SANS

A SANS Spotlight

Thinking like a Hunter: Implementing a Threat Hunting Program

Written by **Matt Bromiley**

April 2019

Sponsored by:

IBM

Introduction

Protecting an enterprise environment can sometimes feel like an uphill battle. Information security teams are often stuck in cyclical patterns where it feels as if the alerts never end and the attackers are constantly successful.

Unfortunately, this pattern is a symptom of organizations that live in reactive mode. In this mode, security and/or response teams are waiting for an alert—internal or external—to tell them where to go next. There is little, if any, direction to *find* threats before they become something worse.

To truly get ahead of attackers, organizations should start thinking proactively; in other words, think like threat hunters. Admittedly, the term "threat hunting" is not a new one. In fact, many mature organizations have various threat hunting programs that are either separate teams or, more often, integrated with the security operations center (SOC) and/or incident response teams.

When many organizations hear the term threat hunting, however, it often gets translated incorrectly to "go find evil." Finding evil is much easier said than done—it's not as if the attackers are waving white flags telling you all the steps they took! Instead, threat hunting is a complex undertaking that needs to take a long-term view on success.

Threat hunting is often much easier said than done; it requires teams to be thinking in a proactive sense, and not be bogged down with unnecessary reactions. If threat hunting is successful, however, be prepared for a quick shift into investigative mode!

The deep roots of successful threat hunting don't exist in the knowledge of attack techniques; they exist in visibility and situational awareness. The true purpose of a successful threat hunting program should be two-fold:

- The first objective is to identify previously unknown or ongoing (aka not remediated) threats within the environment.
- The second objective—the true benefit to the organization—is gaining a deeper understanding of the organization's technical landscape.

In this paper, we focus on bridging the gap between the two objectives and discussing the whats, whys and hows of threat hunting. We'll examine techniques that can be immediately applied to your environment to help you either build a new hunt team or hone your existing one. It's important not only to understand how attackers do what they do but also how their tricks can be identified and remediated within your environment.

The Importance of Threat Hunting

Before taking the plunge and asking your teams to "go hunt," it's important to identify the goals and objectives of the hunt team. As previously mentioned, there should be two main objectives to any hunt program: 1) to proactively search for threats so as to limit attacker impact, and 2) to gain a better understanding of the environment. We'll examine each in detail.

Hunting with Intention

As we mentioned, the first and most obvious goal of any threat hunting team should be to identify previously unknown attacks/threats to the environment. It is very possible that an attacker may have a foothold in an environment but is not tripping any expected alarms the security team is used to responding to. Or worse, the attacker exists in an area where there is no visibility, and thus has inadvertently evaded detection simply by being in the right place at the right time.

Figure 1. Sources of Attacker Activity and How to Apply Them

Intentional hunting often relies on knowing various attacker techniques and applying them to your environment. Figure 1 presents some sources of attacker activity and how they can be applied to your environment.

During threat hunting exercises, hunt teams may also come across previously identified attacks or breaches that were either cleaned up or may still have active remnants. These are great checkpoints and should be used to validate that the latter stages of incident response and remediation (such as system reimaging, malware removal and/or necessary blocks) were done effectively. We certainly don't want the hunt team uncovering a remediated incident that is still active!

It's not uncommon for hunt teams to come across previous attacks or incidents. These are great checkpoints to ensure you are finding malicious activity. Use them as milestones of success!

Hunting with Clarity

The second, and equally important, objective for any hunt initiative should be to gain a deeper understanding of the environment the hunt team is wading through. To identify attacker activity (our first objective), the team will first have to explore what is available to them. Threat hunting is easy if you have absolute visibility. Without it, your team may be charged with performing impossible identifications.

Consider the example shown in Figure 2 of a simple spearphish and malware dropping.

The most important reason an organization implements a threat hunting program is to uncover what it doesn't know. Hunting may not always find attackers, but it should always find areas for awareness improvement.

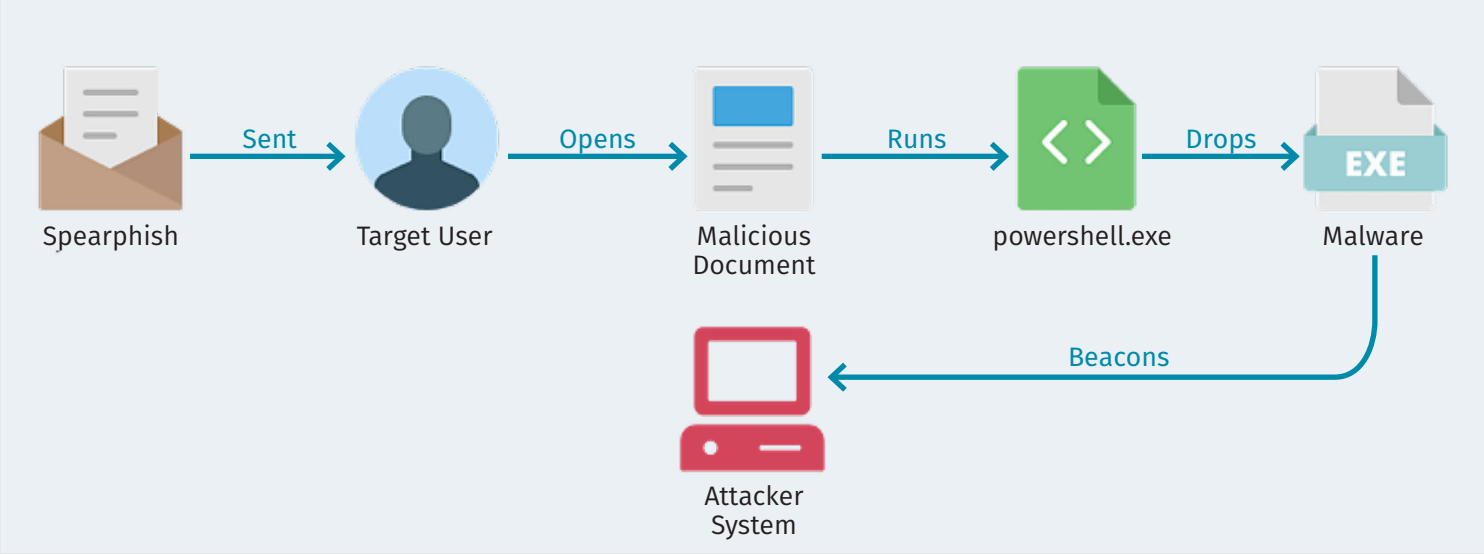

Figure 2. Sample Spearphishing and Malicious Execution Attack

In Figure 2, a user receives a malicious email which, when opened, runs PowerShell code that drops an additional piece of malware on the system. This malware runs, and subsequently reaches back to the attacker to inform it of the infection. Lots of great threat hunting leads can be extracted from this example, including:

- Users with a high volume of spearphishing, quarantined or blocked emails to identify potential campaigns
- Malicious or abnormal document names and/or locations
- Word processor programs (such as Microsoft Word) spawning command line tools such as powershell.exe
- powershell.exe downloading and/or spawning suspicious processes
- Suspicious processes performing network callouts to unknown or suspicious network locations

Maybe you can think of even more!

Once again, we are in the threat-hunting world of "easier said than done." While yes, the above points may serve as great hunting starting points, they also all rely on visibility. During the initial stages of hunting, the team will quickly realize what it does and does not have visibility into. Your hunt team will hit a wall pretty quickly if they are asked to look for something they have no way of seeing!

When the team encounters these roadblocks, two actions should be prompted:

1) The organization should consider whether the visibility gaps the hunt team has identified are root causes of its ongoing security program and work to increase them. For example, in Figure 2, we relied heavily on parent-child process combinations. There are multiple ways to gain insight into this data.
2) Initial hunts should be crafted around what the organization can see, with the intention of coming back and modifying and/or rerunning hunts if visibility increases.

Techniques for Successful Hunting

Given what we discussed in the previous section, it should come as no surprise that hunting is a very environment-specific undertaking. Some techniques can be applied to almost any environment. For example, attacker techniques that abuse a particular executable chain or service will look the same regardless of environment. It is the variables of the environment, however, that may push an attacker toward a certain technique, which can arguably make evil easier to find. Figure 3 features core threat hunting techniques.

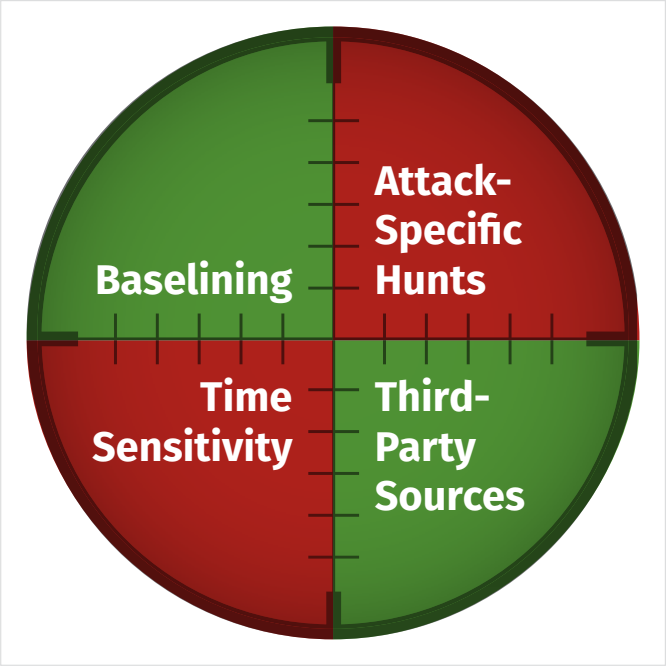

Figure 3. Core Threat Hunting Techniques

Baselining Is Your Advantage

In the previous section, we identified the need and importance of visibility, but once the team has achieved visibility, the work is only half done. The next step is to understand what "normal" looks like within your organization. Often referred to as baselining, this can be one of the more important steps a hunt team can undertake. Looking for a needle in a haystack is never a fun task, but if you start to remove hay in double-digit percentages, the needle may quickly become visible.

Baselining Questions

Admittedly, baselining can be a laborious task. To help minimize the time you spend, combine baseline analysis with attacker techniques (as previously discussed). For example, consider the following questions:

- How prevalent is PowerShell in your environment?
- If prevalent, what does normal system administrator activity look like?
- Where does PowerShell activity typically come from, and what user accounts typically run it?

You may not need to baseline *all* of PowerShell; instead, look for unexpected outliers or attacker-specific command structures.

Identifying environmental baselines can be a laborious task. Once you know how noisy or quiet something is within your environment, however, finding an anomaly becomes significantly easier.

Attack-Specific Hunts

Whereas baselining helps the hunt team gain an understanding of the environment, attack-specific hunts may help track malicious activity faster and provide some early wins. Attack-specific hunts typically involve examining a particular threat actor or threat, and modeling hunts after those particular artifacts. We briefly described an example of a spearphishing attack in the previous section, which could provide an excellent starting point for near-true positive findings.

Beware, however; attack-specific hunts will often throw off false positives. For example, how much Base64-encoded PowerShell would you expect to find in your organization? Depending on your security tools and system administrator behaviors, it may be more than you think. Therefore, baselining combined with attack-specific hunts often yields good results.

Hunts Are Time Sensitive

Another important consideration for the hunt team is that hunts are—and will always be—time sensitive. From a baselining perspective, once you have established good baseline terms, remember to validate them periodically. Ensure that any new software implementations, such as IT management or endpoint security, aren't causing unnecessary traffic that may be throwing off more false-positive data. You may need to tune when new software enters the environment.

From an attacker perspective, remember that attackers will change their techniques on a dime if need be. What was a "state-of-the-art" attack yesterday may be old news tomorrow, and attackers have shifted onto something else. Ad hoc, threat intelligence-based hunts should be validated over time. But don't forget, attackers have been known to resurrect techniques too! So keep those hunts on ice if you need to, but be prepared to hunt again if you notice a resurgence in legacy techniques.

You're Not Alone

When threat hunting, many analysts or teams may feel they are wading in an ocean of data trying to find a single drop of malware. If this is the case, call in a lifeboat! Remember your threat hunts can easily be enriched by third-party sources to help rule out false positives and focus on interesting leads. Your network data can be enriched with third-party IP lookups, geolocation and encrypted traffic metadata. Host-based data can also be enriched, typically with log detection and attacker technique overlays, which may help guide the team to more successful hunts. Furthermore, after visibility has been achieved, third-party tools may also be able to help augment your hunts via automated detection.

After you've acquired third-party data, your threat hunters can further enrich data by utilizing a link analysis tool. Link analysis tools, which help visualize and display relationships, can be instrumental in identifying correlations between internal vs. external or host vs. network data points. Link analysis capabilities are often built into your third-party sources, or may be offered as standalone tools as well.

Closing Thoughts

Unfortunately, some organizations are nervous when they begin threat hunting, thinking their teams are not ready to shift to a proactive stance. There is a constant, sometimes untrue, belief that "the attackers are already inside the house," so why not work to get them out? Even worse, management often fears that if it allows threat hunting to take place, "what else will the hunters find?"

These mindsets will never lead to effective security, as the attackers will always have the upper hand in confidence. Furthermore, organizations scared of what may be uncovered are robbing the security team of the opportunity to truly understand the environment and thus protect it better. It's time to break out of this mold and start securing our enterprises effectively.

If your organization finds itself constantly stuck in a reactive stance, unable to climb the mountain of alerts and false positives, it may be time to consider adding proactive threat hunting exercises to your security program. Proactive threat hunting allows your team to begin exploring the environment and discovering the weaknesses that could be exploited. Additionally, the organization can work to get these weaknesses patched and eliminate those attack surfaces.

As we mentioned earlier, good hunt teams will model their approach after known and modern attack techniques. Even the best searches may come up empty, however. This is good news—it means you may not have been compromised by a particular actor or attack technique! What your team did achieve during the process was the second objective: establishing a baseline and an understanding of the environment.

Lastly, remember that hunts are temporal. Attackers may recycle or retire techniques, and that hunting is a never-ending, always-learning practice. Hunts are never finished—they simply have point-in-time results. As your program grows and your hunting team develops a better understanding of its environment, you'll find your security program will be richer and stronger.

Start hunting!

About the Author

Matt Bromiley is a SANS Digital Forensics and Incident Response instructor, teaching Advanced Digital Forensics, Incident Response, and Threat Hunting (FOR508) and Advanced Network Forensics: Threat Hunting, Analysis, and Incident Response (FOR572), and a GIAC Advisory Board member. He is also a principal incident response consultant at a major incident response and forensic analysis company, combining experience in digital forensics, incident response/triage and log analytics. His skills include disk, database, memory and network forensics, as well as network security monitoring. Matt has worked with clients of all types and sizes, from multinational conglomerates to small, regional shops. He is passionate about learning, teaching and working on open source tools.

Sponsor

SANS would like to thank this paper's sponsor: