MP180360
MITRE PRODUCT

MITRE

MITRE ATT&CK™: Design and Philosophy

Project No.: 01ADM105-PI

McLean, VA

Authors:

Blake E. Strom
Andy Applebaum
Doug P. Miller
Kathryn C. Nickels
Adam G. Pennington
Cody B. Thomas

July 2018

Abstract

The MITRE ATT&CK knowledgebase describes cyber adversary behavior and provides a common taxonomy for both offense and defense. It has become a useful tool across many cyber security disciplines to convey threat intelligence, perform testing through red teaming or adversary emulation, and improve network and system defenses against intrusions. The process MITRE used to create ATT&CK, and the philosophy that has developed for curating new content, are critical aspects of the work and are useful for other efforts that strive to create similar adversary models and information repositories.

This page intentionally left blank.

Executive Summary

This paper discusses the motivation behind the creation of ATT&CK, the components described within it, its design philosophy, how the project has progressed, and how it can be used. It is meant to be used as an authoritative source of information about ATT&CK as well as a guide for how ATT&CK is maintained and how ATT&CK-based knowledge bases are created for new technology-domains and platforms.

Preface

This paper documents the published version of ATT&CK as of April 2018. MITRE has announced plans to evolve and expand ATT&CK throughout 2018 [1]. This paper will be maintained as a living document and will be updated as significant changes are made to ATT&CK and the process used to maintain the content within ATT&CK.

Table of Contents

List of Figures

List of Tables

This page intentionally left blank.

1 Introduction

MITRE's Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK) is a curated knowledge base and model for cyber adversary behavior, reflecting the various phases of an adversary's attack lifecycle and the platforms they are known to target. ATT&CK originated out of a project to enumerate and categorize post-compromise adversary tactics, techniques and procedures (TTPs) against Microsoft Windows™ systems to improve detection of malicious activity. It has since grown to include Linux™ and MacOS™, and has expanded to cover pre-compromise tactics and techniques, and technology-focused domains like mobile devices. At a high-level, ATT&CK is a behavioral model that consists of the following core components:

- Tactics, denoting short-term, tactical adversary goals during an attack (the columns);
- Techniques, describing the means by which adversaries achieve tactical goals (the individual cells);
- Documented adversary usage of techniques and other metadata (linked to techniques).

ATT&CK is not an exhaustive enumeration of attack vectors against software. Other MITRE efforts such as CAPEC™ [2] and CWE™ [3] are more applicable to this use case.

1.1 Background and History

ATT&CK was created out of a need to systematically categorize adversary behavior as part of conducting structured adversary emulation exercises within MITRE's Fort Meade Experiment (FMX) research environment. Established in 2010, FMX provided a "living lab" capability that allowed researchers access to a production enclave of the MITRE corporate network to deploy tools, test, and refine ideas on how to better detect threats. MITRE began researching data sources and analytic processes within FMX for detecting advanced persistent threats (APTs) more quickly under an "assume breach" mentality. Cyber game exercises were conducted on a periodic basis to emulate adversaries within the heavily sensored environment and hunting was performed to test analytic hypotheses against the data collected. The goal was to improve post-compromise detection of threats penetrating enterprise networks through telemetry sensing and behavioral analytics [4]. The primary metric for success was "How well are we doing at detecting documented adversary behavior?" To effectively work towards that goal, it proved useful to categorize observed behavior across relevant real-world adversary groups and use that information to conduct controlled exercises emulating those adversaries within the FMX environment. ATT&CK was used by both the adversary emulation team (for scenario development) and the defender team (for analytic progress measurement), which made it a driving force within the FMX research.

The first ATT&CK model was created in September 2013 and was primarily focused on the Windows enterprise environment. It was further refined through internal research and development and subsequently publicly released in May 2015 with 96 techniques organized under 9 tactics. Since then, ATT&CK has experienced tremendous growth based on contributions from the cybersecurity community. Based on the methodology used to create the first ATT&CK model, a complementary knowledge base called PRE-ATT&CK™ was created to focus on "left of exploit" behavior, and ATT&CK for Mobile was created to focus on behavior

in the mobile-specific domain. As of April 2018, Enterprise ATT&CK now includes 219 techniques across Windows, Linux, and Mac.

2 ATT&CK Use Cases

Adversary Emulation – The process of assessing the security of a technology domain by applying cyber threat intelligence about specific adversaries and how they operate to emulate that threat. Adversary emulation focuses on the ability of an organization to verify detection and/or mitigation of the adversarial activity at all applicable points in their lifecycle.

ATT&CK can be used as a tool to create adversary emulation scenarios to test and verify defenses against common adversary techniques. Profiles for specific adversary groups can be constructed out of the information documented in ATT&CK (see Cyber Threat Intelligence use case). These profiles can also be used by defenders and hunting teams to align and improve defensive measures.

Red Teaming – Applying an adversarial mindset without use of known threat intelligence for the purpose of conducting an exercise. Red teaming focuses on accomplishing the end objective of an operation without being detected to show mission or operational impact of a successful breach.

ATT&CK can be used as a tool to create red team plans and organize operations to avoid certain defensive measures that may be in place within a network. It can also be used as a research roadmap to develop new ways of performing actions that may not be detected by common defenses.

Behavioral Analytics Development – By going beyond traditional indicators of compromise (IoCs) or signatures of malicious activity, behavioral detection analytics can be used to identify potentially malicious activity within a system or network that may not rely on prior knowledge of adversary tools and indicators. 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.

ATT&CK can be used as a tool to construct and test behavioral analytics to detect adversarial behavior within an environment. The Cyber Analytics Repository[1] (CAR) is one example of analytic development that could be used as a starting point for an organization to develop behavioral analytics based on ATT&CK.

Defensive Gap Assessment – A defensive gap assessment allows an organization to determine what parts of its enterprise lack defenses and/or visibility. These gaps represent blind spots for potential vectors that allow an adversary to gain access to its networks undetected or unmitigated.

ATT&CK can be used as a common behavior-focused adversary model to assess tools, monitoring, and mitigations of existing defenses within an organization's enterprise. The identified gaps are useful as a way to prioritize investments for improvement of a security program. Similar security products can also be compared against a common adversary behavior model to determine coverage prior to purchasing.

SOC Maturity Assessment – An organization's Security Operations Center is a critical component of many medium to large enterprise networks that continuously monitor for active

[1] https://car.mitre.org

threats against the network. Understanding the maturity of a SOC is important to determine its effectiveness.

ATT&CK can be used as one measurement to determine how effective a SOC is at detecting, analyzing, and responding to intrusions. Similar to the defensive gap assessment, a SOC Maturity assessment focuses on the processes a SOC uses to detect, understand, and respond to changing threats to their network over time.

Cyber Threat Intelligence Enrichment – Cyber threat intelligence covers knowledge of cyber threats and threat actor groups that impact cybersecurity. It includes information about malware, tools, TTPs, tradecraft, behavior, and other indicators that are associated to threats.

ATT&CK is useful for understanding and documenting adversary group profiles from a behavioral perspective that is agnostic of the tools the group may use. Analysts and defenders can better understand common behaviors across many groups and more effectively map defenses to them and ask questions such as "what is my defensive posture against adversary group APT3?" Understanding how multiple groups use the same technique behavior allows analysts to focus on impactful defenses that span may types of threats. The structured format of ATT&CK can add value to threat reporting by categorizing behavior beyond standard indicators.

Multiple groups within ATT&CK use the same techniques. For this reason, it is not recommended to attribute activity solely based on the ATT&CK techniques used. Attribution to a group is a complex process involving all parts of the Diamond Model [5], not solely on an adversary's use of TTPs.

3 The ATT&CK Model

The basis of ATT&CK is the set of individual techniques that represent actions that adversaries can perform to accomplish objectives. Those objectives are represented by the tactic categories the techniques fall under. This relatively simple representation strikes a useful balance between sufficient technical detail at the technique level and the context around why actions occur at the tactic level.

3.1 The ATT&CK Matrix™

The relationship between tactics and techniques can be visualized in the ATT&CK Matrix. For example, under the Persistence tactic (this is the adversary's goal – to persist in the target environment), there are a series of techniques including AppInit DLLs, New Service, and

Scheduled Task. Each of these is a single technique that adversaries may use to achieve the goal of persistence. Figure 1 depicts the ATT&CK Matrix for enterprise systems.

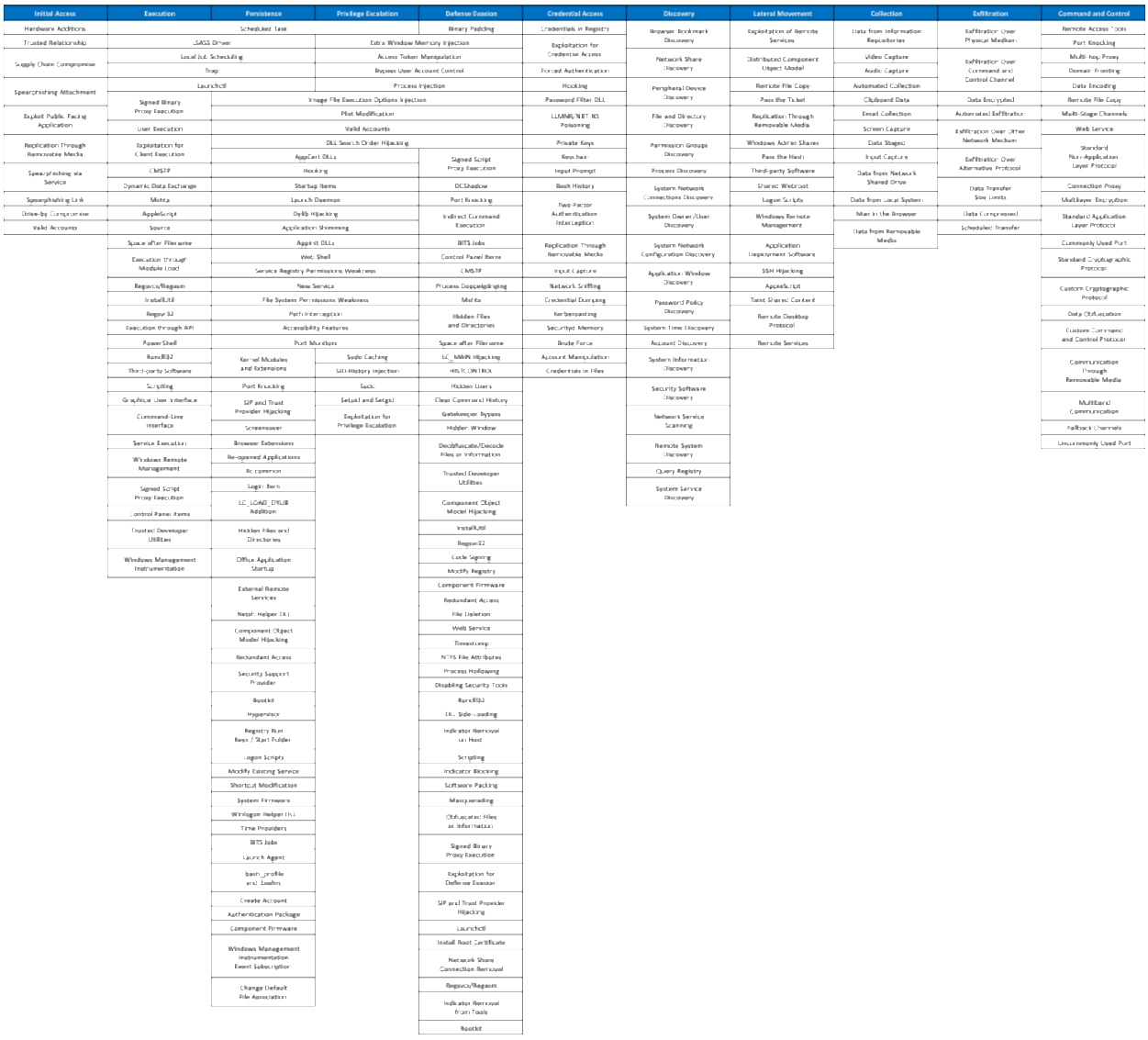

Figure 1. The ATT&CK for Enterprise Matrix

3.2 Technology Domains

ATT&CK is organized in a series of "technology domains" - the ecosystem an adversary operates within that provides a set of constraints the adversary must circumvent or take advantage of to accomplish a set of objectives. To date MITRE has defined two technology domains – Enterprise (representing traditional enterprise networks) and Mobile (for mobile communication devices). Within each technology domain, ATT&CK defines multiple "platforms" - the system an adversary is operating within. A platform may be an operating system or application (e.g. Microsoft Windows). Techniques can apply to multiple platforms. Table 1 lists the platforms currently defined for ATT&CK technology domains.

The scope of ATT&CK also expands beyond technology domains with PRE-ATT&CK. PRE-ATT&CK covers documentation of adversarial behavior during requirements gathering, reconnaissance, and weaponization before access to a network is obtained. It is independent of technology and models an adversary's behavior as they attempt to gain access to an organization or entity through the technology they leverage, spanning multiple domains.

Table 1. ATT&CK Technology Domains

Technology Domain	Platform(s) defined
Enterprise	Linux, macOS, Windows
Mobile	Android, iOS

3.3 Tactics

Tactics represent the "why" of an ATT&CK technique. It is the adversary's tactical objective: the reason for performing an action. Tactics serve as useful contextual categories for individual techniques and cover standard notations for things adversaries do during an operation, such as persist, discover information, move laterally, execute files, and exfiltrate data. Tactics are treated as "tags" within ATT&CK where a technique is associated or tagged with one or more tactic categories depending on the different results that can be achieved by using a technique.

Each tactic contains a definition describing the category and serves as a guide for what techniques should be within the tactic. For example, Execution is defined as a tactic that represents techniques that result in execution of adversary-controlled code on a local or remote system. This tactic is often used in conjunction with initial access as the means of executing code once access is obtained, and lateral movement to expand access to remote systems on a network.

Additional tactic categories may be defined as needed to more accurately describe adversary objectives. Applications of the ATT&CK modeling methodology for other domains may require new or different categories to associate techniques even though there may be some overlap with tactic definitions in existing models.

3.4 Techniques

Techniques represents "how" an adversary achieves a tactical objective by performing an action. For example, an adversary may dump credentials to gain access to useful credentials within a network. Techniques may also represent "what" an adversary gains by performing an action. This is a useful distinction for the Discovery tactic as the techniques highlight what type of information an adversary is after with a particular action. There may be many ways, or techniques, to achieve tactical objectives, so there are multiple techniques in each tactic category.

3.4.1 Technique Object Structure

These terms represent sections and important information included within each technique entry within the **Enterprise ATT&CK model**. Items are annotated by **tag** if the data point is an informational reference on the technique that can be used to filter and pivot on, and **field** if the item is a free text field used to describe technique-specific information and details. Items marked with **relationship** indicate fields that are associated to technique entity relationships with groups

and software that use the technique. Table 2 lists all of the data items currently defined for techniques in ATT&CK. Data items marked with * denote the element is required and additional information about specific requirements dependent on tactic category is in the description.

Table 2. ATT&CK Technique Model

Data Item	Type	Description
Name*	Field	The name of the technique
ID*	Tag	Unique identifier for the technique within the knowledgebase. Format: T####.
Tactic*	Tag	The tactic objectives that the technique can be used to accomplish. Techniques can be used to perform one or multiple tactics.
Description*	Field	Information about the technique, what it is, what it's typically used for, how an adversary can take advantage of it, and variations on how it could be used. Include references to authoritative articles describing technical information related to the technique as well as in the wild use references as appropriate.
Platform*	Tag	The system an adversary is operating within; could be an operating system or application (e.g. Microsoft Windows). Techniques can apply to multiple platforms.
System Requirements	Field	Additional information on requirements the adversary needs to meet or about the state of the system (software, patch level, etc.) that may be required for the technique to work.
Permissions Required*	Tag	The lowest level of permissions the adversary is required to be operating within to perform the technique on a system. *Required for privilege escalation.
Effective Permissions*	Tag	The level of permissions the adversary will attain by performing the technique. Only applies to techniques under the privilege escalation tactic. May have multiple entries if effective permissions can be set when technique is executed. *Required for privilege escalation
Data Source*	Tag	Source of information collected by a sensor or logging system that may be used to collect information relevant to identifying the action being performed, sequence of actions, or the results of those actions by an adversary. The data source list can incorporate different variations of how the action could be performed for a particular technique. This attribute is intended to be restricted to a defined

		list to allow analysis of technique coverage based on unique data sources. (For example, "what techniques can I detect if I have process monitoring in place?")
Supports Remote	Tag	If the technique can be used to execute something on a remote system. Applies to execution techniques only.
Defense Bypassed*	Tag	If the technique can be used to bypass or evade a particular defensive tool, methodology, or process. Applies to defense evasion techniques only. *Required for defense evasion.
CAPEC ID	Field	Hyperlink to related CAPEC entry on the CAPEC site.
Contributor	Tag	List of non-MITRE contributors (individual and/or organization) from first to most recent that contributed information on, about, or supporting the development of a technique.
Examples	Relationship / Field	Example fields are populated on a technique page when a group or software entity is associated to a technique through documented use. They describe the group or software entity with a brief description of how the technique is used. The example of how a specific adversary uses a technique is a direct reference to their procedures, or exact way of how they perform a technique on a system.
Detection*	Field	High level analytic process, sensors, data, and detection strategies that can be useful to identify a technique has been used by an adversary. This section is intended to inform those responsible for detecting adversary behavior (such as network defenders) so they can take an action such as writing an analytic or deploying a sensor. There should be enough information and references to point toward useful defensive methodologies. There could be many ways of detecting a technique but ATT&CK and MITRE do not endorse any particular vendor solution. Detection recommendations should therefore remain vendor agnostic, recommending the general method and class of tools rather than a specific tool. Detection may not always be possible for a given technique and should be documented as such.
Mitigation*	Field	Configurations, tools, or processes that prevent a technique from working or having the desired outcome for an adversary. This section is intended to inform those responsible for mitigating against

	adversaries (such as network defenders or policymakers) to allow them to take an action such as changing a policy or deploying a tool. Mitigation recommendations should remain vendor agnostic, recommending the general method rather than a specific tool. Mitigation may not always be possible for a given technique and should be documented as such.

3.5 Groups

Known adversaries that are tracked by public and private organizations and reported on in threat intelligences reports are tracked within ATT&CK under the Group object. Groups are defined as named intrusion sets, threat groups, actor groups, or campaigns that typically represent targeted, persistent threat activity. ATT&CK primarily focuses on APT groups though it may also include other advanced groups such as financially motivated actors.

Groups can use techniques directly or employ software that implements techniques.

3.5.1 Group Object Structure

Items are annotated by **tag** if the data point is an informational reference on the group that can be used to filter and pivot on, and **field** if the item is a free text field used to describe technique-specific information and details. Items marked with **relationship** indicate fields that are associated to technique entity relationships with techniques and software that use the technique. Data items marked with * denote the element is required

Table 3. ATT&CK Group Model

Data Item	Type	Description
Name*	Field	The name of the adversary group.
ID*	Tag	Unique identifier for the group within the knowledgebase. Format: G####.
Aliases	Tag	Alternative names that refer to the same adversary group in threat intelligence reporting.
Description*	Field	A description of the group based on public threat reporting. It may contain dates of activity, suspected attribution details, targeted industries, and notable events that are attributed to the group's activities.
Alias Descriptions	Field	Section that can be used to describe a groups' aliases with references to the report used to tie the alias to the group name.
Techniques Used*	Relationship / Field	List of techniques that are used by the group with a field to describe details on how the technique is used. This represents the group's procedure (in the

		context of TTPs) for using a technique. Each technique should include a reference.
Software	Relationship / Field	List of software that the group has been reported to use with a field to describe details on how the software is used.

3.6 Software

Adversaries commonly use different types of software during intrusions. Software can represent an instantiation of a technique, so they are also necessary to categorize within ATT&CK for examples on how techniques are used. Software is broken out into three high-level categories: tools, utilities, and malware.

- **Tool** - Commercial, open-source, or publicly available software that could be used by a defender, pen tester, red teamer, or an adversary for malicious purposes that generally is not found on an enterprise system. Examples include PsExec, Metasploit, Mimikatz, etc.
- **Utility** - Software generally available as part of an operating system that is likely already present in an environment. Adversaries tend to leverage existing functionality on systems to gather information and perform actions. Examples include Windows utilities such as Net, netstat, Tasklist, etc.
- **Malware** - Commercial, custom closed source, or open source software intended to be used for malicious purposes by adversaries. Examples include PlugX, CHOPSTICK, etc.

The software categories could be broken down further, but the idea behind the current categorization was to show how adversaries use utilities and legitimate software to perform actions much like they do with traditional malware.

3.6.1 Software Object Structure

Items are annotated by **tag** if the data point is an informational reference on the software that can be used to filter and pivot on, and **field** if the item is a free text field used to describe technique-specific information and details. Items marked with **relationship** indicate fields that are associated to technique entity relationships with techniques and groups. Data items marked with * denote the element is required.

Table 4. ATT&CK Software Model

Data Item	Type	Description
Name*	Field	The name of the software.
ID*	Tag	Unique identifier for the software within the knowledgebase. Format: S####.
Aliases	Tag	Alternative names that refer to the same software in threat intelligence reporting.
Type*	Tag	Type of software: malware, tool, utility.
Platform*	Tag	Platform the software can be used on. E.g., Windows.

Description*	Field	A description of the software based on technical references or public threat reporting. It may contain ties to groups known to use the software or other technical details with appropriate references.
Alias Descriptions	Field	Section that can be used to describe the software's aliases with references to the report used to tie the alias to the group name.
Techniques Used*	Relationship / Field	List of techniques that are implemented by the software with a field to describe details on how the technique is implemented or used. Each technique should include a reference.
Groups	Relationship / Field	List of groups that the software has been reported to be used by with a field to describe details on how the software is used. This information is populated from the associated group entry.

3.7 ATT&CK Object Model Relationships

Each high-level component of ATT&CK is related to other components in some way. The relationships described in the description fields in the previous section can be visualized in a diagram:

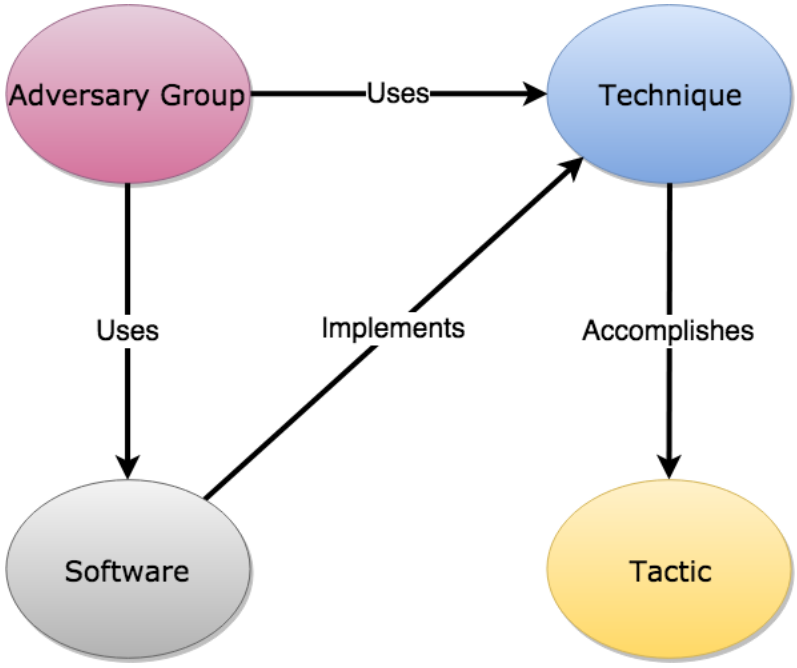

Figure 2. ATT&CK Model Relationships

An example as applied to a specific persistent threat group where APT28 uses Mimikatz for credential dumping:

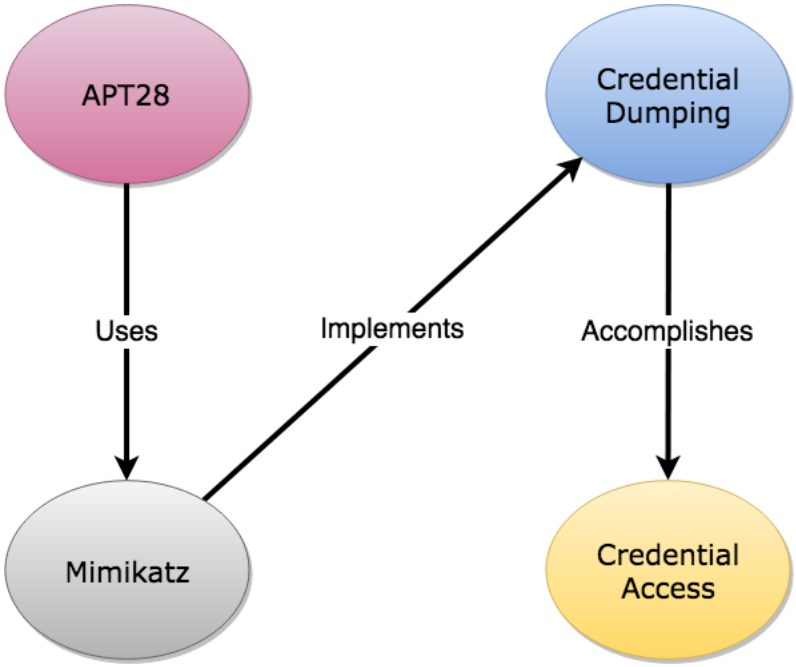

Figure 3. ATT&CK Model Relationships Example

4 The ATT&CK Methodology

The previous sections of this document have described and defined the purpose and structure of the ATT&CK knowledge base. This section describes the conceptual components of the methodology used in the creation and maintenance of ATT&CK. It also describes the process recommended to determine if and when new techniques should be added to the knowledge base and how threat intelligence is used to form the group and software technique profiles.

The information within ATT&CK has evolved over time, as have the considerations used for what information gets included and how it's structured. The process is as much of an art as it is a science but remains focused on an accurate representation of how adversaries conduct operations in a way that's easy to categorize the actions they take and relate those actions to sensors, system configurations, and countermeasures that defenders can use to detect and/or stop those actions.

4.1 Conceptual

There are three conceptual ideas that are core to the philosophy behind ATT&CK:

- It maintains the adversary's perspective;
- It follows real-world use of activity through empirical use examples;
- The level of abstraction is appropriate to bridge offensive action with possible defensive countermeasures.

4.1.1 Adversary's Perspective

ATT&CK takes on the perspective of an adversary in its terminology and descriptions for tactics and techniques described in the model. By contrast, many security models describe desired security from a defender's perspective with a top-down view, such as the CIA[2] model, focus on vulnerability scoring, such as CVSS [6], or primarily account for risk calculations, such as DREAD [7].

ATT&CK's use of an adversary's perspective makes it easier to understand actions and potential countermeasures in context than it would from a purely defense perspective. For detection, oftentimes defensive analysts are presented with alerts with little to no context about the event that caused the alert. This may cause a shallow frame of reference for what caused those alerts and how that cause relates to other events that may have occurred on a system or network.

The perspective shift changes the question from what *did* happen based on a list of available resources to what *could* happen with a framework for aligning a defensive strategy to the adversary's playbook. In part, ATT&CK provides a more accurate frame of reference for how to approach assessing defensive coverage. It conveys the relationships and dependencies between adversarial actions and information in a way that's agnostic of any particular defensive tool or method of collecting data. Defenders are then able to follow the adversary's motivation for individual actions and understand how the actions and dependencies relate to specific classes of defenses that may be deployed in an environment.

[2] Confidentiality, Integrity, and Availability

4.1.2 Empirical Use

The activity described by ATT&CK is largely drawn from publicly reported incidents on suspected advanced persistent threat group behavior, which provides a grounding for the knowledge base so that it accurately portrays activity happening or likely to happen in the wild. ATT&CK also draws from techniques discovered and reported through offensive research into areas that adversaries and red teams are likely to leverage against enterprise networks, such as techniques that can subvert modern and commonly used defenses. The tie to incidents keeps the model grounded to real-world threats that are likely to be encountered rather than theoretical techniques that are unlikely to be seen due to difficulty of use or low utility.

4.1.2.1 Sources of Information

New information relevant to ATT&CK techniques can come from many different sources. These sources are used to help meet the empirical use criteria:

- Threat intelligence reports
- Conference presentations
- Webinars
- Social media
- Blogs
- Open source code repositories
- Malware samples

4.1.2.2 Un(der)reported Incidents

The vast majority of incidents discovered are not reported publicly. Unreported, or underreported, incidents can contain valuable information on how adversaries behave and engage in operations. Often, the techniques used can be separated from potentially sensitive or damaging information and help provide insights into new techniques and variations, as well as statistical data to show prevalence of use.

This type of circumstantial evidence of use is valuable and is taken into consideration as empirical use related data when adding new information into ATT&CK.

4.1.3 Abstraction

The level of abstraction for adversary tactics and techniques within ATT&CK is an important distinction between it and other types of threat models. High level models such as the various adversary lifecycles, including the Lockheed Martin Cyber Kill Chain®, Microsoft STRIDE, etc., are useful at understanding high level processes and adversary goals. However, these models are not effective at conveying what individual actions adversaries make, how one action relates to another, how sequences of actions relate to tactical adversary objectives, and how the actions correlate with data sources, defenses, configurations, and other countermeasures used for the security of a platform and domain.

By contrast, exploit databases and models describe specific instances of exploitable software – which are often available for use with code examples – but are very far removed from the circumstances in which they could or should be used as well as from the difficulty of using them. Similarly, malware databases also exist but typically lack context around how the malware is used and by whom. They also do not take into account how legitimate software can be used for malicious purposes.

A mid-level adversary model like ATT&CK is necessary to tie these various components together. The tactics and techniques in ATT&CK define adversarial behaviors within a lifecycle to a degree where they can be more effectively mapped to defenses. The high-level concepts like Control, Execute, and Maintain are further broken down into more descriptive categories where individual actions on a system can be defined and categorized. A mid-level model is also useful to put lower level concepts into context. Behavior-based techniques are the focus as opposed to exploits and malware because they are numerous but are difficult to reason about them with a holistic defensive program other than regular vulnerability scans, rapid patching, and IOCs. Exploits and malicious software are useful to an adversary toolkit, but to fully understand their utility, it's necessary to understand the context in which they can be used to achieve a goal. The mid-level model is also a useful construct to tie in threat intelligence and incident data to show who is doing what as well as the prevalence of use for particular techniques. Figure 4 shows a comparison of the level of abstraction between high, mid, and low level models and threat knowledge databases:

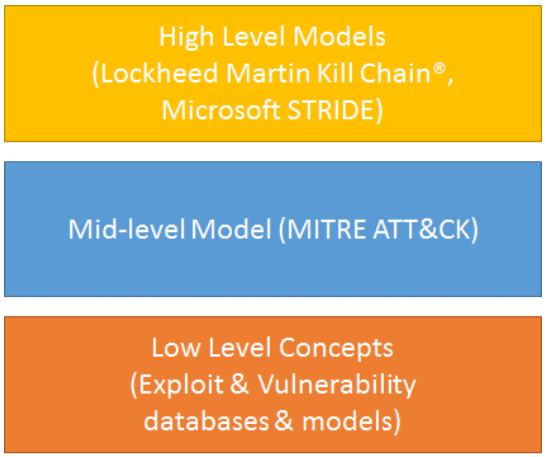

Figure 4. Abstraction Comparison of Models and Threat Knowledge Databases

What the ATT&CK technique abstraction provides:

- A common taxonomy of individual adversary actions and goals understood by both offense and defense.
- An appropriate level of categorization to relate adversary's action and *specific* ways of defending against it.

4.2 Tactics

Since tactics represent the tactical goals of an adversary, these remain relatively static over time because adversary goals are unlikely to change. Tactics combine aspects of what the adversary is trying to accomplish with what platform and domain they are operating within. Often these goals will be similar across platforms, which is why the Enterprise ATT&CK tactics are consistent across Windows, macOS, and Linux, and are even very similar to the Use Device Access tactics in ATT&CK for Mobile. Places where they differ are going to be where adversary goals and platform or domain technologies differ. An example of this is again evident with the ATT&CK for Mobile to cover how adversaries may downgrade or intercept connections between mobile devices and their network or service provider.

There may be cases where tactics need to be refined for better definition of the actions occurring. In the original ATT&CK for Enterprise, Windows the Collection tactic did not exist; instead it was included as part of Exfiltration. This representation fit sufficiently at the time because it was largely seen as one action—an adversary exfiltrates information but did not accurately represent the distinct motives and actions necessary for successful exfiltration. Where the data comes from and how it is obtained is equally as important as how an adversary removes the data from an environment and also represents distinct places where those actions can be detected. There is also a timing difference between when an adversary may collect information and when they exfiltrate it. Thus, a determination was made to break that tactic into two and describe Collection separately.

New tactics will follow the need to define existing, but uncategorized, or new adversary goals as a way to provide accurate context for what an adversary is accomplishing by performing a technique action.

4.3 Techniques

Techniques are the foundation of ATT&CK and represent the individual actions adversaries make or pieces of information the adversary learns by performing an action.

4.3.1 What Makes a Technique

There are several factors to a technique within ATT&CK. All factors are weighed in the decision process to create a technique and contribute to the information that represents a technique within the knowledgebase.

4.3.1.1 Naming

Technique names focus on the aspect of the technique that makes it unique—what the adversary achieves at an intermediate level of abstraction from using the tactic or how it's used at a lower level of abstraction. One example of the former is Credential Dumping [8] for Credential Access where dumping credentials is one method of gaining access to new credentials—and credentials can be dumped in several different ways. An example of the latter is Rundll32 [9] for Execution or Defense Evasion at a lower level of abstraction where there is a specific way the technique can used. Industry-accepted terminology tends to be used if it is already established and documented through conference presentations, blog posts, other articles, etc.

4.3.1.2 Types of Technique Abstraction

Techniques generally fall into three levels of abstraction:

1. General techniques that apply to multiple platforms in a general way (e.g. Obfuscated Files and Information [10])
2. General techniques that apply to multiple platforms in specific ways (e.g. Process Injection [11])
3. Specific techniques that only apply to one platform (e.g. Rundll32 [9])

For the first, breaking out how that technique applies to multiple platforms with specific sections for each platform in the technical description likely does not make sense because the technique describes a general platform agnostic behavior, such as much of the Command and Control tactic. The description is kept general and details are provided with references to the examples from the different platforms as needed.

Techniques that can be performed a few different ways to achieve the same or similar results are grouped under a general category of techniques, such as Credential Dumping. These techniques can apply to multiple platforms in specific ways that are described in the technical description broken down into platform specific sections. Oftentimes these techniques will contain variations for how they apply to a particular platform, like Process Injection.

More atomic techniques generally are specific ways an adversary acts against a particular platform. Rundll32 is one example that only applies to Windows systems. These techniques tend to describe how individual components of the platform are abused by adversaries.

Sometimes techniques can have multiple required steps within them, some of these steps may be relatable to other existing techniques or steps that could be individual techniques. When this occurs, it is important to focus on the distinguishing attribute of the technique or what makes it different than the others.

4.3.1.3 Technical References

Technical references are provided to point users to further research or more detail on techniques. Areas where technical references are useful include: background on the technique, expected use in benign cases, general use examples, variations of a technique, relevant tools and open source code repositories, detection examples and best practices, and mitigation examples and best practices.

4.3.1.4 Adversary Use

ATT&CK also includes information on if (and by whom) a technique is used in the wild and its reported impacts. As mentioned in the empirical use section, there are many sources of this information. ATT&CK remains strongly tied to threat intelligence sources on persistent threat groups. As the scope of ATT&CK has expanded and been refined, so too have the criteria necessary to add information. ATT&CK also includes public offensive research used by red teams against enterprise networks since adversaries have been known to adopt such published techniques. There are also fewer persistent threat incidents reported against Linux and Mac systems than there are against Windows, causing available threat data to be substantially less available. General in-the-wild sources of data that are not necessarily tied to persistent threat

group use may be used in lieu when the techniques align well with how persistent threats typically behave.

There are several general categories of empirical use information that can be used:

- **Reported** – Technique is reported with in the wild use through public sources.
- **Reported, non-public** – Technique use is reported in non-public sources but knowledge of the technique existing is present in public sources.
- **Underreported** – Techniques that are likely being used but are not being reported for some reason. There may also be cases where circumstantial information that a technique is in use exists but it's generally difficult for information to be collected or disseminated stating the technique is in use due to sensitivities related to the source of information or method of collection. Discretion is used based on the credibility of the source.
- **Unreported** – There is no public or non-public source of intel saying a technique is in use. This category may contain new offensive research used by red teams that has been published, but in the wild use by adversary groups is unknown. Discretion is used based on the utility of the technique and likelihood of use.

4.3.1.5 Technique Distinction

Several factors are considered when including new information to determine where and how it fits into the model:

- **Objective-** What the technique is accomplishing. Similar techniques may be performed the same way to accomplish different tactics. Likewise, different techniques may accomplish the same tactic in different ways.
- **Actions-** How a technique is performed. Is the "trigger" different between techniques that distinguishes them even though the result may be the same or similar?
- **Use-** Who is using it? Are there multiple groups? If so, how is the use different or the same?
- **Requirements-** The components that are needed to use a technique, or are affected by use of a technique. For example, files, locations, registry changes, API calls, permissions, etc. What is the overlap of components between the techniques? Are they distinct or similar?
- **Detection-** What needs to be instrumented to detect use of the technique? This is related to requirements and actions but could differ across techniques that are related.
- **Mitigations-** What mitigation options available for the technique? Are they similar to or different from other techniques that are either performed in the same way or have the same result?

Some techniques are breakouts of more general methods.. For example, PowerShell is a subset of scripting, but there are other scripting mechanisms that can be used and would need a place to be defined. PowerShell was broken out separately because it is a very prevalent method of

scripting and execution used by many adversary groups. It also has separately defined logging mechanisms and defenses built around it.

4.3.2 Creating New Techniques

When a potential new technique is identified, there are two possible approaches to including it in ATT&CK:

- Adding an entirely new technique, or
- Enhancing or abstracting an existing technique to make it inclusive of the newly-identified or otherwise previously uncategorized behavior.

This choice is not always clear – the following questions help guide the decision:

- What tactic does the technique fall under? Do multiple tactics apply?
    - Within a tactic, are other techniques similar to this one?
        - If so, how are they similar?
        - Is the similarity enough to categorize them together?
    - Does the empirical use reference support the tactic use?
        - Is it plausible that the technique can be used for that tactic objective even if data is unavailable due to related techniques?
- For similar techniques:
    - How is the technique performed? Is it similar in execution to other techniques? How many different ways can it be performed with existing utilities, adversary malware, and other tools?
        - Would a red or adversary emulation team conceptually group this technique with others or treat it separately?
    - Does the new technique have a different detection method or set of methods than the existing technique?
        - Are there similar data sources or methods for creating analytics that are similar or different than existing techniques?
    - Does the new technique have a different mitigation method or set of methods than the existing technique?
        - Is the implementation or deployment methods of the mitigation fundamentally different than existing techniques that can be inhibited by a similar mitigation?
    - Would creating a new technique be useful for an end user of the model?
        - Would defenders conceptually group this technique with others or treat it separately?

4.3.3 Enhancing Existing Techniques

If a technique is not conceptually different in how it is implemented or defended against, then it likely should be included in an existing technique as a variation. Further questions to consider when adding new information to an existing technique:

- What distinguishes this variation from existing methods of using the technique?
    - How is it performed?
    - What analytic differences, if any, may be necessary to effectively detect use of or system and network side artifacts resulting from the technique being used?
    - Are there different considerations for mitigation?

4.3.4 Named Adversary Groups Using Techniques

It is also important to consider adversary group usage of and variations to techniques to determine how they should be properly documented. These factors may also contribute to whether or not a new technique is created or an existing one enhanced.

- Are there different adversary groups that use this technique?
    - If so, how is it different?
    - Are the differences distinguishing characteristics of that group?
    - Should the differences be documented in the adversary group's profile for how they have been known to implement the technique?

4.3.5 Incorporation Threat Intelligence on Groups and Software within ATT&CK

Information about groups is derived from open source reporting, and each of the techniques used should have a reference to the source that explains how the group uses it. ATT&CK is based upon open source references to ensure the traceability of information and allow users to evaluate information sources.

Sources should be known to be reputable within the cybersecurity community and demonstrate intelligence analysis best practices. Common sources include security vendor blogs, but other sources such as personal blogs or Twitter may be used provided the information is deemed to be reliable. Original sources should be used whenever possible as opposed to secondary reporting about sources.

Examples from publicly-available threat reporting sources are deemed to be reliable based on widely accepted criteria for evaluating information, including:

1. Is the source internally and externally consistent?
2. Is the source known to have reported reliably in the past?
3. Is the source widely used, respected, and referenced by cybersecurity analysts in the community?
4. Does the source contain spelling or grammatical errors?

5. Does the source demonstrate sound analysis methodology (including stating supporting evidence, confidence levels, and gaps)? Does it include analytic "leaps"?
6. Do other sources corroborate information provided?

When documenting techniques used, multiple techniques may simultaneously apply to the same behavior. For example, Command and Control traffic over HTTP port 80 would fall under both the Commonly Used Port and Standard Application Layer Protocol techniques. This is to capture the various technical aspects of a technique and relate them to specific reasons they are used and what data sources and countermeasures can be used by defenders. Analysts should also use caution and not assume a technique was used if it is not explicitly stated or could not have happened in any other way during the reported incident. In the same example, if Command and Control traffic is over HTTP, unless explicitly stated or known, an analyst should not assume the traffic is over port 80 because adversaries may use uncommon ports (designated by Uncommonly Used Port).

Some groups in ATT&CK have multiple names associated with the same set of activities due to various organizations tracking the same (or similar) set of activities by different names. Organizations' group definitions may be only partially overlapping and may disagree on specific activity. There could be several nuances that lead an analyst and organization to categorize adversary activity separately. [12] Despite this challenge, tracking aliases for similar activity is useful to many users of ATT&CK, so the group pages make a best effort to track aliases based on public reporting. Similar to how techniques used must be cited, each alias also must be cited. There could be additional information, or analysis based on incomplete or unavailable data, that may lead to changes in how adversary groups are categorized.

Techniques used by a group should focus on those techniques believed to have been directly performed by adversaries, not those performed through use of a specific software sample. Techniques performed via software should be listed under the appropriate software page, and that software then linked back to the group having used it using the relationship/field noted above.

4.3.5.1 Ungrouped Use of Techniques

Reports often include adversarial behavior and technique use for ungrouped or unnamed activity. This is still a very useful source of information. Just because activity is not correlated to a named group does not mean it should not be included as justification for a technique or enhancing information. Typically, this information is included as a reference within the technical section of a technique describing instances of how the technique may be used.

4.3.6 Examples of Applying the Methodology for New Techniques

This section considers two separate techniques – Process Injection and SQL Injection – and steps through the methodology described above to illustrate when and how to add new techniques to the ATT&CK knowledge base.

Process Injection – Analysis of a technique that exists within ATT&CK by applying the above methodology. Process Injection, sometimes referred to as DLL injection, is a class of behavior that describes how an adversary can use an existing benign, running process as a way to hide the presence of their code executing.

Considerations:

- This technique is used to hide from some common defenses, like process tree analysis. It also could be used to execute within a certain context of another process that has certain user rights or permissions.
- It applies to Windows-based systems and represents benign functionality used by legitimate software that can be used by adversaries for malicious purposes.
- It requires real-time telemetry from the system on running processes and interactions with processes through the Windows API to effectively detect effective use. Some forensic detection of process injection is possible from loaded libraries and other data sources but requires proper timing.
- Mitigation is difficult due to its benign usefulness in software. Some security features may mitigate aspects of this technique, such as application whitelisting that includes analysis of loaded modules, or code integrity that prevents processes from a lower integrity level from interfacing with processes running in at a higher integrity level.
- Many adversary groups use this technique, which is a component of tools, scripts, and malware.
- There are a few variations of process injection, but most follow a common sequence of an initial adversary controlled process requesting access to a non-malicious process, loading code within it, and forcing that process to execute the new code.
- Some variations load DLLs from disk, while others perform reflective loading that do not require a file on disk.
- Related methods of execution require a binary to be put on disk and/or some configuration change that will load and execute the code in a new process representing different opportunities to detect and mitigate.
- Other related methods use different functionality provided by Windows to load and execute code, such as application shims.
- Similar concepts exist in Linux based systems for dynamically loading libraries into processes.

Conclusions:

- The core feature of this technique is loading malicious code within an existing live process.
- The technique is used widely across many groups of adversaries.
- Even though there are a few variations of this technique, the core behavior is distinct enough from other related methods of defense evasion and privilege escalation to warrant an individual entry.
- There are a few variations within this core concept to include in the process injection entry.

- Process injection should be included as an individual technique under defense evasion and privilege escalation. [11]

SQL Injection (SQLi) – an example analysis of a technique that is not explicitly in ATT&CK by applying the above methodology.

SQLi is a method of injecting code through an improperly secured web interface that is interpreted and executed by a database process. The resulting code execution can be used for a number of purposes, including adding or modifying information, gaining access to a system, causing the server to download and execute other code which may result in persistence, credential access, privilege escalation, collection, and exfiltration.

Considerations:

- SQLi may be performed to gain access to an externally facing web server in a DMZ or improperly positioned web server that would result in network compromise. It may also be performed to achieve lateral movement within an enterprise, but in-the-wild reported incidents have been scarce on this use case.
- Fundamentally, SQLi is exploiting a vulnerability in web application software due to poor code design and is not a benign behavior that an adversary could use for some purpose.
- SQLi is a predominant vulnerability that occurs frequently across many different types of web applications, regardless of language or platform they are written in.
- Software has been developed to automate SQLi; it is unlikely that this would be performed manually.
- For the external variation, data sources collecting traffic on the boundary would likely see this behavior. Application logs from the web and database server may be used as well. True positive detection may be difficult due to certain variance that can be used in frequency and timing of attempts and methods to hide indicators.
- For the internal variation, tools that may not normally be present within an enterprise network would likely need to be downloaded and used by an adversary. Depending on the tool and how it is used, it may create an enormous amount of traffic against an internally accessible web server. Internal netflow, packet capture, web logs, and endpoint monitoring may be used to detect aspects of the download and usage of the tool.
- There are many methods on how SQLi may reach a database through various malformed data inputs and parameters. How they are detected or mitigated are not fundamentally different from each other. Database input or web logs can be used to look for common SQLi inputs that result in code execution. Likewise, using secure web development and existing secure programming constructs mitigates a large number of SQLi instances.
- Adversaries have been known to use SQLi as a means of gaining access to externally available web servers. There is not good data available on use within internal networks for other purposes.

Conclusions:

- The context in which SQLi fits within an adversary's tactical goals puts it within attempts to gain access to a system through an existing software vulnerability. An example is for initial access in a network compromise by compromising an externally facing application.
- SQLi is a variation of an exploitation technique against a specific software technology and is an appropriate abstraction within how an adversary performs initial compromise. It would not need to be described in various ways at this technique level due to the limited variations in how it is performed by an adversary, detected by defenders, or mitigated through proper software design. Additional resources can be cited as needed, such as CAPEC, CWE, OWASP that detail specifics.
- Include SQLi in ATT&CK as a technical detail enhancement of Exploit Public-Facing Application for gaining access to exposed web servers or databases. [13]

5 Summary

This paper discussed the motivation behind the creation of ATT&CK, the components described within it, its design philosophy, how the project has progressed, and how it can be used. It is meant to be used as an authoritative source of information about ATT&CK, as well as to help guide how ATT&CK is maintained and how ATT&CK-based knowledge bases are created for new technology-domains and platforms.

Adoption of ATT&CK is widespread across multiple disciplines, including intrusion detection, threat hunting, security engineering, threat intelligence, red teaming, and risk management. It is important for MITRE to strive for transparency about how ATT&CK was created and the decision process that is used to maintain it, as more organizations use ATT&CK. We want users of ATT&CK to have confidence in the information and resources that it can provide and better understand how they can begin to use it—and also how and where they can help ATT&CK grow.

The types of information that went into ATT&CK, and the process used to create and maintain it, may also be useful for other work to derive similar models for other technology domains or for taxonomies of adversarial behavior in other areas. ATT&CK's grounding with empirically driven threat information and its driving use cases for adversary emulation and better measurement of defensive coverage were foundational in how it was perceived and used across the security community. We hope this document can be a useful resource for efforts seeking to follow the process used to create ATT&CK for new areas.

6 References

[1] J. Wunder, "The MITRE Corporation," 5 January 2018. [Online]. Available: https://www.mitre.org/capabilities/cybersecurity/overview/cybersecurity-blog/whats-next-for-attck%E2%84%A2. [Accessed 5 January 2018].

[2] The MITRE Corporation, "Common Attack Pattern Enumeration and Classification," 21 February 2018. [Online]. Available: https://capec.mitre.org/. [Accessed 12 April 2018].

[3] "Common Weakness Enumeration," 3 April 2018. [Online]. Available: https://cwe.mitre.org/. [Accessed 12 April 2018].

[4] B. Strom, J. Battaglia, M. Kemmerer, W. Kupersanin, D. Miller, C. Wampler, S. Whitley and R. Wolf, "The MITRE Corporation," June 2017. [Online]. Available: https://www.mitre.org/publications/technical-papers/finding-cyber-threats-with-attck-based-analytics. [Accessed 14 November 2017].

[5] C. Betz, S. Caltagirone and A. Pendergast, "The Diamond Model of Intrusion Analysis," 2013. [Online]. Available: http://www.activeresponse.org/wp-content/uploads/2013/07/diamond.pdf. [Accessed 16 January 2018].

[6] FIRST, "Common Vulnerability Scoring System v3.0: Specification Document," 2018. [Online]. Available: https://www.first.org/cvss/specification-document. [Accessed 20 December 2017].

[7] D. Leblac, "DREADFul," 14 August 2007. [Online]. Available: https://blogs.msdn.microsoft.com/david_leblanc/2007/08/14/dreadful/. [Accessed 20 December 2017].

[8] The MITRE Corporation, "Credential Dumping," 12 April 2018. [Online]. Available: https://attack.mitre.org/wiki/Technique/T1003. [Accessed 12 April 2018].

[9] The MITRE Corporation, "Rundll32," 5 March 2018. [Online]. Available: https://attack.mitre.org/wiki/Technique/T1085. [Accessed 4 April 2018].

[10] The MITRE Corporation, "Obfuscated Files and Information," 10 April 2018. [Online]. Available: https://attack.mitre.org/wiki/Technique/T1027. [Accessed 12 April 2018].

[11] The MITRE Corporation, "Process Injection," 18 January 2018. [Online]. Available: https://attack.mitre.org/wiki/Technique/T1055. [Accessed 24 April 2018].

[12] F. Roth, "The Newcomer's Guide to Cyber Threat Actor Naming," 25 March 2018. [Online]. Available: https://medium.com/@cyb3rops/the-newcomers-guide-to-cyber-threat-actor-naming-7428e18ee263. [Accessed 4 April 2018].

[13] The MITRE Corporation, "Exploit Public-Facing Application," 11 April 2018. [Online]. Available: https://attack.mitre.org/wiki/Technique/T1190. [Accessed 24 April 2018].