack of a systematic approach—**Without a systematic, repeatable process behind testing, the whole operation can feel untrustworthy and undermine the credibility or effectiveness.
- **Lack of knowledge—**A solution set to solve these problems should remove the need for expertise in attack techniques and put security validation within the reach of all teams, no matter how small or inexperienced.
- **Inability to acquire visibility into infrastructure—**A security validation solution likely cannot address this, but can reveal where key data sources (such as network traffic capture and endpoint security events) required to detect an attack are either wholly unavailable, have gone offline unexpectedly, or were misconfigured, broken or otherwise out of service.

- **Limitations in the use of automation in testing—**A security validation solution should make it easy to repeat single tests and entire sets of tests on a continuous basis.
- **Inability to model unknown threats—**Many teams struggle to implement timely and relevant threat intelligence for detection purposes alone. Adding the desire to not only use the information but to craft attacks based on it can be even more difficult, especially when many adversary tactics, techniques and procedures remain difficult to define.

With these issues in mind, let's dive into continuous security validation and show how this product category takes a modern approach to solving the problems of accessibility, repeatability and visibility.

Continuous Security Validation

The need for easier security validation brought the advent of vendor-built and -supported continuous security validation solutions. This new wave of easily approachable and automatable options lowered the bar for security validation testing to increase accessibility for teams of all sizes.

These solutions often consist of deployable agents or dedicated nodes for initiating tests and telemetry health monitoring. These nodes, combined with the collected event and alert data, paint a picture of the health and capabilities of security controls across people, process and technology. By having the defense team instruct nodes both inside and outside the network to communicate with one another, the nodes can create traffic and endpoint activity that closely imitates a real attack. Then, under the assumption that malicious activity is representative of real attack techniques—and those attacks are seen by security infrastructure—an organization can feel assured that their security controls will fire in a true attack scenario.

Using automated continuous security validation means much of the pain, unreliability and inaccuracy of manual testing can be remedied while bringing the following benefits:

- Facilitating the rapid deployment of validation capability with vendor-backed support
- Reducing the complexity of testing to be within reach of teams of any size
- Enabling blue team members (who are unlikely to be trained in state-of-the-art attack techniques) to run complex attack scenarios
- Outsourcing the difficult task of simulating a realistic attack to vendors who specialize in threat intelligence
- Increasing confidence in the overall posture of cyber defense and the health and configuration of key data sources, an important prerequisite to detecting attacks
- Mapping and reporting validation output to industry standard attack models, which enables benchmarking internally over time as well as to other organizations
- Providing threat intelligence to guide security teams toward the most important attack types to test
- Helping teams identify which security controls truly matter, and which are not returning value

Continuous Is the Key

As previously discussed, one of the negative aspects of previous testing solutions was their manual, single-point-in-time nature. It cannot be overstated that perhaps the biggest improvement that continuous security validation tools bring is the ability to continuously assess posture. This eliminates the point-in-time assessment problem. Instead, tests can be run in a scheduled, automated way, informing teams the moment something has changed or broken. Whether that problem is a data source that has become unavailable, an accidentally disabled alert rule or analytic logic that has been modified in a way that had unexpected consequences, failure to spot an attack will immediately raise an alarm. Once teams are aware of the problem, they not only can fix the issue and return to normal, they also can deconstruct the failure and improve resiliency in the entire system going forward.

Keys to Building an Effective Security Validation Program

The benefits of continuous security validation are clear. A separate, yet related need is picking the best solution. The following information outlines the things to consider when selecting a solution for continuous security validation.

Complete and Trustworthy Test Results

One of the most important factors in building a security validation program is trusting it will deliver accurate and complete results. Since there are many possible interpretations for complete results, it is important to understand what exactly "complete" means when it comes to security validation.

Data Feed Availability and Health Measurement

One way to look at "complete and trustworthy" is how it pertains to security data feed (such as endpoint logs and network traffic) availability and health. While it may not be the first thing that jumps to mind for security testing, collection and centralization of that telemetry is a necessary precondition for success. After the initial configuration, continued checking of the health of that data feed is required. Without the assurance that event and alert data is flowing as expected, all other portions of the validation attempt could be called into question. Therefore, a good first step toward success is identifying a solution that can inform you if any of your sources of data are missing in action, misconfigured or even performing suboptimally, and continuously watch for any problems that may occur. Attackers may purposely cut off logging and log agents can spontaneously crash, either of which can make it difficult to identify a blind spot or a potential incident.

Full Attack Cycle Validation

Testing the full range of pre- and post-exploitation attack tactics, while utilizing multiple techniques for each, is another way to define complete security validation. The Mitre ATT&CK Matrix, for example, shows that there are numerous ways of accomplishing each high-level, post-exploitation tactic. With the Mitre ATT&CK July 2020[4] sub-technique updates (see Figure 4), even a single technique is broken into multiple test cases to be considered.

When executing attacks, the real attack binaries should be used, when available, to ensure the highest level of attack authenticity.

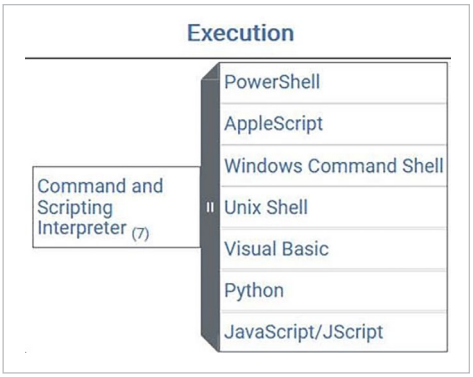

Figure 4. Multiple Methods for Achieving Execution Through Command and Scripting Interpreters

Attacker Tools and Protocol-Based Tests

Tactics and techniques are just one category to consider when looking at complete testing. Another factor to consider are specific items. For example, the malware that attackers use, the protocols those pieces of malware implement, and the network and host-based evidence they leave behind are all different and specific ways to look at complete testing. The better a product can deliver high-level tactics representative of techniques or emulate the tools a threat group may use, the more realistic validation test can be run. For example, knowing if an adversary is known to use WastedLocker ransomware, PlugX as a back door, CARROTBALL downloader or RIG exploit kit to attack browsers—and being able to test against these specific tools—can be even more validating to your testing. Of course, it's not possible to replicate all advanced attacker malware as much of it remains unavailable and closed source (unless your security validation solution vendor can provide it. But, where possible, creation of traffic and activities that mimic these tools is a major plus.

Endpoint and Network Appliance Coverage

Finally, complete security validation should include detecting an attack from both network and host-based tools regardless of the origin and nature of the attack. Tools for validation should take in data from both types of data sources and identify the totality of the attack from both the network and host viewpoint. They should be capable of providing data on whether an attack was spotted by network devices such as firewalls, IDS, network service logs, PCAP and more, as well as endpoint tools such as AV, EDR, host firewall logs, and system and authentication logs.

Threat Intelligence-Backed Testing

The first requirement of intelligence-backed testing is to have in-depth threat intelligence on the wide swath of attackers that are active at any given moment. Since this type of threat intelligence is not something you can produce overnight, working with a vendor that has been present in the incident response space will be important when buying a security validation product. A long history of researching, dealing with incidents, and reverse engineering advanced malware and tools is a good sign that a vendor will be able to provide accurate guidance on the tests that need to be run.

[4] Mitre ATT&CK, July 2020 Updates, https://attack.mitre.org/resources/updates/updates-july-2020/

Organizations looking to quickly start up a security validation program will want swift and relevant intelligence starting from broad targeting statements in the form of "We are a company in regions or industries A, B and C that produces/stores data on X and Y." Being able to correlate what your organization does with past cyberattacks based on geography and incidents at similar organizations, can help quickly orient you to the battlefield and predict your most likely enemies. Ask your vendor where they get threat intelligence. Are they doing research and gathering it from other sources, or do they have an incident response capability that also can produce intelligence from nonpublicized incidents? Do they have a dedicated threat intelligence team producing finished intelligence products giving them much wider than average visibility into the cyber landscape? In this realm, the further the reach, the better the likelihood they can match your organization to the relevant threat actors and the attack techniques they use.

The other requirement of intelligence-backed testing is assisting the organization in focused testing on the right internal assets with specificity. For an attack group to be considered a valid threat, they must possess the *capability*, *opportunity* and *intent* to cause harm to an organization. If any of these factors are missing, that group—as worthy of an adversary as they may be—is not necessarily of high concern. Your vendor should help you focus testing on the highest risk and most important assets your company has. Doing so relies on identifying which groups exist and what their intents and capabilities are, then matching them with what is present in *your* environment. If, for example, an organization has a highly valuable business or manufacturing process, but there are no known adversaries interested in it, then there is no intent. So focused testing around protecting it may not be the best use of time. Conversely, if an organization's threat model for a particular asset only includes attackers with a low skill level, risk of breach for that asset may be extremely low as well. Meaning efforts are best spent elsewhere, where attacker capability is known to exist. Therefore, continuous security validation aimed at providing the most value for the time and money should provide an asset- or data-centric view of what may be of value to an attacker. With this knowledge, organizations can focus testing on areas where there is known interest from adversaries with *intent* to compromise that data or asset type.

The ability to provide this information centers around the wealth of threat intelligence a vendor has collected, makes available to its customers and, ideally, integrates directly within their platform. This threat intelligence should include both high- and low-level detail about attackers and the context around their attacks so that customers can make informed decisions about their true highest risks.

Relevant, Flexible and Actionable Testing

Finally, security validation solutions should have the customer-relevant data consistently available for testing in flexible ways. The results of these tests also should be presented in a clear and actionable way to your security team and management.

Up-to-Date Methods

Up-to-date methods requirement could just as easily go under threat intelligence, if not kept relevant. A continuous security validation solution needs to be backed by a vendor that can identify and push timely updates for the attack flavor of the month. As the cat-and-mouse game between defenders and attacks rages on, assumptions for tools and techniques can quickly become outdated. Ask your vendor how they acquire their threat intelligence and how quickly you can expect an attack you read about in the news to appear as a validation test they can run. If newly released exploits are any indication, attackers that find out about new attack methods may begin to use them in less than 24 hours. The ability to keep a high update tempo can become a make-or-break strategy in the fight to identify new attacks.

Multi-Environment

Another key capability is facilitating testing from any given endpoint or device in one subnet to any other device in another. In other words, a security validation solution should be able to execute attacks in all possible permutations of sources and destinations throughout the organization's network. This is an absolute necessity because lateral movement is a key attack phase in nearly all advanced/targeted attacks. With this being the case, the source and destination subnet of any given attack stage is not only unpredictable, but also likely to happen in multiple combinations, *all* of which need to be visible for defenders. A complete solution for security validation should allow defenders to launch an attack to and from inside the network, outside the network, the cloud, VPN links or any other network zone that may be in play.

Actionable Tests

Last but not least is the quality of the output from the validation tests. Here are some questions to consider about the output reports that are produced:

- Does the vendor make it clear what is working and what is not, as well as recommend how to fix it?
- Can the vendor inform you of how to prioritize testing against the Mitre ATT&CK Matrix, NIST Cybersecurity Framework or other frameworks?
- Are the actionable items prioritized in a way that makes it clear which issues need to be addressed first and why? Can these items be tied to defined business outcomes?
- Can longer-term trends be drawn from the data to show, for example, that the SOC is improving in ability to detect attacks reliably over time?

- Do the reports align with attack frameworks such as the Lockheed Martin Cyber Kill Chain® or the Mitre ATT&CK Matrix, and do they show where there are potential coverage gaps or inadequate coverage compared to other phases?
- Can the vendor produce threat intelligence informing you of how to prioritize testing based on what is most relevant to your organization?

Questions like these will help ensure that a security validation tool and strategy can effectively be turned into action for fixing issues and clear communication about the status of security operations at any given time.

Summary

Security teams are always looking to gain an edge against attackers. Although buying better prevention and detection technologies is one way to do that, those purchases typically add a difficult-to-quantify improvement to security posture. In the worst case, they may actually add no additional coverage beyond what is already implemented. How are security teams to know how well they are covered, and which tools are the most effective for their investment? Continuous testing of tool functionality and capability brings assurances that you will spot attackers when they come knocking. It also can help focus your cybersecurity spend on the tools and technologies that can be objectively demonstrated to produce the largest return on investment. Of course, no testing can predict every possible future attacker action. However, using the combination of the best available threat intelligence that pulls from global adversary, machine, breach intelligence sources and expertise, with inward-facing knowledge of sensitive data, infrastructure and vulnerabilities, cyber defense teams can significantly improve confidence in their ability to deliver on their mission; thus, minimizing business impact of attacker actions and ultimately preventing costly breaches.

About the Author

John Hubbard is a certified SANS instructor who authored the new SEC450: Blue Team Fundamentals: Security Operations and Analysis and co-authored SEC455: SIEM Design and Implementation. As an active security operations center lead and dedicated blue team member, he has firsthand knowledge of what it takes to defend an organization against advanced cyberattacks. John specializes in threat hunting, tactical SIEM design and optimization, and tailoring security operations to enable organizations to protect their most sensitive data.

Sponsor

SANS would like to thank this paper's sponsor:

MANDIANT®