A Framework for Cyber Threat Hunting Part 1: The Pyramid of Pain

A Framework for Cyber Threat Hunting Part 1: The Pyramid of Pain

July 23, 2015 by
Sqrrl Team

A Framework for Cyber Threat Hunting Part 1: The Pyramid of Pain

While rule-based detection engines are a strong foundation for any security organization, cyber threat hunting is a vital capability for security organizations to have in order to detect unknown advanced threats. H unting goes beyond rule-based detection approaches and focuses on proactively detecting and investigating threats.

Cyber hunting "trips" are hypothesis-driven, utilizing an initial question or hypothesis (e.g., a group of executives are traveling to China to conduct business negotiations; they are at high risk of compromise) to engage on an iterative, exploratory search through cybersecurity datasets. Hunting trips focus on collecting Indicator(s) of Compromise (IoC) to find adverasaries, and can provide a strong basis for how to form a hypothesis. Any hunt can and should take advantage of advanced statistical and machine learning techniques to help the analyst predict where to begin and how to proceed. In this blog we will discuss the different types of IoCs that can be used as trailheads.

This post focuses on how to think about IoCs. There are a wide variety of IoCs ranging from basic file hashes to hacking Tactics, Techniques and Procedures (TTPs). Sqrrl Security Architect, David Bianco, uses a concept called the Pyramid of Pain to categorize IoCs. The pyramid organizes IoCs in two ways:

1. How difficult (painful) is it to collect and apply the IoC to cyber defenses? Malicious hash values and IP addresses are relatively easy to acquire and integrate into security tools. TTPs are more difficult to identify and apply, as most security tools are not well suited to take advantage of them.
2. How much pain can the IoCs inflict on cyber adversaries? It is relatively easy for an adversary to obfuscate malware code and change the hash values. IP addresses can be dynamically changed with low cost. TTPs are sticky and expensive for an adversary to change. As a result, security tools that leverage TTPs can inflict more pain on an adversary.

Details on the different levels within the Pyramid of Pain are provided below.

The Pyramid of Pain, originally developed by David Bianco: http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html

Let's start by simply defining the types of indicators that make up the pyramid:

1. Hash Values: SHA1, MD5 or other similar hashes that correspond to specific suspicious or malicious files. Often used to provide unique references to specific samples of malware or to files involved in an intrusion. It is so easy for hash values to change, and there are so many of them around, that in many cases it may not even be worth tracking them.

2. IP Addresses: IP addresses are quite literally the most fundamental indicator, but if they are using an anonymous proxy service like Tor or something similar, they may change IPs quite frequently and never even notice or care.

3. Domain Names: This could be either a domain name itself (e.g., "evil.net") or maybe even a sub- or sub-sub-domain (e.g., "this.is.sooooo.evil.net"). They must be registered, paid for (even if with stolen funds) and hosted somewhere. That said, there are a large number of DNS providers out there with lax registration standards.

4. Network Artifacts: In practice these are pieces of the activity that might tend to distinguish malicious activity from that of legitimate users. Typical examples might be URI patterns, C2 information embedded in network protocols, etc.

5. Host Artifacts: Observables caused by adversary activities on one or more of your hosts that would distinguish malicious activities from legitimate ones. These can be any distinctive identifier such as be registry keys or values known to be created by specific pieces of malware, files or directories dropped in certain places, etc.

6. Tools: Software used by the adversary to accomplish their mission. Mostly this will be things they bring with them, rather than software or commands that may already be installed on the computer. This would include utilities designed to create malicious documents for spearphishing, backdoors used to establish C2 or password crackers or other host-based utilities they may want to use post-compromise.

7. Tactics, Techniques and Procedures (TTPs): At the apex of the pyramid is how the adversary goes about accomplishing their mission, from reconnaissance all the way through data exfiltration and at every step in between. When you detect and respond at this level, you are operating directly on adversary behaviors, not against their tools. "Spearphishing" is a common TTP for establishing a presence in the network. "Spearphishing with a trojaned PDF file" or "… with a link to a malicious .SCR file disguised as a ZIP" would be more specific versions.

A key takeaway from Bianco's Pyramid of Pain is that TTPs are the most valuable indicators. TTPs reflect an attacker's behavior, and behavior requires a significant time and monetary investment to modify. However, TTPs are also difficult to model and detect using traditional tools. Unlike many other indicators, TTPs are only recognizable once you have been able to piece together the narrative of an attack. In the upcoming blogs we will demonstrate how cyber hunting and Sqrrl's Linked Data Analysis approach is uniquely equipped to model and detect TTPs. In part 2 of this blog series we will focus specifically on how security organizations can build intelligence-driven hunting loops to power TTP detection.