SANS | Research Program

WHITE PAPER

# Using MITRE ATT&CK® as an Operational Framework: Prioritizing, Testing, and Sustaining Defense

Written by **Chris Crowley**

March 2026

TIDAL CYBER

# Introduction

**Cybersecurity practitioners today operate in an environment defined by constant change, incomplete information, and real operational consequences. New adversary techniques emerge faster than controls can be deployed, tools are added faster than they can be fully understood, and defenders are expected to explain why certain gaps existed and why specific tradeoffs were made.**

MITRE ATT&CK® has become one of the most important resources available to practitioners navigating this reality. It is an open, impartial, and authoritative repository of what adversaries do, grounded in observed real-world attacks. Across SOCs, detection engineering teams, threat intelligence functions, and security architecture groups, ATT&CK provides a shared language for describing what attackers actually do, not just what tools they use or which vulnerabilities they exploit.

However, practitioners rarely struggle with understanding ATT&CK itself. The challenge is operationalizing it.

ATT&CK is intentionally comprehensive, documenting tactics, techniques, and sub-techniques across the adversary life cycle. That depth makes it broadly applicable, but it also creates friction for teams tasked with turning knowledge into action. No security team has the time, budget, or staffing to implement equal coverage across the entire matrix. Attempting to do so often leads to shallow controls, excessive alert noise, and false confidence driven by static mappings rather than demonstrated capability.

Practitioners are left to answer difficult questions:

- Which techniques should we actually care about?
- Where are we confident our detections and preventions work, and where are we guessing?
- How do we prioritize detection engineering, hardening, and response work when everything appears equally “critical” on paper?
- How do we prove to ourselves, our leadership, and our auditors that our defensive posture is improving rather than quietly degrading?

Too often, ATT&CK is applied as a documentation exercise rather than an operational one. Coverage matrices are built manually, vendor claims are accepted at face value, and scores are assigned based on assumptions about tooling rather than evidence from real systems operating in real environments. When incidents occur, those assumptions collapse under scrutiny.

This paper is written for practitioners who need ATT&CK to do more than describe adversary behavior; they need it to guide daily defensive decisions. It provides practical guidance for using ATT&CK as a framework for prioritization, measurement, and validation, enabling teams to:

- Focus defensive efforts on the techniques most likely to be used against their organization
- Understand how deployed controls actually perform, not just how they are *supposed* to perform
- Integrate threat intelligence, vulnerability data, and system configuration into a coherent view of defensive posture
- Continuously assess whether changes to infrastructure and tooling are improving or degrading security
- Maintain ongoing awareness of shifting adversary activities

The paper begins by examining common patterns of vendor adoption and evaluation, highlighting both their value and their limitations from a practitioner's perspective. From there, the paper introduces an implementation approach centered on dynamic visibility, automation, and continuous validation, moving beyond static matrices toward defensible, evidence-based security operations.

For practitioners, effective use of ATT&CK is not about achieving theoretical coverage. It is about knowing, with confidence, which attacks you can stop, which ones you can detect quickly, and which ones remain high-risk—and why. This paper aims to help security teams build that confidence and sustain it as both their environments and their adversaries continue to evolve.

## ATT&CK Industry Scenarios of Use

In developing the ATT&CK framework, MITRE envisioned several potential uses for this body of knowledge. One was that it would serve as an authoritative repository of threat intelligence information for tracking and spoiling adversary capability. Once capability is known, it becomes feasible to develop defense against it—not that everyone does. But the first step in deploying defense is understanding the capability of the attacks.

Another intended use of ATT&CK is that it would serve to facilitate the development of behavioral analytics. These analytics enable the prevention, detection and response, and information systems engineering to obviate the weakness of the system that would allow unauthorized and unintended use.

An intention of transitioning to behavioral analytics was to move cybersecurity away from a self-limiting focus on specific software or indicators that adversaries use and toward the underlying technique used to accomplish the tactic. This is a noble endeavor, but one of many dumb ideas we as the cyber industry tend to relapse into: enumerating lists of bad things.

Behavioral analytics envision a movement away from lists of bad things: "It is a way of leveraging how an adversary interacts with a specific platform to identify and link together suspicious activity that is agnostic or independent of specific tools that may be used."[1] Detection decoupled from specific tool-based detections is desirable. It's even better if the system could be configured or engineered in a way that eliminates the effectiveness of the technique.

Not all unauthorized or unintended actions require an exploitable flaw. A weak password or a leaked password (reusable passwords as an authentication token, for that matter) are a vulnerability of a system. The point of an authentication token is to ensure only the authorized user is performing an action. The realm of access controls is to specify exactly what an account can and can't do. All of this exists because an organization envisions some idealized control of the information and actions that information systems can take.

If the configuration of the system can't be perfectly tuned to only allow the specific wanted actions to occur, then other mitigations, such as detections and post-execution interruptions, are expected to compensate for these system vulnerabilities.

Organizations are supposed to leverage ATT&CK to perform a defensive gap analysis, and crosswalk actually deployed cybersecurity defenses to available threat intelligence to assess the current defensive posture of the organization. This leads to the identification of gaps, and remediations are to be deployed to mitigate or remove these gaps.

This defensive gap analysis, combined with MITRE or other industry insight, is supposed to then assist with the maturity assessment of the organization's security operations. Greater maturity entails performing the work of integrating emerging threat information, detecting and handling threats at high speed, staffing, sustaining ongoing operations, and integrating newly deployed IT assets, all via cost-effective approaches. The "MITRE ATT&CK Design and Philosophy"[2] paper speaks about guiding cybersecurity investment. But there are no clear calculations, metrics, or direction given on how to select and implement this investment.

[1] "MITRE ATT&CK®: Design and Philosophy," March 2020, https://attack.mitre.org/docs/ATTACK_Design_and_Philosophy_March_2020.pdf, p. 13

[2] "MITRE ATT&CK®: Design and Philosophy," March 2020

# ATT&CK Vendor Adoption

So how to do this? Most security operations teams use tools to help visualize the defensive coverage of ATT&CK matrices. One readily available MITRE-released tool to do this is the MITRE ATT&CK Navigator.[3] The visualization in Figure 1 is recognizable to most. The "Technique Controls," for example, enable scoring on a per-technique basis, producing a red-yellow-green scale (default colors) on a 0–100 basis. This presumes there's some understanding of the deployed cybersecurity tools and architecture that address the techniques covered by these tools

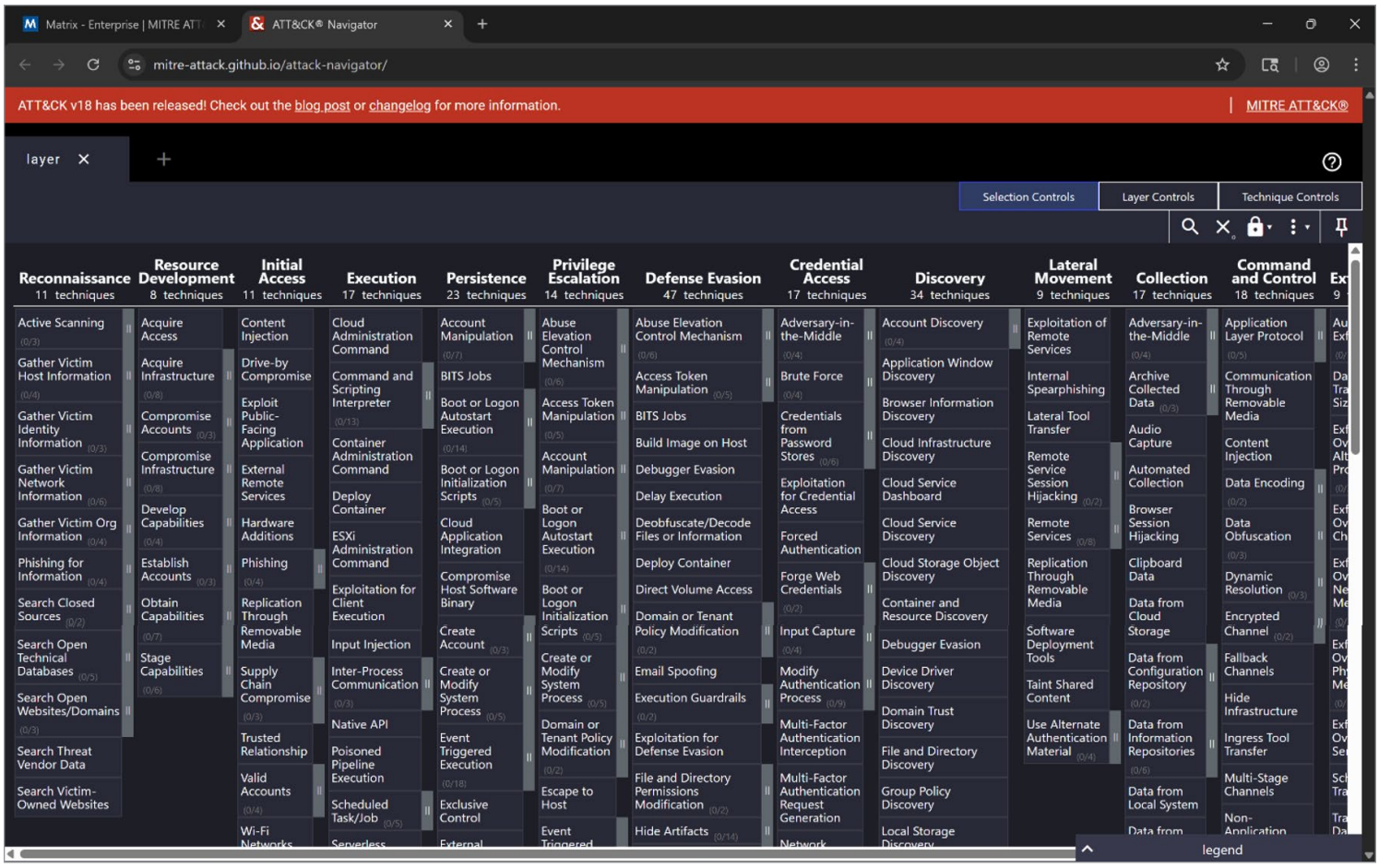


*Figure 1. MITRE ATT&CK Navigator*

Because vendors saw the value of the effort, they began annotating their tools with coverage provided. That is, "vendor_tool_rule-17" provides detective coverage for T1566.001 (Phishing.Spearphishing attachment). Of course, vendors could simply assert this without any sort of proof.

[3] https://mitre-attack.github.io/attack-navigator

As an additional effort, MITRE began their ATT&CK Evaluations in 2018. The first was on APT3[4] and had several techniques in scope. Figure 2 gives a notional view from the ATT&CK Enterprise Matrix at that time.

| Execution | Persistence | Privilege Escalation | Defense Evasion | Credential Access | Discovery | Lateral Movement | Collection | Command And | Exfiltration |
|---|---|---|---|---|---|---|---|---|---|
| CMSTP | Accessibility Features | Access Token Manipulation | Access Token Manipulation | Account Manipulation | Account Discovery | Application Deployment Software | Audio Capture | Commonly Used Port | Automated Exfiltration |
| Command-Line Interface | Account Manipulation | Accessibility Features | Binary Padding | Brute Force | Application Window Discovery | Distributed Component Object Model | Automated Collection | Communication Through Removable Media | Data Compressed |
| Compiled HTML File | AppCert DLLs | AppCert DLLs | BITS Jobs | Credential Dumping | Browser Bookmark Discovery | Exploitation of Remote Services | Clipboard Data | Connection Proxy | Data Encrypted |
| Control Panel Items | AppInit DLLs | AppInit DLLs | Bypass User Account Control | Credentials in Files | Domain Trust Discovery | Logon Scripts | Data from Information Repositories | Custom Command and Control Protocol | Data Transfer Size Limits |
| Dynamic Data Exchange | Application Shimming | Application Shimming | CMSTP | Credentials in Registry | File and Directory Discovery | Pass the Hash | Data from Local System | Custom Cryptographic Protocol | Exfiltration Over Alternative Protocol |
| Execution through API | Authentication Package | Bypass User Account Control | Code Signing | Exploitation for Credential Access | Network Service Scanning | Pass the Ticket | Data from Network Shared Drive | Data Encoding | Exfiltration Over Command and Control Channel |
| Execution through Module Load | BITS Jobs | DLL Search Order Hijacking | Compile After Delivery | Forced Authentication | Network Share Discovery | Remote Desktop Protocol | Data from Removable Media | Data Obfuscation | Exfiltration Over Other Network Medium |
| Exploitation for Client Execution | Bootkit | Exploitation for Privilege Escalation | Compiled HTML File | Hooking | Network Sniffing | Remote File Copy | Data Staged | Domain Fronting | Exfiltration Over Physical Medium |
| Graphical User Interface | Browser Extensions | Extra Window Memory Injection | Component Firmware | Input Capture | Password Policy Discovery | Remote Services | Email Collection | Domain Generation Algorithms | Scheduled Transfer |
| InstallUtil | Change Default File Association | File System Permissions Weakness | Component Object Model Hijacking | Input Prompt | Peripheral Device Discovery | Replication Through Removable Media | Input Capture | Fallback Channels | |
| LSASS Driver | Component Firmware | Hooking | Control Panel Items | Kerberoasting | Permission Groups Discovery | Shared Webroot | Man in the Browser | Multi-hop Proxy | |
| Mshta | Component Object Model Hijacking | Image File Execution Options Injection | DCShadow | LLMNR/NBT-NS Poisoning and Relay | Process Discovery | Taint Shared Content | Screen Capture | Multi-Stage Channels | |
| PowerShell | Create Account | New Service | Deobfuscate/Decode Files or Information | Network Sniffing | Query Registry | Third-party Software | Video Capture | Multiband Communication | |
| Regsvcs/Regasm | DLL Search Order Hijacking | Path Interception | Disabling Security Tools | Password Filter DLL | Remote System Discovery | Windows Admin Shares | | Multilayer Encryption | |
| Regsvr32 | External Remote Services | Port Monitors | DLL Search Order Hijacking | Private Keys | Security Software Discovery | Windows Remote Management | | Remote Access Tools | |
| Rundll32 | File System Permissions Weakness | Process Injection | DLL Side-Loading | Two-Factor Authentication Interception | System Information Discovery | | | Remote File Copy | |
| Scheduled Task | Hidden Files and Directories | Scheduled Task | Execution Guardrails | | System Network Configuration Discovery | | | Standard Application Layer Protocol | |
| Scripting | Hooking | Service Registry Permissions Weakness | Exploitation for Defense Evasion | | System Network Connections Discovery | | | Standard Cryptographic Protocol | |
| Service Execution | Hypervisor | SID-History Injection | Extra Window Memory Injection | | System Owner/User Discovery | | | Standard Non-Application Layer Protocol | |
| Signed Binary Proxy Execution | Image File Execution Options Injection | Valid Accounts | File Deletion | | System Service Discovery | | | Uncommonly Used Port | |
| Signed Script Proxy Execution | Logon Scripts | Web Shell | File Permissions Modification | | System Time Discovery | | | Web Service | |
| Third-party Software | LSASS Driver | | File System Logical Offsets | | Virtualization/Sandbox Evasion | | | | |
| Trusted Developer Utilities | Modify Existing Service | | Group Policy Modification | | | | | | |
| User Execution | Netsh Helper DLL | | Hidden Files and Directories | | | | | | |
| Windows Management Instrumentation | New Service | | Image File Execution Options Injection | | | | | | |
| Windows Remote | Office Application Startup | | Indicator Blocking | | | | | | |

*Figure 2. APT3 ATT&CK Evaluation Partial Matrix Depiction*

Many vendors were included as an initial cohort of tools under evaluation, with many more joining on a rolling admission basis.

This effort was intended to develop something like a *proof of performance* to demonstrate the effectiveness of the vendor's tools against adversary activity. This effort has been repeated extensively since and is generally deemed to be valuable to cybersecurity as it is provided by a reasonably impartial third party, as MITRE is a nonprofit entity held in high regard with a mission of enhancing public safety. This isn't yet a true certification scheme, but it is adequate for current capacity and cyber industry appetite for a public demonstration.

This still is inadequate for any organization to assess the threats that it faces. It is inadequate for the organization to understand its own defensive topography in the context of the systems as deployed and as configured.

MITRE can still be used, but the paradigm of use is something most organizations struggle to envision and fail to deploy. What should you do? Read the implementation guidance section to see the vision and start to implement it.

[4] https://evals.mitre.org/enterprise/apt3

# Implementation Guidance

ATT&CK provides a structural framework. Effectively using it isn't about building a static coverage matrix, although a coverage matrix is a step in the use of MITRE ATT&CK. The objective is to accomplish dynamic visibility into the state of an organization's defensive topography.

The dynamic component of visibility is multifaceted. Dynamic means changing. As you deploy new assets and defensive configurations, the defenses hopefully improve, but improvement is not assured. We would like an ongoing check to see if we're getting better or losing ground. A second facet is the changing threat landscape. New adversaries emerge, known adversaries adopt alternate techniques, and sophisticated adversaries are developing new techniques to accomplish existing tactics.

Every day, patches are released for already deployed software. Vulnerabilities in already deployed, assumed to be defended, assets are discovered. As part of ongoing operational due diligence for cybersecurity teams, this constant churn of discovery requires some mechanism for tracking the dynamic state of assets to be defended. Although ATT&CK strives to move away from tool-specific characterizations, teams are on guard against known techniques newly deployed against existing assets, which have recently been discovered to be vulnerable. ATT&CK framework information can be leveraged to demonstrate the deployed coverage, and to understand if new coverage needs to be deployed given a change in the threat landscape.

Returning to the enterprise matrix in Figure 1, and building upon it in Figure 3, we see a range of 0–100 encoded in three colors: red, yellow, and green. You can think of these as low, moderate, and high or whatever labels you prefer to show (e.g., no coverage, some coverage, or full coverage).

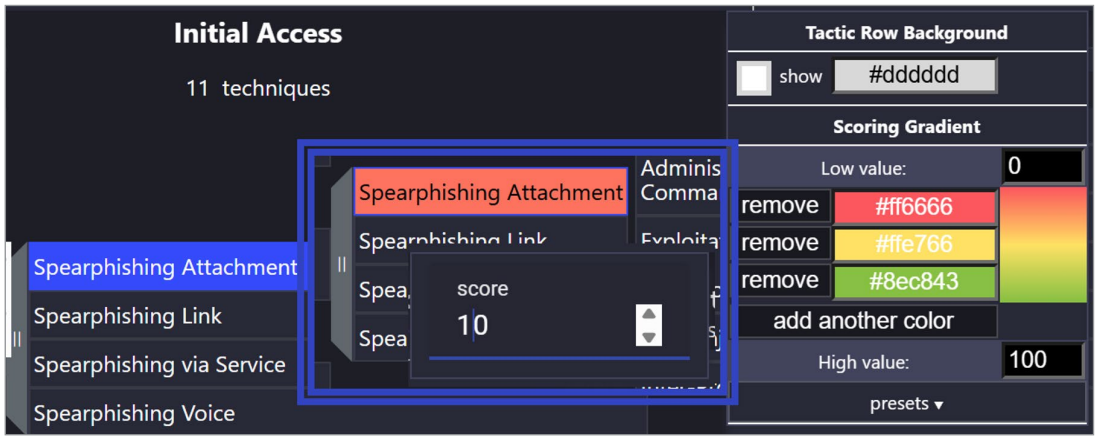


*Figure 3. MITRE ATT&CK Enterprise Matrix 0–100 Scale per Technique, Score 10 Superimposed for Spearphishing Attachment Subtechnique*

You shouldn't be manually inserting these values. If you are, please phase that approach out. You need a dynamic way to assess, score confidence, and report those values. You also need a threshold where you show the boundary at which you start to develop better defenses and a threshold where you implement emergency measures to mitigate the weakness. This emergency could be considered a cybersecurity incident per a classic definition of an incident: "the imminent threat of harm to an information system."

You shouldn't be manually evaluating the capability score for your defensive performance. This is challenging to accomplish, but testing developed during hunting and detection engineering can turn into automated testing in operational (or simulation) environments to assess the real-time defensive posture of your deployed assets. A quantitative assessment of posture on individual assets collectively contributes to the overall perspective of the defensive terrain of all information assets.

The collection of defensive information for ongoing status is an insurmountable task if done without extensive automation. Tools for collecting this information can be developed in house, customized through SOAR tools, extracted from endpoint detection and response (EDR) and asset configuration specifications, and programmatically ingested via SIEMs. But this would typically entail some do-it-yourself programming.

This defensive status information then needs to be integrated with system vulnerability information and targeted threat intelligence. The paper covers this knowledge synthesis next.

## Knowledge Synthesis

Regardless of technology leveraged, the synthesis of your defensive status information will provide guidance on how to enhance cyber posture. The fortification enhancement might take many forms: better detection to trigger responsive actions, disruption of adversary activity once it starts, or prevention through denial of adversary actions, which we'll cover in detail in this section.

Threat intelligence information identifies the adversaries likely to attack your organization. This intel can be externally collected from community sources, purchased in the form of indicator-driven threat feeds, purchased in the form of tactic and technique-driven threat feeds, and purchased in the form of organizationally tailored research by threat intel firms looking for organization information in adversary marketplaces and sharing channels. This intel also can be derived from internal observation based on previous detections and generation of intelligence through an assessment of root cause for intrusions from those detections. Internally generated threat intel has high fidelity to your organization, whereas externally generated threat intel provides forecasting and early warning to prepare for changing adversary techniques, which have not yet been launched against your organization.

Blending the two produces optimal results. Internally generated threat intel that provides an understanding of specific threat actors targeting your organization enhances the selection of external threat intelligence to forecast most likely types of actors who will target you in the future.

Vulnerability information from your internal systems should be combined with the threat information to better understand your defensive topography. There are reasons why systems aren't patched, ranging from carelessness or mistakes to intentional system configurations that can't be changed. Minimizing the errors is a task for the vulnerability management program, and the potential impact of such errors can be highlighted through this process of synthesizing the information. If there's an unexpected weakness in the configuration of a system, the detections and preventions might not work as expected. The actual as-deployed status of systems is dynamic and must be integrated into your knowledge synthesis to drive the organization to sustain its defensive posture. Just saying "patch" doesn't carry the same weight as saying, "Patch because there's an in-the-wild attack that is taking advantage of this specific vulnerability."

Defensive posture includes the cybersecurity software and configurations deployed to the assets. This includes EDR and their corresponding rules. It includes the operating system's configuration specifications as well as the software installed on the system for users and that software's settings. It includes identity and account management and the configuration of administrative accounts, rights, and groups. The host logging specification is important, as is whether or not the logs are shared to a repository off the system to resist tampering by the adversary after a compromise.

Defensive posture also includes network monitoring systems for observation of assets' behaviors from an objective network position. This lacks the details available from within the operating system but helps decide if something is acting in an undesirable way.

With this threat, vulnerability, and defensive posture information in hand, you now can assess what adversaries are attacking your organization and the sectors you operate within. What tactics are commonly leveraged? What techniques are employed to accomplish those tactics? For the techniques you are confident your defensive posture tools address already, you can reduce the priority of adding detections or preventive measures. For techniques where you lack confidence, look at the prevalence of vulnerabilities in the software (operating systems and applications) you deploy. If there is a higher prevalence of released vulnerabilities, pay more attention to adding controls there. If your organization has a difficult time accomplishing full deployment of patching, add controls for these techniques. If the assets that might be targeted have higher criticality, you might consider enhancing controls as well.

As you determine priority for enhancing defense, you can develop a portfolio of fortifications. The Diamond Model provides a useful list of descriptions for adversary interruption actions: detect, degrade, disrupt, deny, and deceive. The model also lists destroy, but that's out of scope for most organizations, I think. Tracking the fortifications in these categories is useful because you can stack protections. Further, each interruption technique can be assigned to specific defensive technology deployed and mapped to the ATT&CK technique and subtechnique it interrupts. There's an additional MITRE project D3FEND.[5] Instead of using the Diamond Model, the D3FEND taxonomy could be leveraged to track countermeasures deployed. The D3FEND taxonomy uses model, harden, detect, isolate, deceive, evict, and restore as its taxonomy categories, with ATT&CK references throughout. This can assist with the decision of what fortification to deploy.

As defensive posture is enhanced, it should be tested for validation of ongoing performance, as described earlier. This should be accomplished with automated testing. A commonly used toolkit for this testing is from Atomic Red Team.[6] This toolkit is open source and delivers MITRE ATT&CK-mapped tools. It also can be executed via the Invoke-AtomicRedTeam PowerShell module. This is certainly not the only tool available to help validate. Further, it's not an effective tool for tracking chained adversary behavior.

These tests are well-suited for validating controls at initial deployment and for repeated execution over time to confirm detections continue to function as intended. When used consistently, this approach helps maintain confidence in the system-to-SOC pipeline and supports ongoing assessment of detection coverage and basic SOC readiness.

This ongoing validation assists in avoiding self-defeating unauthorized configuration changes, loss of visibility through unexpected data inconsistencies, and undesirable cyber tool performance issues introduced through unrelated changes.

To strive for operational excellence, the team should develop purple and red team exercises. This enables tracing of campaign-oriented movements of adversaries, and moves away from granular, individual test units into adversary behavioral emulation. This could assist with assessing SOC analyst performance when challenges become more complex, and for assessing the ability for security to coordinate with other teams.

As defensive fortifications are deployed and tested in this manner, even new bypasses available to attackers are ruined. They won't be able to avoid the other snares and detections you have deployed.

[5] https://d3fend.mitre.org

[6] www.atomicredteam.io

# Conclusion

MITRE ATT&CK has succeeded in its original mission: It provides the cybersecurity community with a shared, behavior-based understanding of how adversaries operate. What remains challenging is the disciplined application of that knowledge in real operational environments.

For practitioners, the value of ATT&CK is not realized through static coverage maps, vendor-aligned matrices, or theoretical scoring exercises. It is realized when ATT&CK informs daily defensive decisions (e.g., what to build next, what to test, what to monitor more closely, and where risk is knowingly accepted). When used correctly, ATT&CK becomes a living model of adversary pressure against your actual systems, not an abstract catalog of threats.

The approach described in this paper reframes ATT&CK as a framework for *continuous defensive management* rather than a one-time assessment. It emphasizes evidence over assertion, automation over manual effort, and validation over assumption. Most importantly, it aligns defensive investment to the adversaries, techniques, assets, and constraints that are specific to your organization.

Security teams looking to move from ATT&CK awareness to ATT&CK effectiveness should begin with the following steps:

- **Define your threat focus.** Identify the adversary types most likely to target your organization based on sector, geography, and business model. Map only the relevant ATT&CK tactics and techniques for those adversaries rather than attempting blanket coverage of the entire matrix.
- **Inventory and map actual defenses.** Document what is truly deployed and enabled across endpoints, identities, networks, and cloud environments. Map these controls to ATT&CK techniques based on how they operate in practice.
- **Replace assumed coverage with evidence.** Stop assigning confidence scores manually. Use detection engineering tests, red team tooling, or frameworks such as Atomic Red Team to validate whether controls actually detect, prevent, or disrupt specific techniques in your environment.
- **Automate posture measurement.** Build or integrate automation to continuously collect defensive telemetry from EDR, SIEM, SOAR, configuration management, and asset inventories. Treat defensive posture as a dynamic signal that changes as systems, configurations, and threats evolve.
- **Synthesize threat, vulnerability, and posture data.** Combine targeted threat intelligence, vulnerability data, and defensive capability into a single analytical view. Use this synthesis to prioritize detection gaps, hardening efforts, and compensating controls where patching or prevention is not feasible.

- **Stack and track defensive fortifications.** For high-risk techniques, deploy layered controls across detect, disrupt, deny, and deceive actions. Track these fortifications explicitly and map them back to ATT&CK techniques to understand where redundancy exists and where single points of failure remain.
- **Continuously validate and re-test.** Incorporate automated testing into operational workflows to ensure detections continue to function over time. Re-run tests after tool upgrades, configuration changes, and infrastructure deployments to prevent silent degradation of defensive capability.
- **Make risk decisions explicit and defensible.** When gaps remain—and they will—document why. Tie accepted risk to business context, asset criticality, and adversary likelihood. This transforms security decisions from reactive justifications into intentional, defensible choices.

ATT&CK does not make organizations secure. Practitioners do, by using ATT&CK to guide disciplined prioritization, continuous measurement, and ongoing validation of defensive effectiveness. Organizations that treat ATT&CK as a living operational framework gain more than visibility. They gain confidence in their ability to detect, respond, and adapt as adversaries evolve.

## Sponsor

**SANS would like to thank this paper's sponsor:**