

**CYBER NATIONAL MISSION FORCE
TACTICS, TECHNIQUES, AND PROCEDURES (TTP)
BASED HUNTING METHODOLOGY**



**Prepared for the Cyber National Mission Force
Prepared by the Joint Cyber Warfare Center**

Executive Summary

Attempts to detect malicious activity through signatures of easily-changed attributes such as Internet Protocol (IP) addresses, domains, or hashes of files, are brittle and quickly become outdated. This approach is often referred to as signature-based detection, though tool-driven detection is more accurate. Operational assessments provide ample evidence that this approach is ineffective against adaptable threats. This is because malicious actors easily and frequently change those attributes to avoid detection.

Anomaly-based detection employs statistical analysis, machine learning, and other forms of "big data" analysis to distinguish malicious behavior from normal system activity. This approach has traditionally suffered from high false positive rates, can require significant investment in large scale data collection and processing, and does not always provide enough contextual information around why something was flagged as suspicious, which can make the tuning of analytics challenging.

A growing body of evidence from industry, MITRE, and Joint Cyber Warfare Center (JCWC) experimentation confirms that collecting and filtering data based on knowledge of malicious actor¹ tactics, techniques, and procedures (TTPs) is an effective method for detecting malicious activity. JCWC believes that employing this approach will give the defense a significant and rare (to date) advantage over offense in cyberspace operations, which will flip the balance of power in this domain. Target technology on which malicious actors operate constrains the number and types of techniques they can use to accomplish their goals post-compromise. There are a relatively small number of these techniques, and they occur on systems owned by the victim organization. All malicious actors must either employ these known techniques or expend vast resources to develop novel techniques regardless of their capabilities or higher-level mission objectives. This paper expands on existing best practices in detecting malicious actors through these techniques and describes the requirements to succeed in this approach.

These three approaches are not mutually exclusive. Signature-based, anomaly-based and TTP-based detection are complementary approaches to one another. However, the relative costs and effectiveness of each approach dictate a significant shift in how these approaches are employed. Significant resources should be allocated to use TTP-driven hunting, as doing so will significantly improve hunting effectiveness.

Based on existing best practices and JCWC experimentation, JCWC recommends changes in the following 4 areas: operations, intelligence, training, and capabilities. Operations should be structured around the approach described in this document. Data should be collected and analyzed continuously, leveraging partnerships between Cyber Security Service Providers, Service Cyber Protection Teams and National Cyber Protection Teams. Intelligence support to Defensive Cyberspace Operations (DCO) should focus on collecting timely and actionable intelligence on the techniques malicious actors must employ across all cyber terrains of interest to the Department of Defense (DoD). An organization should be tasked to establish and maintain a detailed description of Offensive Cyberspace Operations (OCO) scheme of maneuver and a

¹ In this paper, "adversary" and "malicious actor" will be used synonymously, consistent with the use of the term "adversary" in referenced literature, recognizing that in some contexts outside of this paper, those terms are not used interchangeably.

description of how to detect all relevant techniques. This information must be made available at an actionable (unclassified) level for defensive forces. The Cyber Mission Forces (CMF) should be trained in this approach, including training on malicious techniques, data collection requirements, and hypothesis-driven analysis. Capabilities should be deployed across the DoD Information Network (DODIN) to collect the kind of data required to support this approach.

Acknowledgements

The content in this paper was developed by the DOD FFRDC within the MITRE Corporation specifically under the direction and sponsorship of the Cyber National Mission Force HQ and the USCYBERCOM. The sponsor's intent is to make this information broadly available to the maximum extent possible to all of the Cyber Mission Forces and any other cyber defenders within the DOD. It is expected that this information will be subsequently republished by the sponsor in other forms such as administrative or operational orders, directives, guidance and/or policy, as appropriate.

Table of Contents

| | |
|-------------------------------------------------------------------------------------|----|
| 1 Introduction | 3 |
| 1.1 Definition of Hunting..... | 3 |
| 1.2 Analysis Space | 3 |
| 1.3 Traditional Detection and Prevention Methods | 5 |
| 1.3.1 Whitelisting and Blacklisting..... | 5 |
| 1.3.2 Indicators of Compromise and Network Security Monitoring | 5 |
| 1.3.3 Anomaly-Based Detection | 6 |
| 1.3.4 Host-Based Event Data | 6 |
| 1.4 TTP-Based Detection..... | 6 |
| 1.5 Comparison of Published Methodologies | 7 |
| 2 Methodology | 8 |
| 2.1 Overview | 8 |
| 2.2 Characterization of Malicious Activity (Left Side of the “V”)..... | 9 |
| 2.2.1 Gather Data and Develop Malicious Activity Model | 9 |
| 2.2.2 Develop Hypotheses and Abstract Analytics..... | 10 |
| 2.2.3 Determine Data Requirements | 10 |
| 2.3 Filter..... | 12 |
| 2.4 Execution Phase (Right Side of the ‘V’)..... | 13 |
| 2.4.1 Identify and Mitigate Collection Gaps..... | 13 |
| 2.4.1.1 Confirm Existing Data Sources – Presence and Validity | 13 |
| 2.4.1.2 Deploy Required New Sensors to Fill Gaps with Existing Collected Data | 14 |
| 2.4.1.3 Addressing Collection Gaps | 14 |
| 2.4.2 Implement and Test Analytics | 15 |
| 2.4.3 Hunt: Detect Malicious Activity and Investigate..... | 15 |
| 2.4.3.1 Tune analytic(s) for initial detection | 17 |
| 2.4.3.2 Evaluate Hits | 19 |
| 2.4.3.3 Possible Causes of False (Benign) Hits | 19 |
| 2.4.3.4 Document Malicious Hits | 21 |
| 2.4.3.5 Gather Contextual Information | 22 |
| 2.4.3.6 Investigate Malicious Hits | 23 |
| 2.4.3.8 Impose Cost | 25 |
| 2.5 Report..... | 25 |
| 3 Implications and Future Work | 26 |
| 3.1 Implications for Operations | 26 |

| | |
|--------------------------------------------------|----|
| 3.2 Implications for Intelligence | 27 |
| 3.3 Implications for Workforce Development | 27 |
| 3.4 Implications for Capabilities | 28 |
| 3.5 Future research..... | 28 |
| 4 References/Bibliography..... | 30 |

List of Figures

| | |
|------------------------------------------------------------|----|
| Figure 1 Analysis Space..... | 4 |
| Figure 2 Methodology “V” Diagram | 9 |
| Figure 3 Context vs. Volume of Host and Network Data | 11 |
| Figure 4 CAR Data Model..... | 12 |
| Figure 5 General Hunt Process Flow | 16 |
| Figure 6 Heat Map | 17 |

1 Introduction

This paper builds upon a growing body of evidence from the cyber security community to present a robust and successful approach to detecting malicious activity based on an understanding of malicious actors' tactics, techniques, and procedures (TTP) in cyberspace. It attempts to show that, by describing adversarial activity *at the right level of abstraction*, appropriate sensors (host and network-based) can be deployed and analytics can be designed to detect adversaries with high accuracy, while being robust to modifications in particular implementations. The approach presented, TTP-based hunting, is complementary to existing practices such as using indicators of compromise (IOCs) or using statistical analysis of data to detect anomalies. This paper makes recommendations for how the Cyber Mission Forces (CMF) can implement that approach.

1.1 Definition of Hunting

The word "hunting" is an emerging term within cybersecurity for which the exact definition is still evolving. In the 2017 Threat Hunting Survey, the SysAdmin, Audit, Network, and Security (SANS) Institute (Lee & Lee, 2017) defines threat hunting as, "a focused and iterative approach to searching out, identifying and understanding adversaries that have entered the defender's networks." Sqrrl (2016) defines threat hunting as, "... the process of proactively and iteratively searching through networks to detect and isolate advanced threats that evade existing security solutions." Endgame defines hunting as, "the process of proactively looking for signs of malicious activity within enterprise networks without prior knowledge of those signs, then ensuring that the malicious activity is removed from your systems and networks." (Scarfone, 2016, p. 1). For this paper, "hunting" is defined as the proactive detection and investigation of malicious activity within a target network.

1.2 Analysis Space

Malicious activity in cyberspace can be considered in three dimensions: time, terrain and behavior. Every event in cyberspace can be represented as a specific behavior (benign, malicious or suspicious) at a specific time, on a specific machine, process, subnet, or other element of cyber terrain. Each of these dimensions is described below. Figure 1 provides a visualization of these three dimensions with example values for behavior and terrain types.

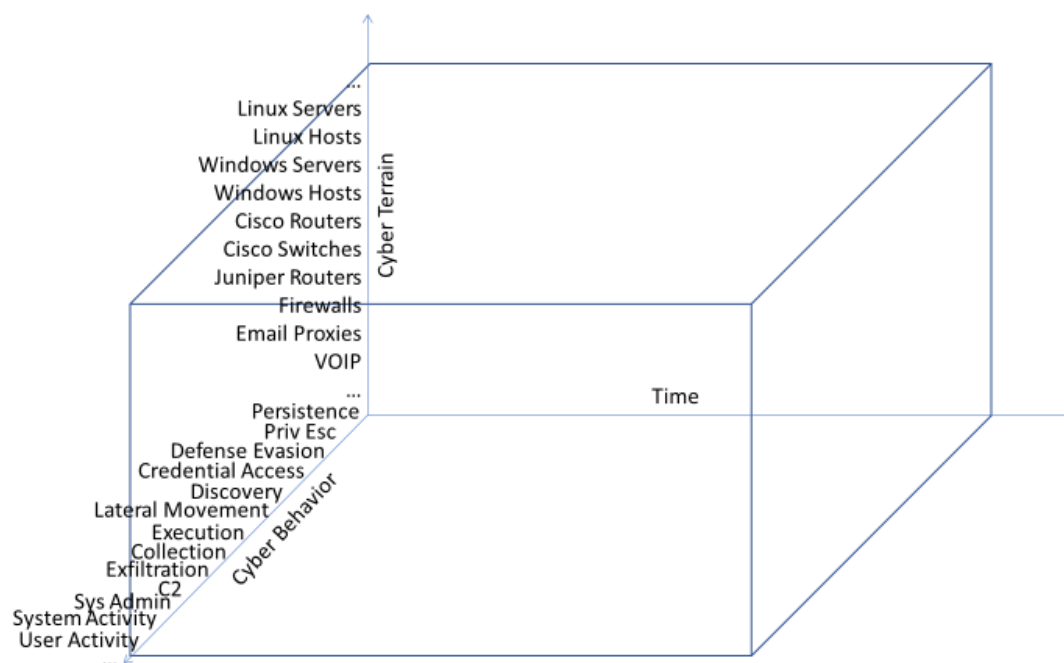


Figure 1 Analysis Space

The time dimension is relatively straightforward. Most evidence of malicious actor activity is transient in nature (process content and behavior, for example) and must be collected during the intrusion as it cannot be obtained after the fact. Data from continuous monitoring is generally preferred over forensic data collection because hunting a malicious actor starts with a large, unknown window of time and most forensic data (and its related data collection capabilities) only cover a narrow slice of the time domain for these transient events. It is far more likely that malicious actor activity will fall outside of this slice, thus these tools are less effective for hunting. If used as part of an ongoing investigation, forensic tools can complement other data sources but are much less effective as primary data sources for detection.

By terrain, we mean cyberspace itself, to include any host, network segment, or other area where the malicious actor is currently operating. For the purposes of this methodology, terrain is restricted to where defenders have authority to operate and a responsibility to defend – within an enterprise or enclave monitored by a Security Operations Center (SOC) or hunt team. Particular focus should be paid to areas that might be highly targeted by a malicious actor (e.g., “crown jewels”) (MITRE, 2018); areas that the malicious actor may need to traverse to complete their objective (Internet access points, Trusted Internet Connections, Domain Controllers, etc.); or areas that, if damaged, will hamper defensive forces in countering the intrusion (SOC analysis systems, perimeter and host sensors, log collection architecture, etc.). What constitutes key terrain to monitor for the purposes of hunting a malicious actor is an open question that will require more research.

Behavior refers to malicious activities in cyberspace. Data should be collected to observe those activities. For example, if the malicious actor can launch malware from ordinarily benign

processes, defenders should capture data on process launches and the process's parent information from hosts within the terrain of interest. Another example is encryption of malicious command and control (C2) communications across the network. Collecting network communication information from the host may allow defenders to potentially mitigate this visibility gap.

1.3 Traditional Detection and Prevention Methods

Traditional detection and prevention approaches include whitelisting and blacklisting; sweeping for IOCs and network security monitoring (NSM); and anomaly detection – with the majority of data collection focused on network sensors and perimeter proxies. Each of those approaches has benefits and limitations.

1.3.1 Whitelisting and Blacklisting

Whitelisting can often be too restrictive or impractical to implement. The rapid modification of technology along with the unpredictable nature of human behavior makes a whitelist very difficult to define and keep updated. Blacklisting can be brittle to subtle changes in adversary tradecraft implementation. A signature written to detect IP addresses, domains, file hashes, or filenames associated with malicious activity, without triggering on benign instances, is often very brittle to polymorphism, metamorphism and other implementation modifications which are relatively cheap for an adversary to use. David Bianco captured this through his “Pyramid of Pain” (FireEye, 2014). Defining those brittle signatures and indicators often requires extensive resources, through reverse engineering and static analysis, and are often dependent on detection through some other means (often after having been successfully used on by adversaries in other breaches and independently detected, reported, and disseminated).

1.3.2 Indicators of Compromise and Network Security Monitoring

Prior to 2016, threat hunting processes appear to have been primarily organized around sweeping for IOCs; which include static characteristics of malware, such as hashes, filenames, libraries, and strings; or similar signatures (e.g., Yara rules), gathering disk and memory forensics artifacts, and analyzing them for anomalous activity that might indicate adversary activity.

Traditionally, continuous activity monitoring has primarily focused on collecting and analyzing network traffic, usually focused at perimeter boundaries, as part of a Network Security Monitoring (NSM)-focused defensive operation. The historical focus on network perimeter monitoring developed in part from the convenience and cost efficiency of placing a limited set of sensors at a relatively small number of heavily controlled network gateways.

However, each of these approaches fail to provide sufficient visibility into adversary behavior to support comprehensive TTP-based hunting activities. Indicator sweeping fails to identify novel or changing threats that don't match known indicators, and only provides detection capabilities after the fact. Network perimeter sensors provide little to no insight into adversary activity outside of the initial successful breach and data exfiltration stages, including especially lateral movement and privilege escalation within the compromised environment. Network sensors deployed within an environment may assist in mitigating some of these shortfalls but it can be

difficult to deploy enough network sensors to comprehensively monitor any but the smallest enterprises. Used correctly, however, internal network sensing remains an important component of an enterprises defenses and can complement host-based sensing for TTP-based detection.

1.3.3 Anomaly-Based Detection

Anomaly-based detection employs statistical analysis, machine learning, and other forms of "big data" analysis to distinguish malicious behavior from normal system activity. This approach has traditionally suffered from high false positive rates, can require significant investment in large scale data collection and processing, and does not always provide enough contextual information around why something was flagged as suspicious, which can make the tuning of analytics challenging.

Anomaly detection approaches often result in too many false positives. The benign activity of software, system administrators, software developers and everyday users across enterprise networks is often so variable across time, users, and network space, that defining "normal" behavior can be a futile exercise.² The volume of data required to be processed for anomaly and statistical analysis can be prohibitive to collect and retain. There must be sufficient data collected, from a sufficient number of data sources and locations within an environment, to enable trend and statistical analysis. Given that what constitutes "sufficient" in this context is poorly defined makes this type of detection harder to utilize and hard to measure effectively.

1.3.4 Host-Based Event Data

Host event data collection has generally been identified as the most desired data source for hunting (Lee & Lee, 2017, p. 16) but "many respondents feel that endpoint data is more obscure and harder to obtain" (Lee & Lee, 2017, p. 17). Recent advances in operating system capabilities (e.g. Windows 10 event tracing and event forwarding; auditd and the integrity measurement architecture (IMA) on Linux) show that host-based data of sufficient granularity and abstraction is increasingly available. There is a pervasive notion in the community that endpoint data is too voluminous to collect and analyze. However, research into malicious file detection conducted by Invincia Labs (Berlin, Saxe, & Slater, 2015) suggests that 100-200 megabytes of audit data from Windows workstations per day was sufficient to detect 85% of the malware executed on a system. While this experiment was not conducted via a threat hunting effort but rather a machine learning one, it suggests an achievable target for scoping the volume of data collection and storage requirements.

1.4 TTP-Based Detection

Rather than characterizing and searching for tools and artifacts, a more robust approach is to characterize and search for the techniques actors must use to achieve their goals. These techniques do not change frequently, and are common across actors due to the constraints of the target technology. Models such as the MITRE Adversarial Tactics, Techniques & Common

² https://www.utdallas.edu/~muratek/courses/dmsec_files/oakland10-ml.pdf

Knowledge (ATT&CK) framework³ or the Office of the Director of National Intelligence (ODNI) Cyber Threat Framework⁴ are effective ways to characterize those techniques. These models focus on identifying all adversary TTPs and align them within the phases of the Cyber Attack Lifecycle.

These models have been found to be very useful in defensive operations, helping to identify new adversary behaviors, helping prioritize detection for techniques utilized by multiple actors, and, in conjunction with data modeling, allows for identification of visibility and defensive capability gaps that orient around the threat to an organization. It allows defenders to frame detection hypotheses, focused on the adversary's actions and phases of their operations, that lend themselves to specific, implementable analytics within the defender's analysis platform (e.g. a security information and event management (SIEM) or other data analysis system).

A good data model for hunting will relate what objects and actions an analyst wishes to capture to the key data (fields, values, attributes) needed from the environment's sensors. It ties the data of what the sensor can observe to the actions and events the analytics are meant to identify and detect (see section 2.2.3 *Determine Data Requirements* for more information). One example of data modeling useful for hunting is the Cyber Analytic Repository (CAR) data model, which attempts to describe adversary actions in terms of the data required to identify those actions, irrespective of a specific tool or product (MITRE, 2015b).

The methodology proposed in this paper will utilize the ATT&CK framework, in conjunction with the CAR data model, as an example of a generic adversary model that attempts to identify all possible adversary behaviors to analyze and detect. Other frameworks may be used if they satisfy the requirements of identifying specific adversary TTPs that can be decomposed into actionable analytics within an analysis platform.

1.5 Comparison of Published Methodologies

According to SANS (2017), few organizations utilize an existing hunting methodology, citing a lack of published or accessible methodologies. Through literature review, this appears to still be correct. Sqrrl has published a hunt methodology, described in the *A Framework for Hunting* whitepaper released in 2016, and a maturity model. Endgame has published a hunting process, titled *The Hunt Cycle*. (Scarfone, 2016, p.9) Beyond those published models, it appears that most organizations' hunting methods are defined internally (27.1%) or consist of ad-hoc processes without a documented methodology (45.1%). (Lee & Lee, 2017)

Sqrrl's methodology is organized around four phases: 1) Create hypotheses; 2) Investigate via Tools and Techniques; 3) Uncover New Patterns and TTPs; and 4) Inform and Enrich Analytics. They describe, in general terms, that one should start hunting with a hypothesis, based around proposed adversary activity one wishes to hunt for or around a portion of the environment one suspects the adversary may be operating within. They suggest choosing a hypothesis based on available intelligence about the adversary, and then selecting tools and techniques to find that

³ https://attack.mitre.org/wiki/Main_Page and Strom (2017)

⁴ <https://www.dni.gov/index.php/cyber-threat-framework>

behavior. However, they limit the description of these phases to general discussion and do not describe in detail what data to use, why one would use it, how to model it, or how to modify or tune analytics. There is no discussion about how to structure intelligence or the TTPs and hypotheses one wishes to hunt for in a new or existing adversary model, such as the ATT&CK matrix (released 2015) or the ODNI Cyber Threat Framework (released 2017). Sqrrl's methodology does extend further than hunting activities, as it also describes some capabilities an organization may require or skillsets that the team should have to improve the organization's hunting maturity. They organize this in a notional Hunting Maturity Model.

Endgame's methodology, as described in *The Hunter's Handbook* eBook, is also organized around four phases: 1) Survey (a selected portion of the environment); 2) Secure (the hunted environment); 3) Detect (the adversary); and 4) Respond (and remediate the intrusion). The initial focus on gaining a better understanding of the environment, via Survey, is relatively novel in the literature, yet reflects general understanding among experienced hunting practitioners that the results of an organization's early hunting attempts will be focused around a better understanding of the environment rather than actually identifying an undetected adversary. Endgame then suggests securing the hunting environment by locking down lateral movement capabilities. As the methodology does not describe how to do this, one must assume that their methods assume Endgame products are being utilized. Again, as with Sqrrl, there is no discussion about how to organize hunting hypotheses or intelligence using an adversary model. Nor is there a description about how to model data to inform detection and data gaps.

The methodology described in this paper differs from these methodologies by attempting to create a general hunting methodology focused on identifying adversary behavior, structured within an adversary model, that does not depend on specific tools or products but rather describes what data is necessary, what types of data should be available from sensors, and how to utilize that data with analytics to conduct a hunt.

2 Methodology

2.1 Overview

The approach to hunting described below has two components: *Characterization* of malicious activity, and hunt *Execution*. These components should be ongoing activities, continuously updated based on new information about adversaries and terrain. The flow of updates is visualized in the *V Diagram* below (Figure 2). There are three layers of related activities, focusing on the malicious activity, analytic processes, and data processes. Examples given in the following sections are described using a notional hunt team, composed of individual analysts and a team lead, to illustrate key points.

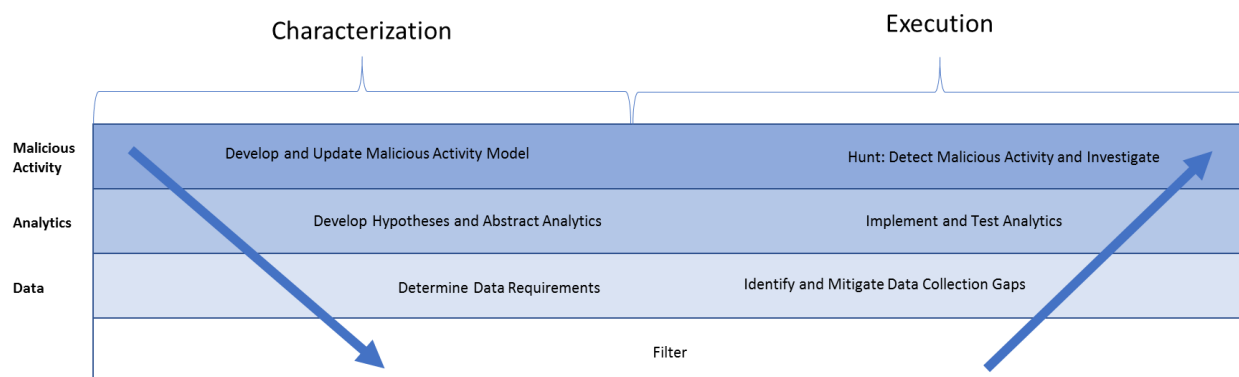


Figure 2 Methodology “V” Diagram

Characterization of malicious activity starts with developing or updating the generic adversary model of behavior to identify all TTPs that an adversary may use – regardless of actor group, environment, or target. For each TTP identified in the model, an analyst proposes one or more detection hypotheses that are formulated as abstract analytics. These hypotheses and abstract analytics are used to determine what data is necessary to collect. For each hunting operation, the hunt team should filter these data collection requirements and analytics based on the specifics of the terrain and situation of that hunt.

Execution employs the filtered data requirements and data model to conduct a gap analysis of sensors and data sources within the environment. If necessary, additional sensors (network or host-based) may be deployed at this stage to address visibility gaps. Once data is flowing into the analysis system, the analyst leverages the data model to implement analytics within the analysis system. The hunt team then executes the hunt by selecting specific analytics strongly associated with malicious behavior to try and obtain an initial detection. Analytic tuning and triaging suspicious and correlated events to positively identify the presence of an adversary follows this initial detection.

Each of these steps are described in greater detail below.

2.2 Characterization of Malicious Activity (Left Side of the “V”)

2.2.1 Gather Data and Develop Malicious Activity Model

Through intelligence collection (including but not limited to the National Security Agency (NSA), USCYBERCOM, and the commercial sector contributors), threat information sharing by other organizations (e.g., FireEye reports, MITRE ATT&CK framework), and research efforts, the defensive operations community collects information on how adversaries behave across various terrain types (e.g., Windows Enterprise Networks, ICS/SCADA systems, weapons systems, infrastructure devices, mobile devices, Internet of Things (IoT) devices, etc.). It is important during this analysis to consider which aspects of adversarial behavior are transient, or easy for the adversary to change or mask, and which aspects of behavior are likely to remain constant or prove difficult for the adversary to change (e.g., TTPs). The focus is on information

that can be converted into TTP-based analytics rather than brittle indications of compromise such as file hashes, IP addresses or domain names (i.e. focus on the top of David Bianco's "Pyramid of Pain"). This information needs to be organized in such a way as to facilitate filtering by dimensions in the analysis space (time, terrain, behavior), by the adversary, or the phase of the adversary's operation (e.g. Cyber Attack Lifecycle or ATT&CK Tactic category).

2.2.2 Develop Hypotheses and Abstract Analytics

Based on this knowledge of adversarial behavior, the analyst proposes a hypothesis to detect that behavior in the form of an abstract analytic. For example, knowing that adversaries sometimes move files between systems using Server Message Block (SMB) protocol, and then execute them using scheduled tasks (*schtasks*), an abstract analytic might be to detect when a *schtasks* execution occurs as a result of a file moved through an SMB session.

2.2.3 Determine Data Requirements

In order to hunt effectively, one needs to have data that adequately captures the activity of the adversary from data sources and sensors in the correct location (within the terrain) to successfully observe it. Specific data requirements for hunting fall into two broad categories: collection requirements and modeling requirements.

To identify collection requirements, a list of required data and data sources should be created based on the set of abstract analytics developed. For example, in the analytic described above, data collection needs might include capturing network traffic and host logs associated with SMB, and contextual data associated with any invocation of *schtasks* on each desktop in the enterprise (e.g. which file is executed, the date/time for execution and the user associated). As a result, comprehensive data requirements can be aggregated across all analytics; linking each data requirement and the terrain type, adversary and kill-chain phase associated with that requirement.

Sensor and data source selection play a key role. It is helpful to consider the merits of which sensor or data source to select based on the amount of contextual information they provide the analyst balanced against the volume of data generated by each data source. It is generally true that more context equates to more volume (network bandwidth utilized for collection; storage, indexing, and analysis resources for processing), so it is unlikely to be possible to capture all possible data (full content collection from hosts and network devices) to support a hunt. However, context is critical to effectively triage suspicious events and separate truly malicious activity from suspicious but ultimately benign activity.

The diagram in Figure 3 illustrates the relationship between the amount of contextual information present from a data source, the generalized amount of data generated by the data source, and whether the data source is primarily host-based or network-based. The higher the data source is on the chart, the more likely it is to be able to capture the context needed for hunting. A successful hunt will involve correlating information from multiple data sources in multiple locations (host and network) to create a comprehensive understanding of the activities taking place on the network.


| Data category | | Host Data | Network Data |
|---------------------------------------------------------------------------------------------------------------------------|-------------------------|-------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------|
| <div>High context High Volume</div>  | Full content | Instruction-level Tracing | Full packet capture |
| | Partial content | API and Syscall logging Auditd and application debug logging EDR data (events and content) | Bro network sensor Web and Email proxy (headers and content) |
| | Metadata / Session data | EDR events (Sysmon, Endgame, Carbon Black, etc.) Application logs Hostflows (netflow from host) | Web and Email proxy (headers only) Netflow |
| Low context Low volume | Summarized data | HIDS alerts (HBSS, McAfee, A/V) | NIDS Alerts (Snort, Sourcefire, etc.) Firewall "deny" logs |

Figure 3 Context vs. Volume of Host and Network Data

Sensors that provide activity, event, or content data are preferred over signature and alert-based ones, as the nature of hunting requires examining activity that was not initially detected as malicious. Many traditional sensors, such as signature-based host or network intrusion detection systems (HIDS/NIDS), focus on detecting very specific, discrete information present in an attack (usually some highly identifiable malware attribute). These types of sensors are generally not useful for hunting, as they are focused on automated detection of well-known malicious activity – there’s no hunting involved – and the data generated by these sensors is generally limited to specific alerts. They do not provide contextual event and activity data around any suspicious activity being investigated. Host sensor selection will likely involve some combination of endpoint detection and response (EDR) agents, application logs, and operating system event logs.

Data from host, network, and application proxy sensors must be specific enough to be able to ascertain what occurred on a host or network (on that part of the terrain) but not so specific that the amount of data generated is too large to feasibly collect, aggregate, and analyze. For instance, network flow data may have extremely limited value if it lacks application layer deep packet inspection. Similarly, host-based sensors that may initially appear adequate might later be found inadequate due to an inability to represent fine-grained process detail (e.g., parent process, execution path, command line arguments, etc.).

To help narrow the scope of what data is necessary, it helps to construct a data model. Good data model requirements for hunting will involve relating what objects and actions one wishes to capture to the key data (fields, values, attributes) needed from sensors. It ties the data of what the sensor can observe to the actions and events your analytics are meant to identify and detect.

For example, the CAR data model (MITRE, 2015b) uses an object:action pair to describe some specific adversary activity with the data fields linked to that object:action pair that help identify this activity. Figure 4 shows the process object within CAR, describing process activity that would be useful in detecting and tracking an adversary – in this case, process creation and termination events – and the data fields required. The hunt team can use the model to identify what information the sensors and data sources need to capture to enable analytics and the hunting mission.

| | | |
|---------|---------------------|-------------------|
| process | | command_line |
| | | exe |
| | create terminate | fqdn |
| | | hostname |
| | | image_path |
| | | md5_hash |
| | | parent_exe |
| | | parent_image_path |
| | | pid |
| | | ppid |
| | | sha1_hash |
| | | sha256_hash |
| | | sid |
| | | signer |
| | | user |

Figure 4 CAR Data Model

2.3 Filter

Upon completion of the *Characterization* phase, the team has a generic adversary model of all known TTPs; proposed hypotheses and abstract analytics defined to detect those TTPs; and data requirements and a data model necessary to enable those abstract analytics. Now the team needs to filter what they plan to analyze to hunt the adversary – essentially this is where the team initially constrains the analysis space to focus on what time, terrain, or behavior will enable them to start hunting the adversary.

Filtering on time is relatively straight-forward – the team may have information that indicates an adversary was operating within the target environment at a certain time, so the initial window should be bounded around that time period. Another possibility is to start from the present and

retrospectively look backwards for a finite period (e.g. from now to two weeks ago). Often, the time window is automatically constrained by the retention period of the data storage and analysis system (e.g. the SIEM contains logs that cover a rolling 30-day period).

Once the terrain in which to hunt is known or selected, the team can filter based on the types of systems and data available – if the environment is primarily Windows systems with some Linux servers, data requirements can be reduced to only data sources relevant to Windows and Linux systems. Given that the data requirements are tied to analytics that are then tied to TTPs, this automatically reduces the number of analytics necessary for the team to execute.

Lastly, the team can filter on behavior – specifically selecting which adversary TTPs to detect within the environment. There are numerous ways to accomplish this but two general approaches to use are: filter based on the likelihood that the TTP will be easily identified as malicious (i.e. “this TTP is not used by our system administrators as part of their work”), or on the likelihood of a specific adversary group known to target the environment using the TTP (i.e. “this TTP is associated with an adversary we’ve been attacked by”).

Filtering on ease of detection requires knowledge about what activities are normal within an environment and likely involve trial and error to reduce false positives. In a large enterprise network, there is significant variation in behaviors exhibited by users and system administrators in performing their duties, so what may appear abnormal initially may be benign (or “normal”) behavior for a particular user. Repeated hunting operations will help identify which behaviors are uncommon, and therefore more useful for detection, within an environment.

Filtering on which adversary groups are targeting an environment may be useful, with some caveats. It is unlikely that an organization or environment will be targeted by a single adversary group, so filtering down to just known behavior of that group may cause a hunt to miss the presence of another adversary that has successfully compromised the environment. Additionally, adversaries can, and often do, adapt as new TTPs are identified. Filtering only on TTPs that were previously identified as associated with the adversary may allow the adversary to escape detection. Therefore, this type of filtering is more likely to be beneficial in *prioritizing* which TTPs to search for first, but *should not be used to deselect* TTPs from being considered during a hunt.

These methods of initial filtering will also be useful during the hunt execution for tuning and refining analytical results.

2.4 Execution Phase (Right Side of the ‘V’)

2.4.1 Identify and Mitigate Collection Gaps

2.4.1.1 Confirm Existing Data Sources – Presence and Validity

When first embarking on a hunt, and periodically throughout the hunt, analysts should assess how well existing data collection meets the requirements. For example, analysts might need to determine if the data are present, valid (free from configuration errors and adversary tampering),

and collected across the terrain of interest continuously. One method to check that the data is present is simple frequency analysis of relevant event codes over time to detect periods of time when collection of that event may have been disrupted. One way to perform a validity check is to compare results from different data sources to ensure consistency (e.g., host-based network connections corresponding with flow data from a network sensor). Frequency analysis of event counts by IP address or hostname can be used to identify coverage gaps across the terrain.

2.4.1.2 Deploy Required New Sensors to Fill Gaps with Existing Collected Data

One common problem with hunting missions is the lack of available data to observe the adversary activity from a part of the terrain that lacks sensor coverage. This could be an area of the network that doesn't have a network sensor positioned on it; a host that does not have adequate logging configured; or some other visibility gap that precludes being able to hunt effectively due to a lack of data. Hunt teams should assess what coverage is available within the environment and supplement that coverage with necessary configuration changes, centralized data collection, and deployment of additional sensors and capabilities to mitigate those visibility gaps. If, at any time before or during a hunt, significant gaps are detected between the desired and actual data collection, the team should assess how to handle each gap. When possible, new sensors should be deployed to fill the gap. A portfolio analysis approach is recommended to determine the most efficient set of sensors to deploy to maximize coverage of required data. The team should bear in mind that some sensors are less costly or easier to deploy than others. For example, *Event Tracing for Windows (ETW)* or other host audit logging capabilities are often already present and can be activated with configuration changes, whereas EDR tools may require acquisition, deployment, and calibration to start collecting the right data. The hunt team should also consider operations security (OPSEC) in the deployment of new sensors, and balance the value of the additional data collected with the visibility of that sensor to the adversary. The mission owner might have concerns about how deployment of new sensors will impact their business or mission functions. The hunt team should be prepared to communicate effectively to the mission owner about the pros and cons of each aspect of their hunt plan relative to OPSEC, mission impact, and probability of successful hunting.

Note that sensor deployment and data collection that starts post-compromise may be less effective in comparison to continuous, ongoing monitoring due to the issues in covering the time domain mentioned above. Additionally, sensors deployed on already compromised hosts may not be able to observe activity effectively due to anti-monitoring and anti-forensics capabilities of the adversary's tools (i.e. *Defense Evasion* tactic in ATT&CK). However, it is unlikely that the adversary can subvert every host, network device, or sensor, if the sensoring coverage is comprehensive enough. In these cases, it can be more effective to search for the side-effects of an evasive technique or search within data sources that would be unaffected by that technique (e.g. searching for one-sided network connections between two hosts may indicate missing data from a compromised host's sensing).

2.4.1.3 Addressing Collection Gaps

If deploying new sensors is not possible or practical, the team should assess if other data collected can be used to fill the gap, perhaps with lower confidence or granularity of visibility. This can be done by mapping data sources to the analytics they enable. This mapping allows the

team to assess the impact on the hunt mission due to the lack of a particular data source and adjust their analytics to adapt. A mapping of analytics to threat knowledge, terrain and mission should then be used to assess overall hunt efficacy impact.

Knowing which adversarial techniques are not visible due to a data gap – where the blind-spots are with respect to terrain or time – and thus, which hunts have reduced analytic coverage can help the team determine how to proceed and to communicate to the mission and network owner(s) about the impact. If certain adversarial techniques are no longer visible due to the gap, the hunt team may need to adjust their overall analytic approach as they seek initial detections and connections between detected adversarial behavior. This could include modifying which behaviors are included in initial detection analytics, or increasing tolerance for missing evidence in linking two suspicious events.

If certain areas of cyber terrain are not covered by existing data collection, increased scrutiny can be placed on links between covered terrain and those blind spots (e.g., network connections made from covered systems to those without coverage). If certain windows of time lack data collection, links between events on either side of that window will be more tenuous. At a minimum, the hunt team should at least be aware of, and communicate to mission and networks owners about, the gaps in visibility and the impact that has on the hunting results.

2.4.2 Implement and Test Analytics

The abstract analytic, the data model, and available data sources can now inform the creation of an analytic within the team's analysis system. The form of the analytic may vary depending on the specific system used. For example, if the team is using Splunk, the analytic will be in the form of one or more Splunk queries.

The analytic should be written to specifically identify the behavior noted in the TTP and leverage the data model as much as possible. The risk of writing analytics without modeling the data first is that the analytic will be more implementation and environment-specific. For example, if the analyst models data on process creation from Linux hosts and Windows hosts using a 'process creation' alias, the implemented analytic can refer to 'process creation' without having to specify specific Windows event IDs or Linux events within the analytic itself. The analytic becomes more useful across terrain types and data types – ideally, one analytic to run and query regardless of terrain. This ideal may not be practical for all analytics but serves as a goal for implementation guidance.

2.4.3 Hunt: Detect Malicious Activity and Investigate

Hunting is an iterative process that requires creativity and flexibility. It is enabled by a core sequence of steps that provide a foundation for that flexibility. The flow chart in Figure 5 below describes that core sequence. It begins with collected data and knowledge of malicious TTPs and illustrates fundamental processes to leverage that knowledge to filter the data efficiently and find

the malicious activity. Once that activity is sufficiently understood, costs can be imposed on the adversary. Each step in this process is described in greater detail in the following sections.

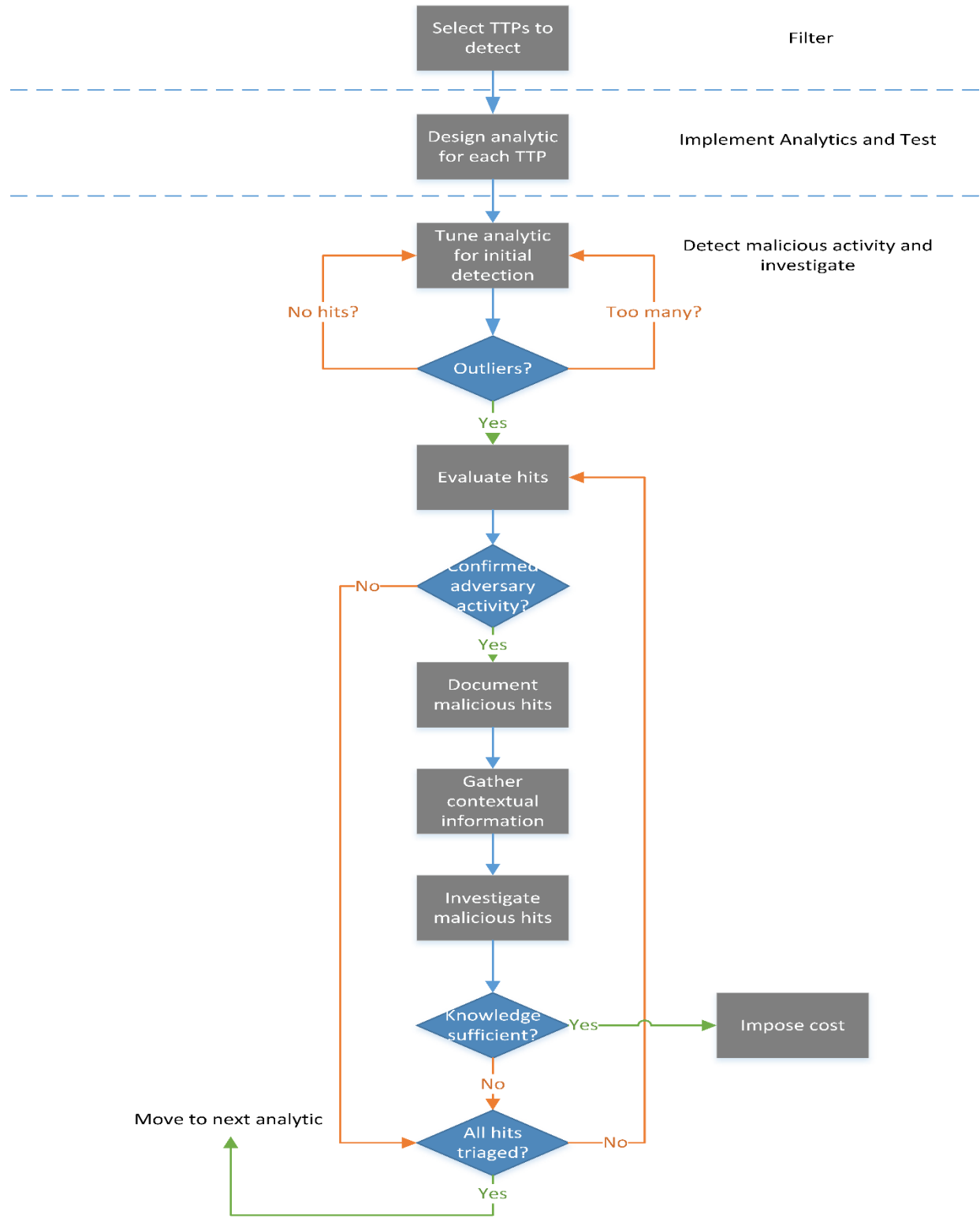


Figure 5 General Hunt Process Flow

2.4.3.1 Tune analytic(s) for initial detection

The first challenge for an analyst is to tune the analytic efficiently to the subset of hits with malicious activity. Narrowing the space across which results are queried will (normally) reduce the total number of events to be analyzed. Broadening it may result in greater total events, but it may also reveal patterns that would otherwise elude notice. It may be useful to count the number of occurrences of events for a unit of terrain and time (e.g., on each machine, over the course of one day). This results in three-dimensional data which can be represented as a heat map (see Figure 6) where x and y axes are time and terrain, respectively, and the color of each square corresponds to the count of that list of behaviors for that terrain in the timeframe specified.

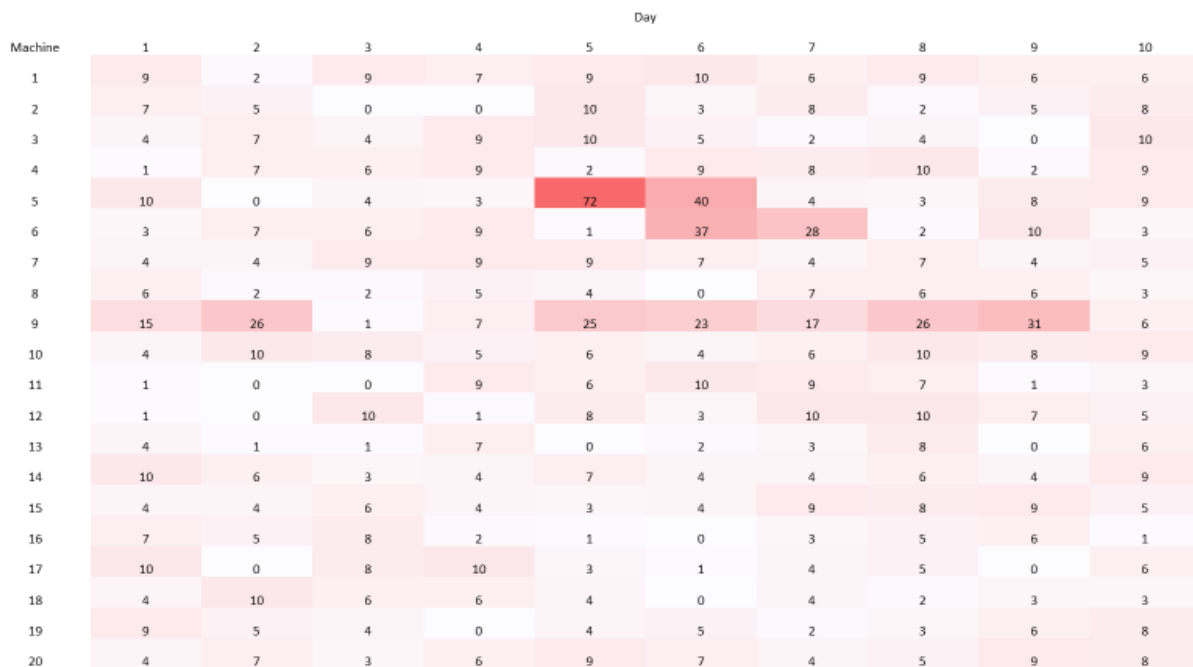


Figure 6 Heat Map

The heat map represents the number of instances of a set of behaviors (those associated with adversary behavior, but not prevalent as benign in this network) occurred on each machine for each day. This heatmap enables the analyst to quickly focus on Machine 5, day 5 as a lead to pursue.

Switching the axes around also yields interesting results. For example, switching the Terrain and Behavior axes produces a map detailing the prevalence of certain behaviors occurring throughout the network. This could be used by an analyst to identify instances where specific behaviors are increasing (or decreasing) and may even reveal the overall flow of an attack. For example, during Day 1 of an attack there was a surge of behaviors related to the *Discovery* tactic. This was followed by a surge in events related to *Lateral Movement* in Day 2 and finally a surge in *Exfiltration* events in Day 3.

Each of these behavioral analytics will have a false positive rate. We recommend beginning with analytics with a relatively low false positive rate individually, however this is difficult to know in advance of an event. Terrain specific knowledge could be used here to inform the choice of

analytics. For example, the hunt team would exclude the Sysinternals tool *psexec* from the list of commands they are searching for in a network with frequent but benign use of the *psexec* command.

Constraining the analysis space in terms of behaviors can be challenging, because what behaviors constitute normal can vary wildly from environment to environment and what may stand out as abnormal in one environment could be an everyday occurrence in another. For example, one could anticipate that users rarely if ever use Remote Desktop Protocol (RDP) as part of their daily work, however upon analyzing the logs the team may find that there are in fact several users who do so every day. To compensate for this, the hunt team may need to adjust certain aspects of the behavior they are looking for. This could be whitelisting certain aspects of the behavior that are frequently occurring (and thus constitute a large portion of the noise), or reducing the number of behaviors being searched for by the analytic (e.g. removing an executable that is observed to be running across the entirety of the environment).

The analysis space can also be tuned based on known good behavior. This decision could be informed by open source research into standard behaviors things like applications and protocols or by asking the network owner or administrator if there is known-good activity on the network that is likely to be observed using a given analytic. After several rounds of recalibrating the analytic, it can be beneficial to ask the network owner or system administrator about the observed noise and if they can deconflict it.

To further refine the search for malicious activity, the hunt team can modify the unit of aggregation in time or terrain until this heat map shows significant outliers relative to background noise. For example, one could begin with a time unit of one day and a terrain unit of one machine. Depending on the situation, a time unit of one hour or one week might be more effective at separating signal from noise. Similarly, using a unit of a subnet or username might be more effective than a machine for the terrain dimension.

Constraining the analysis in the time dimension to shorter durations might help detect adversarial activity that is pervasive across the terrain but is concentrated in time (e.g., a massive initial infection, reconnaissance phase or exfiltration). Conversely, limiting the analysis to smaller units of terrain could help highlight adversarial activity that is “low-and-slow”, but focused on certain places in cyberspace.

Constraining the analysis in the terrain dimension can also be rewarding for the analyst. Analyzing everything across the operational environment is computationally infeasible, so the hunt team could identify the cyber assets that are most critical to execution of an organization’s mission. Another means of constraining the terrain dimension is to assign different members of the team to focus on different segments of the terrain.

Occasionally an analyst will find themselves in a position where an analytic returns nothing. This does not necessarily mean the logic behind the analytic is flawed, but may be the result of over-tuning of the analytic. Depending on the situation, any of the three dimensions could be relaxed to reveal activity. For example, the time window could have been too narrow, the adversary might not have reached that terrain yet, or the adversary is not utilizing the technique sought in

this instance. Whatever the case, incrementally expanding the scope of what the analysts are looking for can help reveal additional information without overloading the analyst.

2.4.3.2 Evaluate Hits

Once the number of events generated by a given analytic is reduced to a number small enough to devote some hunt team resources to pursuing each, the hunt team needs to resolve each of the hits returned. Events belonging to an outlier group are not necessarily malicious, so each one needs to be evaluated in depth. The methods used for evaluating results do not necessarily follow a prescribed order; the analyst decides which methods to pursue based on available information, experience, and expertise.

Once suspicious activity has been identified, it may be necessary to once again widen the aperture (in time or terrain) on the set of data the analyst examines to see if this is truly suspicious. For example, suspicious activity on one machine might actually be benign if the same activity occurs on all the machines in that network, and has for a considerable amount of time. At this juncture, it might be prudent to inquire of a network administrator and/or user as to the nature of the activity.

Some events will require deeper inspection to make a determination regarding maliciousness. What form this takes depends heavily on the event in question, but two examples are parsing out the full command line from a process creation and extracting data communicated over a network connection. As a hunt team's processes mature, these cases will ideally decrease as the team becomes more familiar with the kind of data required by the analysts.

Contextual information is often needed to determine if an event is malicious or not. Adversaries do not perform actions in isolation and thus the traces of activity they leave behind do not exist in isolation either. There will be a chain of causality to follow that can be used to connect seemingly disparate events. Therefore, if the analyst can draw a direct connection between the event under investigation and another event or piece of intelligence that is known to be malicious, the certainty that this event is also malicious increases significantly. For example, a command prompt was observed running an executable that, while unusual, is not in itself malicious. However, upon examination, the parent process of that command prompt is discovered to have been spawned by a previously identified malicious executable. Additionally, the user account responsible was also previously identified as executing malicious programs. For these reasons, an analyst could reasonably deduce that the event currently in question should be considered malicious as well.

2.4.3.3 Possible Causes of False (Benign) Hits

Not all activity identified by an analytic is necessarily attributable to an external adversary. Three possibilities to consider are that the activity is legitimate, if uncommon or unusual; the activity is caused by the hunt team itself; or the activity may be an insider threat.

Analytic findings could be the result of legitimate, explainable activity. For example, system administrator activity is one of the most common sources of false positive events. Administrators frequently perform activity that resembles *Lateral Movement* techniques or *Account*

Manipulation, Scripting, Data Compression, and many other techniques found in the ATT&CK model. Administrators are also often responsible for deploying new software to the environment, which can cause unexpected events from both host-based and network data. Ideally, these kinds of planned changes to the environment will be coordinated with the hunt team so that they are prepared when the deployments happen but that is not always feasible, so the analysts need to prepare for this eventuality. These kinds of issues may require the analyst to inquire the after the activities of administrators and/or individual users to deconflict results.

Software developers can also introduce unexpected behavior to the environment. For example, a web development team may be standing up and tearing down web servers multiple times a day and running performance tests against them, causing huge spikes in traffic to specific addresses. Alternatively, a team doing research into new adversary detection methods could cause instability with endpoint sensors and as part of their testing may recreate artifacts of some of the same attacks that they are trying to detect.

System or service misconfigurations can cause false positives in hunting analytics, and often go unnoticed until the hunting activity uncovers them. For example, tools can have inconsistent configurations, servers may have the incorrect auditing policies being enforced, and Domain Name System (DNS) servers can be misconfigured. The analyst needs to be prepared to identify instances where such is the case and to notify the party responsible so it can be addressed. This kind of issue could have the unintended side effect of changing the baseline of the environment, so any anomaly-based analytics in use may need to account for that fact.

Analysts must be cognizant of the possibility of detecting their own hunting activities or sensors rather than the adversary, because various methods used by adversaries can be used by analysts to collect and aggregate data. For example, some teams may use *PowerShell* scripts running as administrators to collect data from endpoints or they might run a vulnerability scan to scan for misconfigurations or vulnerabilities on the network. For this reason, teams should be aware of their footprint in the environment so if they are the cause of what looks like an event it can be quickly dismissed. This use case also exemplifies the need for communication within the team so team members are aware of what each other are doing and can quickly deconflict results. Additionally, this could occur when there are multiple defensive teams operating within the same environment. An example of such a situation would be where an external team comes in to augment existing manpower. If the two teams are not properly coordinating activities with each other, they run the risk of both duplicating effort and tracking each other rather than the adversary.

In rare cases, the activity may be related to an insider threat. In these cases, the hunt team will need to involve any local counter-intelligence or insider threat group that may exist within the organization. Sometimes however, behavior that may initially look like an insider threat may be a relatively benign policy violation. This kind of activity could come from a variety of motives, ranging from ignorance of accepted policies, to an over-enthusiastic can-do attitude, or even pressure from leadership to willfully break from policy. The response to each of these possibilities is outside the scope of this document and needs to be addressed on a case-by-case basis but it is important to note such deconflictions to ensure such activity is not misattributed.

2.4.3.4 Document Malicious Hits

If the detected event is determined to be malicious then it should be captured in such a way that the information can be shared between team members as well as other parties with an interest in the investigation (e.g. management, other defensive teams, etc.). There are numerous ways that this information can be captured, here are some examples:

- **Adversary Timeline** - The Adversary Timeline is a simply a list of observed activity in chronological order. The list should contain more than just the event that was observed, but also contextual information like the user (if any) and host/IP address responsible. By adding this additional information, analysts can gain a greater appreciation of how the events are related. Once enough events have been identified the team should consider trying to group the raw events into segments of activity. Doing so will help the team achieve a “big picture” view of the activity which may aid in understanding the overall “campaign”.
- **Host List** - A list that contains relevant information regarding the various hosts that have been identified as being related to confirmed malicious activity. Some of the information that a team would want to capture is:
 - Hostnames
 - Users
 - Owners
 - IP Addresses
 - Why this host is on the list
- **User List** - A list that contains information on users that have been confirmed as performing malicious activity. Additionally, consider adding users whose credentials may have been compromised, even if those credentials have not been tied to malicious activity. This may also include relevant information about the user that the hunt team may find useful like:
 - Contact Information
 - Supervisor
 - Location
 - Role
 - Assigned Machines
- **Malware List** - A condensed list of the malware that has so far been found within the environment. Any utilities or built in programs that are being used by the adversary can also be tracked here. Some of the information that should be captured here is:
 - Malware/Program Name
 - Any Aliases
 - General Description
 - Other Pertinent Details about the Malware
- **Activity Graph** - A map that describes the chain of activity between the various hosts identified. The purpose is to provide a visual representation of the malicious activity occurring on the network. The important details to capture on the graph are:
 - Hosts that have been confirmed as having malicious activity take place on them. For this purpose, the hostname, rather than the IP address, is more useful as a given

computer could have multiple IP addresses assigned to it for many reasons. However, there will likely be instances where an IP address is all that is available to use (such as an external C2 server).

- Network connections made between each of the hosts to show where the adversary pivoted in their operation. It is important to note that capturing every network connection made between each host is unrealistic so only a select few should be rendered. Initial malicious connections between two hosts are important to note as it helps establish how the adversary is moving around the network. As part of the connection information, it is important to capture the time/date of the connection as well as the protocol or method used.
- User credentials used (if any) are important to note as well. If legitimate user credentials are being used then noting that can help inform directions that the hunt team needs to investigate further in. For example, if a user is observed making a malicious RDP connection to a host but no information regarding what that user did on that host has been found yet, that is something that should be investigated. Conversely, if a user's credentials are being used maliciously to navigate the network then the hunt team needs to trace back those connections to try and find the moment where those were compromised.

2.4.3.5 Gather Contextual Information

Contextual information can be extremely important, as outlined above, and for that reason collecting it is of utmost importance as a team continues its hunt. Not only does it add context to events that have been identified as malicious, but can also be used to drive direction for further investigation. Oftentimes, the most important information to capture is that which can help to establish a chain of causality. In layman's terms, this just means information about what caused the event in question and information that reveals anything the event in question may have caused. By capturing these pieces of information, the team can focus their efforts on events that are directly tied to a known bad event. Events that precede a known malicious event should be considered very suspicious and events that were caused by it should be considered malicious. The following paragraphs highlight some of the things that an analyst should capture in relation to a given event. This list is by no means exhaustive, but should provide a solid starting point for developing the team's own preferred methods of connecting known malicious events together to create an understanding of what happened.

Related Processes

Identifying related processes can be an invaluable tool as they have very well-defined relationships with other processes. Through these relationships it is relatively easy to establish chains of activity. The most important pieces of information to capture in this regard are "child" and "parent" process name/image paths, process ID's, and command lines. Additionally, the full command line of processes should be captured if possible, as it often contains invaluable information about the event. The arguments contained within show how exactly that executable is being used and may also reveal additional information like any files that may have been used/modified or network connections that should be investigated further.

Network Information

Any network related information that can be tied to a given event is also very important to capture as it will potentially reveal how that event fits within the greater campaign occurring across the network. Without the context of the related network events, an analyst is left with isolated series of activity with no direct ties to events happening on other hosts. The primary pieces of information that an analyst needs to capture relating to network activity are any IP addresses, ports, and any details regarding the content of the communication itself. The last item in that list is difficult to define as it may vary considerably based on protocol and available information. For example, if the analyst observes a Secure Copy Process (SCP) create to a remote address, then the analyst will have information regarding the file being transmitted. If, however, the analyst's visibility is limited to just netflow events, then the nature of the file being transmitted may be impossible to discern. Resolving any IP addresses identified to hostnames will also be beneficial for further investigations as well as coordinating with other team members.

System Files

Even in "file-less" attacks, adversaries will almost certainly interact with files on a system at some level. For example, they may exfiltrate a user's documents or run an executable that, while normal for windows, is being run from an unusual directory. As an investigation progresses it is important to keep track of pieces of information that are tied to relevant files. Ideally, these would be captured in a standardized data model, however some items that can be tracked are the file name, the file path of the executable, a hash of the file (especially if it is a binary or executable file), and any timestamp information. Some types of files to keep an eye out for include email attachments preceding other observed activity, creations/deletions/modifications of files around the time of other events, and any files that are directly observed as being part of an event itself (e.g. any found within the command line of a malicious process start, or observed being transmitted over a network connection).

User Information

Leveraging user information can provide additional context regarding the adversary's activity. Not only can it reveal related information from the same data source, but it can be used to pivot across many of the host-based objects found in the data model. It can be used to identify additional processes being run by the same user, to look for files that that user was responsible for editing, as well as establish boundaries of activity by looking at log in and log out times and seeing how those log ins were accomplished. Other compromised hosts can also be identified by looking for all those things occurring on other hosts. While that activity cannot necessarily be considered malicious, if the activity appears to be the same on both then there is a good chance that further investigation of that other host is warranted.

2.4.3.6 Investigate Malicious Hits

To pursue a malicious hit, the hunt team should "pull the thread" both backwards and forwards to find the activity which caused the hit (ideally back to the initial infection), as well as subsequent activity to determine the scope and scale of the adversary's actions.

In most cases, to begin pursuing the adversary, we recommend working backwards to find the causes of the detected event. This will help determine the full scope of the activity, attribute the events to a specific adversary group, and gain the most useful knowledge for planning decisive response action. Ideally, the hunt team will have the required data collection and analytic capability to determine each link in the causal chain of events leading to this initially-detected event.

For example, on a Windows operating system, the responsible process could be found through identifying the parent process, *schtasks* command that scheduled this process start, the user event that triggered process start, or other methods as enumerated in ATT&CK's *Execution Tactic*. To trace the chain of causal execution across network traffic, the analyst might look for *Lateral Movement* methods like *Remote File Copy*, *Exploitation of Remote Services*, or other methods.

If no causal events are found, the analyst will need to relax the requirement for finding evidence of each link in the causal chain. The analyst should consider the range of processes, systems, etc. that could have resulted in the event under consideration. For example, recent network connections, other activity by the same user or machine in the recent past, or other machines exhibiting identical behavior (e.g., same command line or network traffic).

In parallel with, or after sufficient information has been obtained regarding causally preceding events, the hunt team should investigate caused or related subsequent activity. Similar to the investigation of preceding events, analysts should look first for evidence of directly-caused activity such as child processes, file creations, or opened network connections. When needed, the analyst should expand the investigation to include other machines exhibiting identical behavior and other suspicious files, processes, or activity on the same system. As the investigation proceeds, analysts can consider the direct descendants of known-malicious activity to be malicious, while considering "sibling" processes and parent processes as only suspicious pending further investigation and context.

Throughout these pursuit investigations, analysts should continually refine the characterization of findings. As they gather more information, they should update a common knowledge repository (e.g., textual reporting, graph of activity) about the currently known chain of events, to include information regarding whether they are indicative of a specific set of adversaries, whether this activity is indicative of a certain stage in the Cyber Attack Lifecycle, and adversary intention. As new information is added to a shared repository, the team should also regularly determine what gaps in knowledge and/or visibility should be filled next and who and/or what could help fill them.

2.4.3.7 Identify Similar Behavior Across the Network

Looking for similar behavior across the network may reveal other instances of compromise that were initially missed. What exactly an analyst might look for is dependent on the event that is currently being looked in to, but some examples include:

- After successfully identifying an executable being used maliciously with specific arguments, those arguments are used to identify other instances of that executable being used even if it is under a different name.

- A connection is made over a specific port, which is then followed by the writing of a file that has been discovered to be the 2nd stage payload for the adversary's malware.
- A malicious instance of encrypted file compression is observed on one host is observed. While this may not be unusual for the environment overall, there were several other instances of the same user using the exact same syntax across multiple machines.

2.4.3.8 Impose Cost

Throughout, the team must be mindful of the courses of actions possible for responding to the intrusion under consideration by the mission owners, and tailor the investigation accordingly. There may be different choices made depending on whether the intent is to determine the full scale and scope of the intrusion versus quickly attributing the activity to an adversary group. As a result, the team may alternately prioritize finding the source of the activity, finding the subsequently-targeted systems, or performing deep forensic analysis to better understand the characteristics of the activity or artifacts likely to aid in attribution.

Over time, the knowledge gained by the hunt will be sufficient to make decisions on courses of action (e.g., quarantine, movement of the adversary to a deception environment, placement of honey credentials or misinformation, perimeter blocking, offensive actions, or kinetic/legal/diplomatic actions). This may occur when the full extent of the adversarial activity is known, or when the defensive team's knowledge and ability to destroy, deny, degrade, or disrupt the adversary exceeds adversary's ability to destroy, deny, degrade, disrupt operations. The hunt team must strike the right balance between waiting too long to act, and acting prematurely. Too much emphasis on learning the full extent of the activity may hamper timely responsive action. Acting before sufficient knowledge is gained could result in tipping one's hand to the adversary without having significant impact on their presence in the network, or their ability to accomplish their objectives. This is a strategic decision which should incorporate an understanding of the adversary's activity, but also their intent and capabilities as well as the potential or actual impact to the defended environments. This is a ripe area for future research.

2.5 Report

There are several reporting requirements that should be addressed as part of planning for, conducting, and concluding a hunting operation. When planning a hunt, communication and reporting channels will need to be created for all stakeholders. These will include the hunt team's senior leadership and the mission owner, especially for hunting on an environment that is not owned by the hunt team's organization. Key items to report are the general timeline for the hunt, what phase of the timeline the team is in, any confirmed presence of an adversary, systems affected (both systems that are being investigated as well as known compromised systems) and what damage or risk is currently posed by the adversary. In many cases, this will be an assumption based on available data. Avoid excessive speculation on the adversary's intent and capabilities, but instead focus on what facts have been uncovered from ongoing analysis. Be sure to establish regular update cycles, so stakeholders know when to expect new information.

Reporting also entails knowing the purpose(s) of the hunt – if the hunt is for remediation, reporting should be tailored towards informing stakeholders and remediation personnel where to pre-stage remediation capabilities and what the full scope and scale of the intrusion is to enable

decisive action. Communication channels should be as “out of band” as possible and not be conducted on the environment being hunted, to avoid alerting the adversary and allowing them to react to hunting efforts.

3 Implications and Future Work

There are significant gaps between the current situation and what is needed to successfully hunt using this new methodology. Changes are required in operations, intelligence, training, and capabilities. USCYBERCOM can take actions today to begin making those changes. There are also many open questions that will require future research to determine best practices. The Joint Cyber Warfare Center (JCWC) is well-positioned to begin addressing those research questions in future experiments.

3.1 Implications for Operations

In the short term, operations will need to shift to incorporate new and ongoing host-based collection and analysis methods. We recommend the operational community make the following changes.

- Commanders should direct employment of this TTP-based approach, requiring planning artifacts demonstrating employment of this approach.
- Commanders should establish readiness conditions for tactical forces based on their ability to effectively deploy capabilities and detect adversary TTPs, under Commander-specified conditions. Tactical forces should be able to deploy host-based sensors and establish data flows to their analytic platforms for analysis.
- USCYBERCOM (or Joint Forces Headquarters (JFHQ)-DODIN) should levy a requirement that host-based sensing be deployed across DoDIN. Hunting is most likely to be effective when sensing captures all adversary activities, beginning with the breach. In the interim time before continuous monitoring is established (and possibly after an adversary breach), mission owners need to empower operators to adjust the sensor posture and deploy sensing capabilities that enable hunt operations to detect ongoing adversary behavior.
- Cyber National Mission Forces (CNMF) should advocate for host-based sensing across non-DoD partners’ terrain.
- Service Cyber Components should develop an operational baseline of friendly TTPs against adversary TTPs, assessed for effectiveness. That is, for each adversary TTP, a description of what friendly capability and TTPs will be used and under what conditions they are effective. This baseline should be reported to JCWC, who will establish a Joint standard and advocate for disseminating effective approaches.
- In garrison, Cyber Protection Teams (CPTs) should focus their efforts on developing detection hypotheses, consuming and analyzing candidate data sources, and creating analytics for detecting adversary TTPs to prepare for hunt operations.

3.2 Implications for Intelligence

Current intelligence production does not typically focus on adversary behavior naming specific TTPs, or at least not in a structured way utilizing a generic adversary model. Creation of this generic adversary model containing all known TTPs should be an ongoing function of intelligence, incorporating public, open, and closed intelligence sources to identify and update the model with new TTPs. The generic adversary model needs to be kept current and actionable for use by the tactical force. We recommend the intelligence community make the following changes.

- USCYBERCOM J2, in partnership with ODNI, needs to assign responsibility to develop and keep current a model of adversary model, which includes a scheme of maneuver that is a superset of all known adversary TTPs across all phases of operations and all types of terrain. A view of this model must be available and actionable at the unclassified level.
- Intelligence gathered about adversary operations should be analyzed to identify information relevant to the three axes of the hunting analysis space:
 - For behavior, identify information about specific phases of their operations and what specific techniques were utilized during these phases. Identify new techniques to add to the generic adversary model. Identify the operational tempo of how quickly the adversary moves from tactic to tactic. Identify any patterns to the order in which they execute tactics. Identify how the adversary searches for and locates defensive capabilities of their targets. Identify whether, and how, their activities and operational tempo change if they identify hunt team activities or incident response measures.
 - For terrain, identify what type of terrain was targeted within each phase and try to identify patterns of movement within the terrain. How many hosts or accounts does the adversary target? Do they favor particular terrain targets during certain phases of their operation?
 - For time, identify how long each phase was, how long did the adversary take to transition to each phase, and the timing or apparent triggers of their activities.

3.3 Implications for Workforce Development

The Cyber Mission Force needs to understand OCO scheme of maneuver. Accordingly, we recommend the following Services make the following changes to better develop the work force.

- Services should provide operators, analysts, and planners education on the generic adversary model. They should be fluent in the operational art of OCO.
- Operators and intel analysts should receive sufficient education to recognize when they have observed new TTPs and can report on their observations. The Services should update Job Qualification Requirements (JQRs) to incorporate these requirements.
- Training for the operational force will need to be adapted to include data analysis, data modeling, and TTP-based analytics creation.

- Planners should be educated sufficiently to understand what capabilities are needed to get sensor coverage of adversary TTPs across the terrain. The Services should provide high-level training on adversary scheme of maneuver for all planners.
- Individual and team proficiency should be measured against the adversary model in terms of what adversary TTPs the team can detect; to what level of detection, characterization, and attribution; and under what conditions. The Services should update JQRs to incorporate these requirements.

3.4 Implications for Capabilities

Data from continuous monitoring is generally preferred over forensic data collection (see time domain discussion within section 1.2 *Analysis Space*). Forensic tools can complement other data sources but are much less effective as primary data sources. This point is critical to the successful use of this hunting approach, and suggests that DoD cannot primarily rely upon deployment of a fly-away breach response team and toolkit for mission success. Instead, DoD must focus substantial resources primarily on proactive preparation of DoDIN and friendly environments with sensor and log collection well ahead of any possible breach. Accordingly, we recommend the following changes for capabilities.

- USCYBERCOM should direct that program office develop and maintain host-based continuous monitoring capabilities. Recognizing that the generic adversary model reflects a continuously evolving set of TTPs, these capabilities should be continuously updated to be able to detect all techniques described in the current generic adversary model.
- Service Cyber Components should establish and maintain an operational baseline for Defensive Cyberspace Operations (DCO) capabilities. Initially, they should organize operational test and evaluation activities to determine CPT capability and TTP pairing effectiveness against adversary TTP in various conditions. Service Cyber Components should provide the output of those activities to JCWC and the Office of the Secretary of Defense (OSD) Operational Test & Evaluation (OT&E).

3.5 Future research

While this approach offers significant improvements to operational effectiveness over the status quo, it can still be improved. Areas for research include:

- This approach focuses on better employing intelligence in the form of the generic adversary model. It is not clear how much advantage, if any, improved knowledge of the terrain and supporting mission will inform hunt operations. The state of practice in terrain mapping and mission mapping (wherein missions are functionally mapped onto the terrain) is still fairly immature. As these disciplines mature, terrain and mission maps should be evaluated for their impact on hunt mission effectiveness and efficiency.
- The approach, as described, does not examine how the team or personnel organizational structure impacts hunt effectiveness or efficiency. Further research is needed to identify what organizational model and member skills combine to provide optimal effectiveness.

- At present, this approach is being developed with CPTs in mind. It is expected that CPTs will transition this concept to the organic cyber defense forces. How can organic forces and CPTs collaborate when using this methodology?
- This approach focuses on the challenges to detect and characterize adversary activity in friendly space. Additional work must be done to support attribution, recognizing that most characteristics of capabilities and TTPs can be manipulated to lead to misattribution.
- Follow-on research efforts identifying how to incorporate generic adversary models into intelligence planning and products such that they enable hunt would be beneficial.
- Existing data models in many SIEM and analysis platforms focus primarily on data normalization rather than shaping analytical queries towards identifying behaviors. Further research should be undertaken to examine what elements of a data model are more effective at assisting analysts in identifying adversary activity captured within the sensor data. Initial modeling efforts should focus on commonly available data sources found within DoD environments (e.g., Windows Event Logs, Linux Audit Logs, existing EDR solutions within the CPT kits,).
- Further work identifying and recording the data collection, modeling, and analysis requirements to enable this methodology would be of benefit to drive capability development. Open questions include:
 - How well can host-based event collection, for hunting purposes, scale up to very large environments (e.g., 1 million hosts).
 - Is large-scale centralized collection for hunting feasible or even necessary, as opposed to a decentralized collection model?
- As the current center of gravity for threat hunting research exists outside DoD, this approach was written in language utilizing terms commonly used in industry and academic communities. This was done to ensure the concepts and methods are described accurately. The terminology should be transitioned to military terminology as soon as possible, noting when metaphors from other domains are inadequate and misleading. We note that many, but not all, current doctrinal concepts have an appropriate mapping into cyberspace.

4 References/Bibliography

- Berlin, K., Saxe, J., & Slater, D. (2015). Malicious behavior detection using Windows audit logs. Retrieved from <https://arxiv.org/pdf/1506.04200.pdf>
- Cole, E. (2016). Threat Hunting: Open Season on the Adversary: A SANS Survey. Retrieved from <https://www.sans.org/reading-room/whitepapers/analyst/threat-hunting-open-season-adversary-36882>
- Director of National Intelligence. (2017). Cyber Threat Framework. Retrieved from <https://www.dni.gov/index.php/cyber-threat-framework>
- FireEye (2014). The Pyramid of Pain: Intel-Driven Detection & Response to Increase Your Adversary's Cost of Operations. Retrieved from: http://rvasec.com/slides/2014/Bianco_Pyramid%20of%20Pain.pdf
- Gartner (2017). How to Hunt for Security Threats
- Lee, M. & Rascagneres, P. (2018). Who wasn't responsible for Olympic Destroyer? (blog). Retrieved from <http://blog.talosintelligence.com/2018/02/who-wasnt-responsible-for-olympic.html>
- Lee, R., & Lee, R.M. (2017). The Hunter Strikes Back: The SANS 2017 Threat Hunting Survey. Retrieved from <https://www.sans.org/reading-room/whitepapers/analyst/hunter-strikes-back-2017-threat-hunting-survey-37760>
- MITRE (2015a). Adversarial Tactics, Techniques & Common Knowledge (ATT&CK). Retrieved from <https://attack.mitre.org>
- MITRE (2015b). Cyber Analytics Repository (CAR). Retrieved from <https://car.mitre.org>
- MITRE (2018). Crown Jewels Analysis. Retrieved from: <https://www.mitre.org/publications/systems-engineering-guide/enterprise-engineering/systems-engineering-for-mission-assurance/crown-jewels-analysis>
- Scarfone, K. (2016). The Hunter's Handbook – Endgame's guide to adversary hunting. Retrieved from <https://www.endgame.com/resource/white-paper/hunters-handbook-endgames-guide-adversary-hunting>
- Soria-Machado, M., Abolins, D., Boldea, C., & Socha, K. (2017). CERT-EU Security Whitepaper 17-002: Detecting Lateral Movements in Windows Infrastructure. Retrieved from: http://cert.europa.eu/static/WhitePapers/CERT-EU_SWP_17-002_Lateral_Movements.pdf
- Sqrrl (2016). A Framework for Cyber Threat Hunting. Retrieved from: <http://sqrrl.com/media/Framework-for-Threat-Hunting-Whitepaper.pdf>
- Sqrrl (2018). Huntpedia: Your Threat Hunting Knowledge Compendium. Retrieved from: <http://info.sqrrl.com/huntpedia>
- Strom, Blake, et al. Finding Cyber Threats with ATT&CK-based Analytics. <https://www.mitre.org/sites/default/files/publications/16-3713-finding-cyber-threats%20with%20att%26ck-based-analytics.pdf>