Hunt Evil

Your Practical Guide to Threat Hunting

Includes checklist, scorecard and examples

sqrrl

Chapters

Part 1

Setting Up Your Threat Hunting Program

CHAPTER 1

Intro to Hunting – What it is, Why It's Important, And Some Common Myths

You might have heard a lot of buzz around this topic of "Threat Hunting" and want to try your hand at proactive detection. Great! But how does one actually go about building a hunting program?

While there are a number of great resources available about what hunting is and how it can assist you, it might be challenging to cross over from the realm of the theoretical into the practical. As any hunter will tell you, orientation and planning is one of the critical aspects of effective threat hunting. This guide will help you orient and plan by laying out some basic tips and instructions on how to direct your hunting activities. It will also give you direction on how to practically carry them out using a variety of hunting techniques.

To begin, let's clarify what threat hunting is: Threat hunting is the human-driven, proactive and iterative search through networks, endpoints, or datasets in order to detect malicious, suspicious, or risky activities that have evaded detection by existing automated tools.

Threat hunting has been around for a while, but it has only recently become a focus of modern enterprise Security Operation Centers (SOCs). Hunting can revolutionize the threat detection efforts of an organization, and many have already recognized that proactive hunting needs to play a role in their overall detection practices (a common mantra one often hears is "prevention is ideal but detection is a must"). According to a recent survey on threat hunting conducted by the SANS institute, 91% of organizations report improvements in speed and accuracy of response due to threat hunting. It's clearly worth your time, but it's also worth knowing what exactly you're investing in. Before going any further, let's take a look at 3 common myths about hunting that will help clarify what it is.

3 Common Myths About Hunting

1 Hunting can be fully automated

Hunting is not a reactive activity. If the main human input in a hunt is remediating the result of something that a tool automatically found, you are being reactive and not proactive. You are resolving an identified potential incident, which is a critically important practice in a SOC, but not hunting.

Hunting requires the input of a human analyst and is about proactive, hypothesis-based investigations. The purpose of hunting is specifically to find what is missed by your automated reactive alerting systems. An alert from an automated tool can certainly give you a starting point for an investigation or inform a hypothesis, but an analyst should work through an investigation to understand and expand on the context of what was found to really get the full value of hunting. To put this another way, hunters are the network security equivalent of beat cops; they search for anomalies by patrolling through data, rather than investigating a call in from dispatch.

2 Hunting can only be carried out with vast quantities of data and a stack of advanced tools

Though it may seem like a new term, security analysts across a variety of sectors have been hunting for years. Basic hunting techniques can still be very useful and effective in helping you find the bad guys (e.g. you can perform basic outlier analysis, or “stack counting”, in Microsoft Excel). An analyst who wants to begin threat hunting should not hesitate to dive into some of the basic techniques with just simple data sets and tools. Take advantage of low hanging fruit!

Of course, having purpose-built tools like a Threat Hunting Platform can help you hunt at scale and simplify the more advanced hunt procedures. Sqrrl’s Threat Hunting Platform has been specially created to make the process of fusing different data sets together and leveraging more advanced techniques significantly more simple.

3 Hunting is only for elite analysts; only the security 1% with years of experience can do it

As you’ll learn, there are many different hunting techniques that have differing levels of complexity. However, not all these techniques take years to master. Many of the same analysis techniques used for incident response and alert investigation and triage can also be leveraged for hunting. The key to getting started is simply knowing what questions to ask, and digging into the datasets related to them. You learn to hunt by doing it, so if you’re an analyst who has never hunted before, don’t be afraid to dive in.

These Are All Good Points to Clear Up

It is also important to keep in mind that successful hunting is tied to capabilities in three different areas:

Planning, preparation, and process

Experience, efficiency, and expertise

Tools, techniques, and technology

A complete project (successful threat hunting)

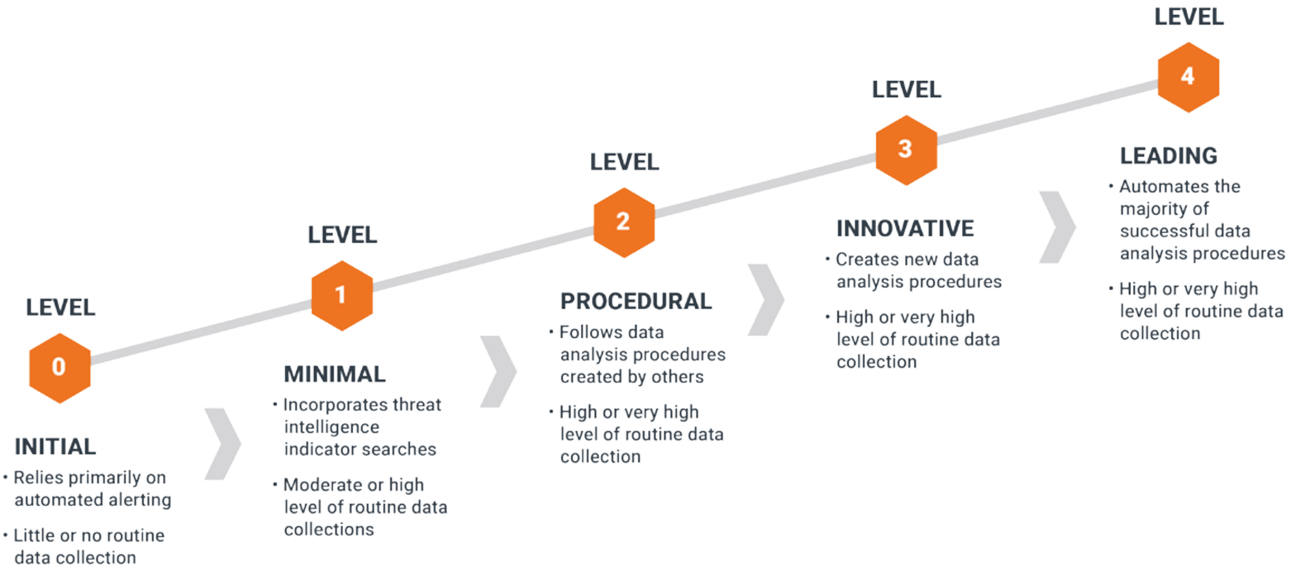
This book will touch on each of these categories to different extents, but keep in mind that they are all related and interconnected. Making sure you have a grasp on each one will bring you success in defending your organization.

> Now let's dive into the first and arguably most important category, your hunting process, and take a look at what should be your first step in creating or expanding your hunting capabilities: determining your hunting maturity.

CHAPTER 2

Determining Your Hunting Maturity

As mentioned, there are many different kinds of techniques and practices you can pursue in hunting. Your hunting maturity is a measure of what kinds of techniques and data you can work with. To help assess your current hunting capabilities and determine how you should be aiming to grow them, we've developed the Hunting Maturity Model (HMM).


LEVEL
0
INITIAL
• Relies primarily on automated alerting
• Little or no routine data collection
LEVEL
1
MINIMAL
• Incorporates threat intelligence indicator searches
• Moderate or high level of routine data collections
LEVEL
2
PROCEDURAL
• Follows data analysis procedures created by others
• High or very high level of routine data collection
LEVEL
3
INNOVATIVE
• Creates new data analysis procedures
• High or very high level of routine data collection
LEVEL
4
LEADING
• Automates the majority of successful data analysis procedures
• High or very high level of routine data collection

The Hunting Maturity Model describes five levels of an organization's proactive detection capability. Each level of maturity corresponds to how effectively an organization can hunt based on the data they collect, their ability to follow and create data analysis procedures (DAP), and their level of hunting automation. The HMM can be used by analysts and managers to measure current maturity and provide a roadmap for improvement. Often these improvements focus on a combination of tools, processes, and personnel.

If you want to determine your current level of hunting maturity, below is a list of questions you can answer to find out. You can then take your maturity level and align it to our suggestions about where you should be focusing your efforts next.

Basic Requirements

1. Do you have automated security alerting (SIEM, IDS, etc)?
2. Do you already have a dedicated incident detection or response team(s)?

Okay, you've got the basics covered

If you answered no to any of these, stop and see the section that says "Getting Started" on the next page.

Minimal Capability

1. Do you routinely collect security data from all three data domains (network, host, & application logs) into a centralized repository?
2. Do you utilize threat intelligence to drive detection (open or closed source)?
3. Do analysts in your SOC leverage Indicators of Compromise (IoCs) from reports?

Alright, you can bag some prey

If you answered no to any of these, stop and see HM 0 on the next page.

Procedural Approach

1. Do analysts in your SOC follow published hunting procedures to find new security incidents?
2. Do analysts in your SOC hunt on a regular recurring schedule: daily, weekly, etc?
3. Do you have designated hunters in your SOC or a set rotation of analysts who hunt so that there is always some proactive detection effort being carried out?

Good, you can carry out some real hunting

If you answered no to any of these, stop and see MM1 on the next page

Innovative Practices

1. Are your hunters utilizing a variety of data analysis techniques and applying them to identify malicious activity?
2. Do your hunters develop or publish original hunting procedures adapted from hunts they carry out in your environment?
3. Are you collecting security data tailored to your environment and your hunting practices?
4. Do you utilize a specialized threat hunting platform to facilitate streamlined hunting processes and collaboration in your hunt team?

Great, you're ahead of the pack!

If you answered no to any of these, stop and see HM2 on the next page.

Leading Programs

1. Are you automating successful hunting procedures/using the outputs of your hunts to improve alerting or automated detection efforts?
2. Do you employ data science techniques to support your hunting procedures and help isolate anomalies in large quantities of data?
3. Do you have a methodology for scaling your ability to carry out the hunting procedures you are continually creating?

Awesome, your hunt program is cutting-edge!

If you answered no to any of these, stop and see HM3.

Based on Your Results, What Improvements Should You Focus On?

Getting Started

If you are at this stage, focus on building a set of core security capabilities by establishing a formal SOC:

- Acquire an automated detection system (SIEM, IDS, etc.)
- Create a centralized logging system and start collecting logs (e.g. web proxy, firewall, switches, routers, Bro logs, host endpoint alerts, event logs, AD logs, etc)
- Establish a specialized incident response team (even if it is only a single analyst) which can perform alert resolution and incident investigation
- Acquire external signature feeds and intel feeds that can compliment your automated detection

At HM0 > Next step: Move to begin hunting

If you have a basic automated detection system and an analyst or team who can perform incident response, you are ready to begin to build out an effective hunting program. Focus on these points to grow your maturity:

- Good data is the basis of good hunting, ensure regular collection of at least some security data, at least one sources from each data domain (network, host and application)
- Analysts should practice some basic hunting, such as searching for key indicators to find threats in specific datasets

At HM1 > Next step: Develop ability to identify and carry out existing hunting procedures

At HM1, you can do some basic searching and hunting. This is a great start. To move to the next level of hunting maturity, focus on these points:

- Find and identify published hunting procedures you want to carry out on your network (If you're not sure where to begin, we recommend threathunting.net)
- Increase the scale of your data collection to include input data required to carry out published hunting procedures that you want to pursue
- Develop a schedule for applying these procedures on a regular basis

At HM2 > Next step: Develop ability to create new procedures

At HM2, you can follow procedures that are outlined and created in other places, which can be highly useful for protecting your organization against threats you have prioritized. The primary focus for moving to HM3 should be:

- Create a hunt team that includes both security and data analysis expertise, which can understand and apply a variety of different types of data analysis and hunting techniques
- Begin crafting new hunting procedures based on the security concerns of your organization and the threats that you have seen in the past

At HM3 > Next step: Develop automation for current procedures

A SOC at the “Innovative” level is one that is hunting in an advanced way. Consider sharing the tested and true hunts you develop with the hunting community! To move to the final level of hunting maturity:

- Create a process for fully automating the successful hunting procedures you develop. This will ensure that your hunters are not wasting time by repeating hunts, but always finding new things to hunt for.

At HM4 > Next step: Onward and upward!

If you are at HM4, you are running a cutting edge hunting program. From here, your focus is ever onward and upward, especially in growing your team in scale and efficiency, as well as expanding to meet the growing security concerns and needs of your organization.

Of course, the Hunting Maturity Model is just a prescriptive model, and many organizations will sometimes be at varying levels of capabilities: excelling at some criteria and less advanced in others. For example, you might have a tremendous log collection capability, but you might not be utilizing any hunting procedures. This is OK, you still have taken care of some critical aspects of hunting! Just try to focus on shoring up the areas that you are lacking in before you try to expand your other capabilities. If you’re just a fledgling hunting program, fear not! There’s a lot of room to grow and every step will provide you with tangible, actionable benefits!

You now have a better idea of where you stand in the grand scheme of hunting organizations. Now let’s take a look at how you can evaluate the hunting practices that you’re carrying out or will soon be developing, with some basic metrics for measuring performance.

CHAPTER 3

Metrics for Measuring Your Hunting Success

Clearly, hunting has its merits, but it's not magic. It's a process that requires people, technology and knowledge.

Inevitably, you will be beholden either to yourself or to others to show that what you're doing is actually having some effect. So how can you show that hunting is worth the investment in your own organization? How can you keep track of how you're doing and determine what areas of hunting you need to improve?

Below is a list of 10 key metrics, developed by veteran hunter Jack Crook, that you can use to evaluate your hunting activities. Some will be easier to empirically keep track of than others. What's important is that you try to measure them in whatever way you can so as to gain visibility into your progress.

Key Metrics	Why It's Important / What to Look For
1. **Number of incidents by severity**	You will never be able to know for certain how many incidents are lurking in your network until you find them, but ultimately keeping track of the rate at which you find incidents is a worthy metric to maintain context.
2. **Number of compromised hosts by severity**	Measuring the trend of how many hosts are discovered as compromised over time can help orient analysts to the state of endpoint security on their network. This can include hosts that have had misconfigured security settings on them.
3. **Dwell time of any incidents discovered**	Whenever possible, try to determine how long discovered threats have been active on your network. This can help you determine if there are steps of the kill chain (or other attack model) you may be focusing on too much. Dwell time has 3 metrics: time from infection until detection, time from detection to investigation, and time from investigation to remediation.
4. **Number of detection gaps filled**	One high-level goal of hunting is to create new automated detections -- identifying and filling detection gaps should be part of the team's mission.
5. **Logging gaps identified and corrected**	Gaps in logging or data collection can make it difficult for a SOC to maintain awareness and context, so trying to identify and improve any existing gaps should be an important actionable metric for a hunt team.

6. **Vulnerabilities identified**

   Vulnerabilities can lead to exploitation and exploitation can lead to compromise -- in other words, identifying vulnerabilities is important. It's always useful to keep track of how many of these you are uncovering.

7. **Insecure practices identified and corrected**

   Insecure practices can lead to unauthorized access and unauthorized access can lead to incidents -- identifying insecure practices can prevent future incidents.

8. **Number of hunts transitioned to new analytics**

   Since you want to create new automated detections, your team should try to transition each hunt into automated detection. Ideally you would want the ratio here to be 1:1. For every successful hunt you carry out you should be attempt to create a new analytic, update a rule, or at least log a new IoC.

9. **False positive rate of transitioned hunts**

   Once you discover a successful way to find something and create a rule or analytic to automate that process, it is useful to keep track of how many false positives have been created by those automated analytics, to see if they require improvements.

10. **Any new visibility gained**

    In addition to discovering an incident and creating new threat intel, a hunt can inform analysts about their own networks, including misconfigurations, and identify friendly intelligence that can be highly useful in future investigations.

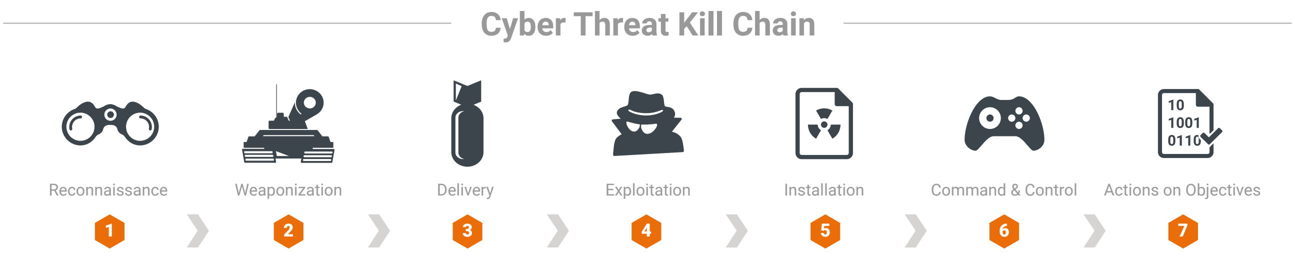
Impress the C-suite

Keeping track of these kinds of metrics is one of the major benefits of hunting from a managerial standpoint. Instead of having to wait until your organization is compromised before receiving more resources to grow your security team, hunting can help you show actionable improvements in initiatives that you undertake, as well as create a real, tangible profiles of threats that your organization might be facing. Some metrics, such as number of compromised hosts found and vulnerabilities identified, are critical pieces of information that should be compelling to any executive or board member.

It's important to note that when you start hunting for the first time, some of these metrics, like number of compromised hosts, might initially be very high. This is a kind of "start up bump" that all hunting programs will run into, because no previous proactive detection efforts have been undertaken. Although it may be overwhelming or seem problematic when looking at these metrics over time, you are finding new incidents you were not finding before but were always there anyway. This is decidedly a good thing.

CHAPTER 4

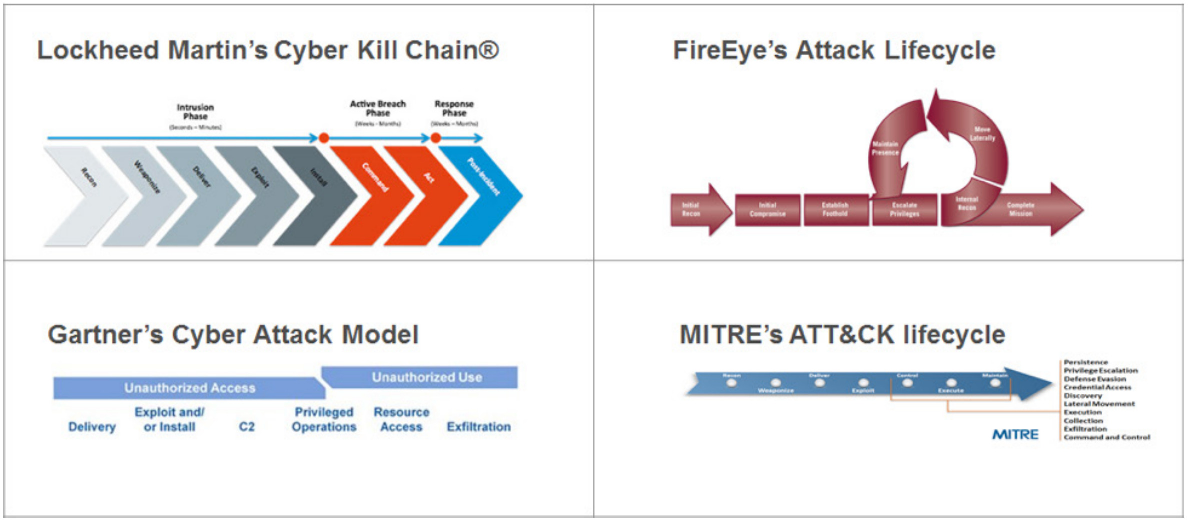
Determining What to Hunt For and How Often



So you have now determined what your hunting maturity level is and what metrics to use to chart your success, but how do you decide what to actually hunt for? While hunting approaches will vary from company to company, the three steps laid out below can help you kickstart your hunting program.

1 Choose Your Favorite Attack Model

There are many kinds of bad guys out there that may be trying to attack you for a variety of reasons, but the general progression of how most attacks are carried out tend to share many common elements. These are mapped out in what is known as the Cyber Kill Chain. There are several variations of the kill chain, all of which define what actions adversaries must complete in order to achieve their objective while operating within an enterprise network.

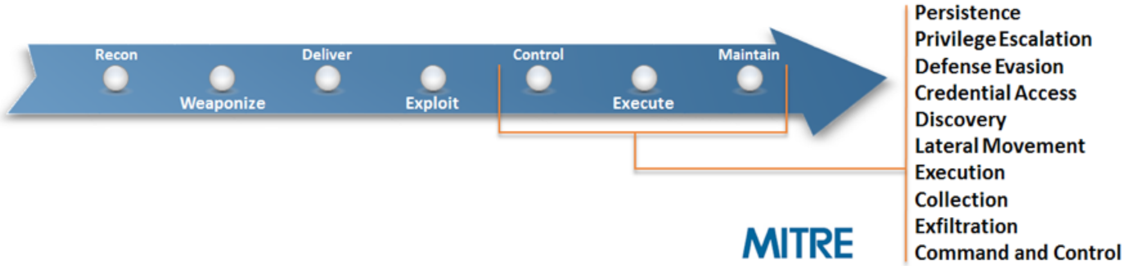


Choose what makes the most sense to you.

The kill chain will help you identity TTPs and attacker behaviors that you should hunt for. For this example, we will select and use MITRE's ATT&CK lifecycle.

2 Identify Most Concerning Activities

After selecting a model, the next step is to go through each of the phases in the model and identify attacker activities that you are most concerned with. Each phase in a model can include multiple categories of higher level tactics that an adversary might employ, which can then be broken down to a number of actual attacker activities, which you will hunt for.



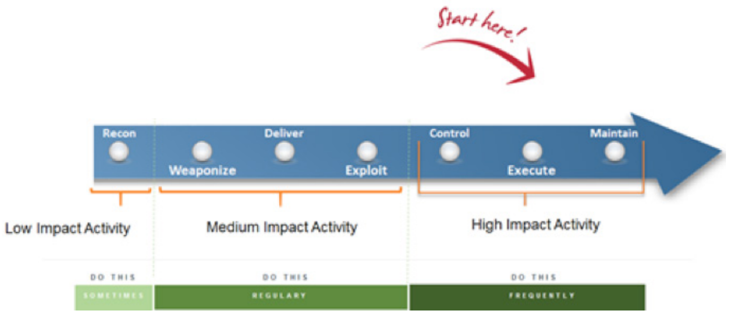
For example, the later stages (Control, Maintain, and Execute) of MITRE's seven-stage ATT&CK lifecycle include categories like lateral movement and data exfiltration, under which many kinds of activities can exist. **Here's an example list of potential attacker activities and techniques you might identify:**

- Malware Beaconing
- DLL Injection
- Pass the Hash (PtH)
- Shared Webroot
- DNS Tunneling

Make sure you are considering activities that are specific to your network environment and which assets you suspect an attacker would attempt to target. For example, a manufacturer may list potential attacker activities that are specific to their industrial control systems. You should also aim to try and hunt for TTPs you have previously not found, not the ones you can already detect. Leave those to your automated detection systems.

3 Build Your Threat Hunting Calendar

After creating a prioritized list of activities for each phase, the next step is to create your hunting calendar and set a cadence for the frequency of your hunts. It's important to start near the end of the kill chain, as these are the point where the attacker is about to achieve their objective. That means you want to stop those absolutely. Organize each of the various phases by low, medium and high impact activity.



For example, the rightmost stages of MITRE's seven-stage ATT&CK lifecycle (Control, Maintain, and Execute) can be considered High Impact activity. The higher impact activity the more that it should be hunted for on a more regular basis. Here's an example of weekly hunting sprints over two months:

Month 1

- Two weeks hunting High Impact Activity
- Two weeks Medium Impact Activity

Month 2

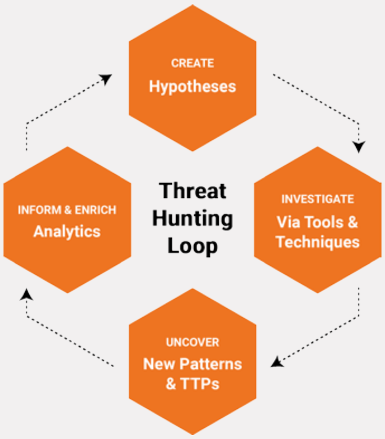
- Two weeks on High Impact Activity
- One week on Medium Impact Activity
- One week predicting attacks

> Once you have that all down, you will be able to determine what your hunting schedule is actually going to look like.

The Hunting Loop

Once you determine how often you want to hunt and what you want to hunt for, you're just about ready to actually begin hunting. But while the above model can help you formulate how to determine how often you will hunt, what is that practical process of carrying out a hunt in itself?

In order to answer this question, Sqrrl has developed the Threat Hunting Loop, which can guide an analyst in the tactical implementation of a hunt. We have written about this in a number of other places, so we won't go as in depth on it here. However, since a great deal of the hunting content we'll cover aligns to this loop, we'll briefly explain it.



The Hunting Loop breaks down into four steps:

1. A hunt starts with creating a hypothesis, or an educated guess, about some type of activity that might be going on in your IT environment. Hypotheses are typically formulated by analysts based on any number of factors, including friendly intelligence and threat intelligence, as well as past experiences.
2. A hunter follows up on hypotheses by investigating via various tools and techniques. We'll discuss tools and techniques in more detail below, but in general, analysts can use these to discover new malicious patterns in their data and reconstruct complex attack paths to reveal an attacker's Tactics, Techniques, and Procedures (TTPs).
3. Using manual techniques, tool-based workflows, or analytics, a hunter then aims to uncover the specific patterns or anomalies that might be found in an investigation. What you find in this step is a critical part of the success criteria for a hunt. Even if you don't find an anomaly or attacker, you want to be able to rule out the presence of a particular tactic or compromise. In essence, this step functions as the "prove or disprove your hypothesis" step.
4. Finally, successful hunts form the basis for informing and enriching automated analytics. Don't waste your team's time doing the same hunts over and over. If you find an indicator or pattern that could recur in your environment, automate its detection so that your team can continue to focus on the next new hunt. Information from hunts can be used to improve existing detection mechanisms, which might include updating SIEM rules or detection signatures. The more you know about your own network, the better you can defend it, so it makes sense to try to record and leverage new findings as you encounter them on your hunts.

CHAPTER 5

Top Considerations for Effective Tech

Now that you are starting to piece together a solid hunting process, let's pause for a moment to talk about the kinds of tools you'll need to use in hunting.

Although various types of tools can be used for hunting (e.g., SIEMs, purpose-built hunting platforms like Sqrrl, open source software, etc.), there are several questions that you should consider when picking a threat hunting tool. Though tools will vary greatly, there are generally 3 criteria that you should consider: how it assists in investigations, what analytics it can leverage, and how it deploys and deals with that data you'll be hunting through.

Below is a list of questions that can help inform your requirements for and selection of a hunting tool.

Investigation Capabilities

1. *Which of the standard hunting techniques does the tool generally enable you to carry out?*
   In the next section, we will cover a list of hunting techniques that you can apply in various detection situations. It's important that whatever tool you're considering be able to carry out most if not all of these techniques. Otherwise, you will be limiting the techniques at your disposal.
2. *How does the tool support the creation of hypotheses on which to base a hunt?*
   You should always be creating hypotheses yourself, so this isn't a necessity, but it's always helpful when a tool can help you come up with where to begin a hunt with some degree of priority.
3. *What ability does the tool have to import outside intelligence or custom indicators in order to assist analysts with the investigation of hypotheses?*
   This is critical. Having guidance from intelligence feeds or even your own friendly intelligence channeled into your data can revolutionize the way that you conduct your hunts and help you confirm with greater certainty when you think you've found something.
4. *What capabilities does the tool have that allow an analyst to pivot through different data sets?*
   Another critically important feature that a tool should have is the ability to pivot through data. You don't want to get caught up not being able to answer a question because the answer lies in a different part or type of the data that you can't access.
5. *How does the tool support the collection and storage of new Indicators of Compromise that might be found over the course of a hunt?*
   What's really going to make your hunt worthwhile is if you can take what you find in it and use it to improve your automated defenses. The first step of that is going to be exporting the indicators that you find.

Analytics Supported

6. *What kind of analytics does the tool support that will help you facilitate more streamlined proactive investigation?*
   It will be a great relief to have the backup of some analytics to assist you in finding and identifying anomalies that end up being malicious adversary activity, if not to simply add a degree of certainty to confirm a suspicion you might have.
7. *Does the tool enable the creation and customization of detection analytics?*
   Being able to create analytics will become immensely valuable as you customize your hunting process to your organization's needs, and depending on what you're hunting for.
8. *Does the tool utilize any machine learning or data science techniques?*
   Having machine learning capabilities isn't an absolute necessity, but the more it has the more you will ideally be able to rely on the fidelity of what the tool finds.

Deployment & Data

9. *What data sources does the tool support?*
   You should ideally be able to pivot through any data type that you might need or an investigation might require.
10. *To what extent is the tool able to scale its data storage capacity?*
    The amount of data that your tool can look through is Important. Can it search petabyte scale levels of data in real time? If it cannot, you might find yourself limited in terms of trying to scope the full extent of a major incident.
11. *Through what process does it ingest or stream data?*
    If the tool needs to intake data in order to analyze it, it's important to know how quickly and smoothly the process of bringing that data into it will be.
12. *What integrations with other security tools does it support?*
    Hunting is never an isolated practice. A hunting tool should integrate with automated detection systems, orchestration tools, and any preventative measures.

Part 2

Practical Hunting

CHAPTER 6

High Impact Activity to Hunt For

As we discussed in the first section, adversaries will come in many forms and will deploy a wide variety of different Tactics, Techniques and Procedures (TTPs). In order to defend yourself, you must know your enemy. Similarly to how you orient your overall hunting plan, the kinds of techniques you use to hunt will depend largely on what you're trying to defend against, which in turn will depend largely on what you're trying to protect.

Here are some high impact activities and TTPs that you can start hunting for

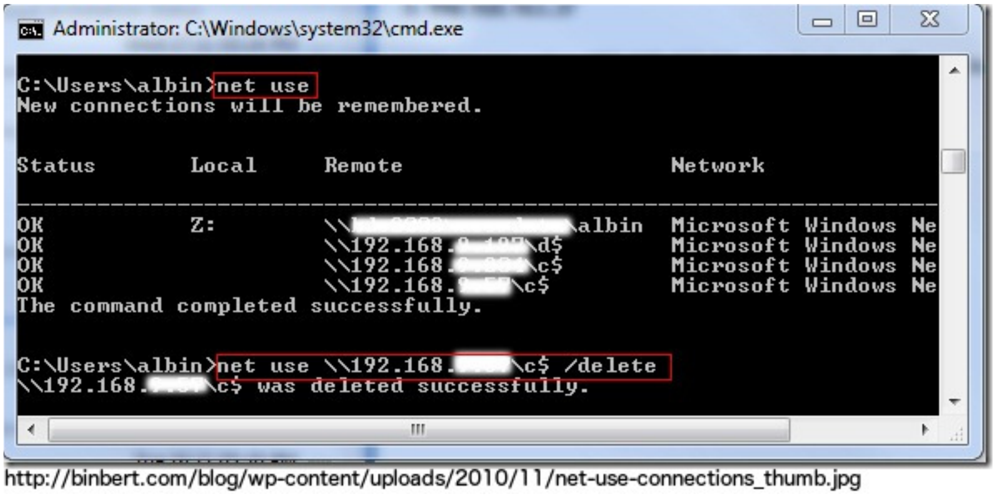
These TTPs are grouped by tactic categories from MITRE's ATT&CK Matrix™.

Internal Reconnaissance

How attackers determine where they're going

Host enumeration

Determines details about a local host, which includes establishing an understanding of local user context and local host configuration. User context lets you, as an attacker, know what user you are logged in as and what privileges are allotted to you. Local host configuration includes information about the host itself, including hostname and IP address.

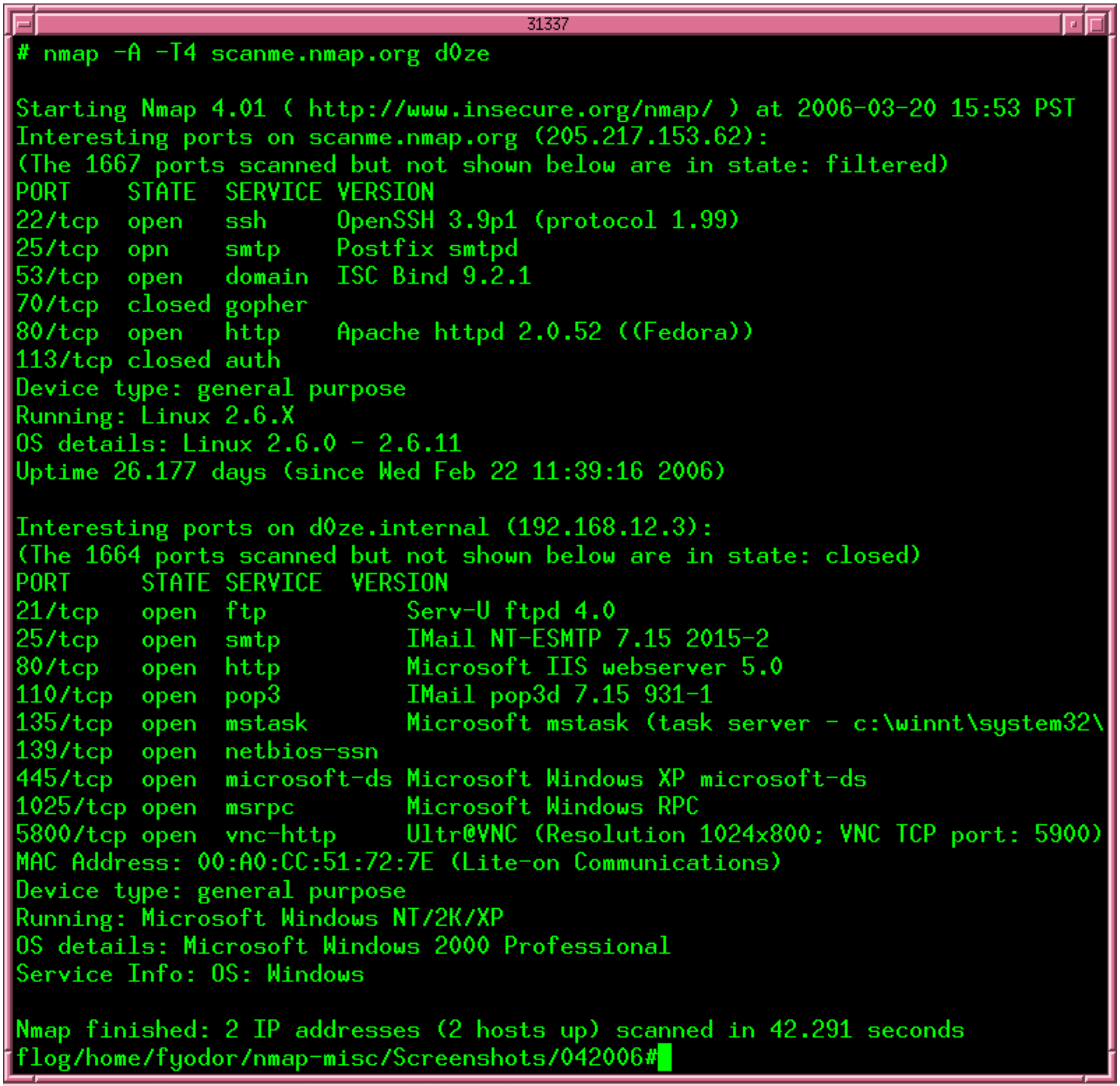


```
C:\Users\albin>net use
New connections will be remembered.

Status       Local     Remote                       Network

-------------------------------------------------------------------------------
OK           Z:        \\[redacted]albin    Microsoft Windows Ne
OK                     \\192.168.[redacted]d$           Microsoft Windows Ne
OK                     \\192.168.[redacted]c$           Microsoft Windows Ne
OK                     \\192.168.[redacted]c$           Microsoft Windows Ne
The command completed successfully.

C:\Users\albin>net use \\192.168.[redacted]c$ /delete
\\192.168.[redacted]c$ was deleted successfully.
```

http://binbert.com/blog/wp-content/uploads/2010/11/net-use-connections_thumb.jpg

Network enumeration

Establishes what other hosts are remotely accessible from the local host. Once attackers have compromised an initial host, they will need to determine how to move around the network and where they can go. Network enumeration lets you, as an attacker, see what access the host you are on has and what active connections there are to other systems and assets.



```
# nmap -A -T4 scanme.nmap.org d0ze

Starting Nmap 4.01 ( http://www.insecure.org/nmap/ ) at 2006-03-20 15:53 PST
Interesting ports on scanme.nmap.org (205.217.153.62):
(The 1667 ports scanned but not shown below are in state: filtered)
PORT    STATE  SERVICE VERSION
22/tcp  open   ssh     OpenSSH 3.9p1 (protocol 1.99)
25/tcp  opn    smtp    Postfix smtpd
53/tcp  open   domain  ISC Bind 9.2.1
70/tcp  closed gopher
80/tcp  open   http    Apache httpd 2.0.52 ((Fedora))
113/tcp closed auth
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.0 - 2.6.11
Uptime 26.177 days (since Wed Feb 22 11:39:16 2006)

Interesting ports on d0ze.internal (192.168.12.3):
(The 1664 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE      VERSION
21/tcp   open  ftp          Serv-U ftpd 4.0
25/tcp   open  smtp         IMail NT-ESMTP 7.15 2015-2
80/tcp   open  http         Microsoft IIS webserver 5.0
110/tcp  open  pop3         IMail pop3d 7.15 931-1
135/tcp  open  mstask       Microsoft mstask (task server - c:\winnt\system32\
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds Microsoft Windows XP microsoft-ds
1025/tcp open  msrpc        Microsoft Windows RPC
5800/tcp open  vnc-http     Ultr@VNC (Resolution 1024x800; VNC TCP port: 5900)
MAC Address: 00:A0:CC:51:72:7E (Lite-on Communications)
Device type: general purpose
Running: Microsoft Windows NT/2K/XP
OS details: Microsoft Windows 2000 Professional
Service Info: OS: Windows

Nmap finished: 2 IP addresses (2 hosts up) scanned in 42.291 seconds
flog/home/fyodor/nmap-misc/Screenshots/042006#
```

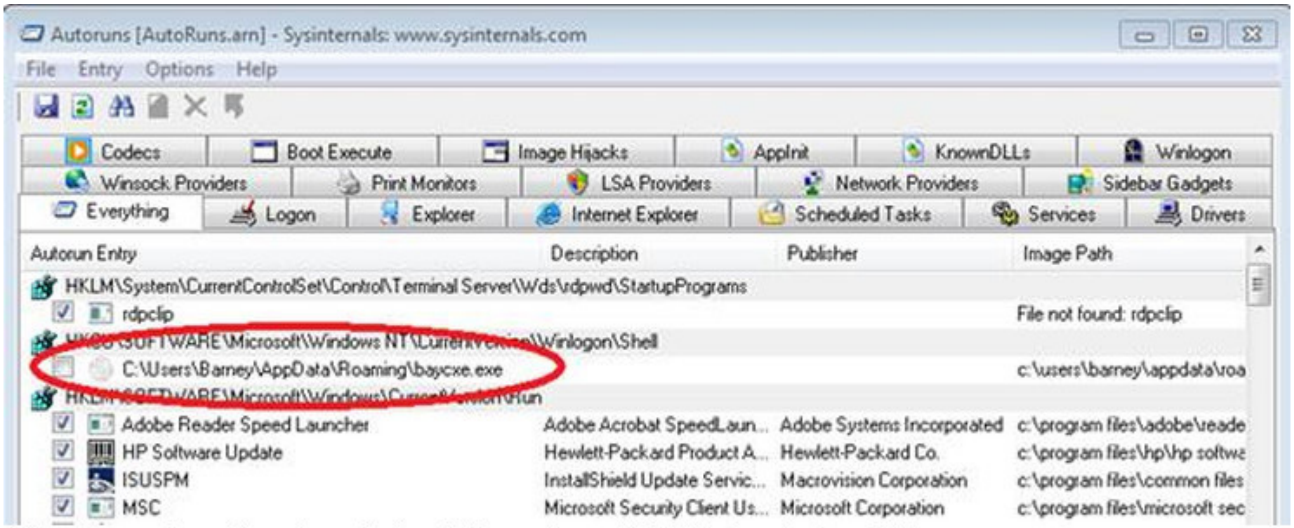
https://nmap.org/images/nmap-401-demoscan-798x774.gif

Persistence

How attackers survive a reboot and simple remediations

Scheduled Task Execution

This is the process of queueing up programs or scripts that can be operationalized at a later point. Tasks can be also scheduled remotely, assuming that the attacker has the authentication to use Remote Procedure Call (RPC). This allows attackers to run programs when a system starts up, or according to a schedule.



Autorun Entry	Description	Publisher	Image Path
HKLM\System\CurrentControlSet\Control\Terminal Server\Wds\rdpwd\StartupPrograms			
☑ rdpclip			File not found: rdpclip
HKCU\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell			
□ C:\Users\Barney\AppData\Roaming\baycxe.exe			c:\users\barney\appdata\roa
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run			
☑ Adobe Reader Speed Launcher	Adobe Acrobat SpeedLaun...	Adobe Systems Incorporated	c:\program files\adobe\reade
☑ HP Software Update	Hewlett-Packard Product A...	Hewlett-Packard Co.	c:\program files\hp\hp softwa
☑ ISUSPM	InstallShield Update Servic...	Macrovision Corporation	c:\program files\common files
☑ MSC	Microsoft Security Client Us...	Microsoft Corporation	c:\program files\microsoft sec

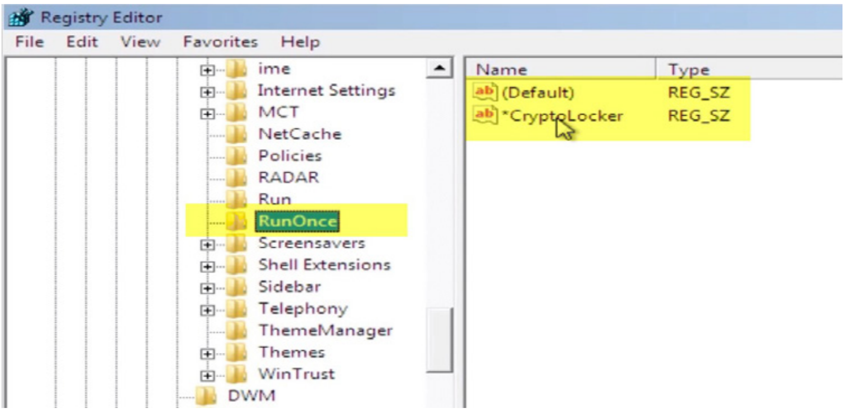
https://www.microsoft.com/security/assets/images/_security/sir/strategy/malware11.jpg

DLL Injection

Using this technique, adversaries will run malicious code by using another process to load and execute the code. This allows adversaries to hide malicious activity by incorporating it as part of a benign or routine process. It also allows attackers to access a system's process memory and permissions.

Registry modification

The additional values to the RUN and RUNONCE registry key allow for malware binaries to execute upon system boot and session login. This is the most common technique for persistence seen in the last decade.



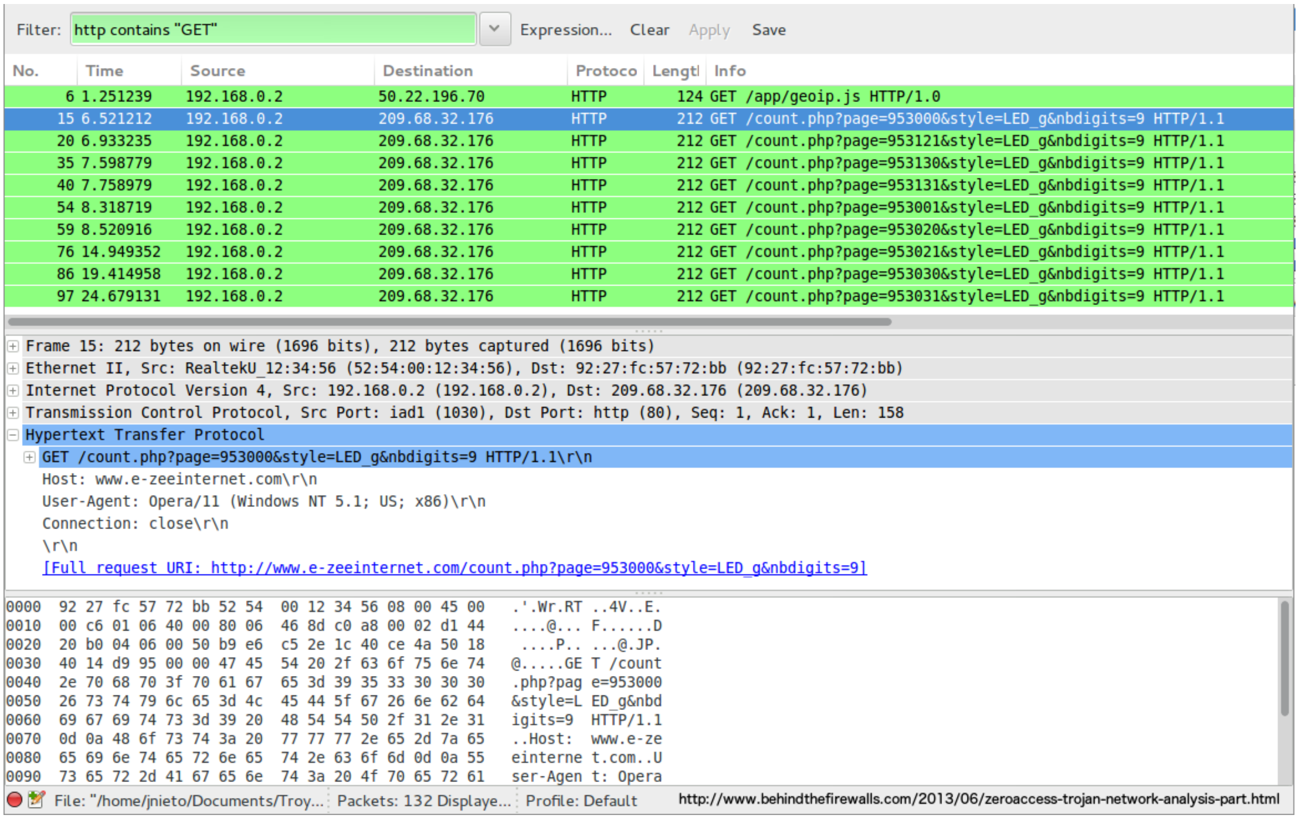
Name	Type
(Default)	REG_SZ
*CryptoLocker	REG_SZ

Command & Control

How attackers utilize their tools

Common Protocol, Common Port

Using this method, attackers will seek to hide in plain sight by blending in with routine network traffic. Oftentimes this takes the form of using HTTPS, DNS tunneling, and high-traffic ports to establish command and control.



Filter: http contains "GET"

```
No.   Time       Source        Destination    Protoco Lengtl Info
    6 1.251239   192.168.0.2   50.22.196.70   HTTP       124 GET /app/geoip.js HTTP/1.0
   15 6.521212   192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953000&style=LED_g&nbdigits=9 HTTP/1.1
   20 6.933235   192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953121&style=LED_g&nbdigits=9 HTTP/1.1
   35 7.598779   192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953130&style=LED_g&nbdigits=9 HTTP/1.1
   40 7.758979   192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953131&style=LED_g&nbdigits=9 HTTP/1.1
   54 8.318719   192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953001&style=LED_g&nbdigits=9 HTTP/1.1
   59 8.520916   192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953020&style=LED_g&nbdigits=9 HTTP/1.1
   76 14.949352  192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953021&style=LED_g&nbdigits=9 HTTP/1.1
   86 19.414958  192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953030&style=LED_g&nbdigits=9 HTTP/1.1
   97 24.679131  192.168.0.2   209.68.32.176  HTTP       212 GET /count.php?page=953031&style=LED_g&nbdigits=9 HTTP/1.1
```

```
⊞ Frame 15: 212 bytes on wire (1696 bits), 212 bytes captured (1696 bits)
⊞ Ethernet II, Src: RealtekU_12:34:56 (52:54:00:12:34:56), Dst: 92:27:fc:57:72:bb (92:27:fc:57:72:bb)
⊞ Internet Protocol Version 4, Src: 192.168.0.2 (192.168.0.2), Dst: 209.68.32.176 (209.68.32.176)
⊞ Transmission Control Protocol, Src Port: iad1 (1030), Dst Port: http (80), Seq: 1, Ack: 1, Len: 158
⊟ Hypertext Transfer Protocol
  ⊞ GET /count.php?page=953000&style=LED_g&nbdigits=9 HTTP/1.1\r\n
    Host: www.e-zeeinternet.com\r\n
    User-Agent: Opera/11 (Windows NT 5.1; US; x86)\r\n
    Connection: close\r\n
    \r\n
    [Full request URI: http://www.e-zeeinternet.com/count.php?page=953000&style=LED_g&nbdigits=9]
```

```
0000  92 27 fc 57 72 bb 52 54  00 12 34 56 08 00 45 00   .'.Wr.RT ..4V..E.
0010  00 c6 01 06 40 00 80 06  46 8d c0 a8 00 02 d1 44   ....@... F......D
0020  20 b0 04 06 00 50 b9 e6  c5 2e 1c 40 ce 4a 50 18    ....P.. ...@.JP.
0030  40 14 d9 95 00 00 47 45  54 20 2f 63 6f 75 6e 74   @.....GE T /count
0040  2e 70 68 70 3f 70 61 67  65 3d 39 35 33 30 30 30   .php?pag e=953000
0050  26 73 74 79 6c 65 3d 4c  45 44 5f 67 26 6e 62 64   &style=L ED_g&nbd
0060  69 67 69 74 73 3d 39 20  48 54 54 50 2f 31 2e 31   igits=9  HTTP/1.1
0070  0d 0a 48 6f 73 74 3a 20  77 77 77 2e 65 2d 7a 65   ..Host:  www.e-ze
0080  65 69 6e 74 65 72 6e 65  74 2e 63 6f 6d 0d 0a 55   einterne t.com..U
0090  73 65 72 2d 41 67 65 6e  74 3a 20 4f 70 65 72 61   ser-Agen t: Opera
```

File: "/home/jnieto/Documents/Troy... Packets: 132 Displaye... Profile: Default

http://www.behindthefirewalls.com/2013/06/zeroaccess-trojan-network-analysis-part.html

Uncommon Protocol, Uncommon Port

By utilizing this technique, adversaries bypass heavily monitored ports and send data through uncommon ports. This allows attackers to operate with a high degree of stealth, allowing them to evade detection from both human operators and routine detection systems.

Lateral Movement

How attackers move around in your network

Pass the Hash (PtH)

Using this technique, attackers will capture valid password hashes via a Credential Access technique and use that to authenticate themselves. This allows them to bypass using a user's cleartext password and enables them to perform actions on both local and remote systems.

Remote Desktop Protocol

A remote desktop allows a user to access a computer's desktop interface using a remote system. If a system's remote desktop protocol is enabled and an attacker knows their target's account credentials, the attacker can use that information to gain access to and exploit the target system.

Shared Webroot

If a firm has an internally accessible website or intranet network, attackers may upload malicious content to said website and then execute it using a web browser. Since the content is run under the permissions of the Web server process, this can result in the attacker gaining local or administrative privileges.

Path Interception

This technique entails placing an executable file in a specific path so that it is mistakenly run by a legitimate application. Examples of this include using unquoted paths, path environment variable misconfigurations, and search order hijacking. This allows adversaries to escalate their privileges if the executable is run as part of a higher privileged process.

Exfiltration

How attackers steal your data

DNS Tunneling

This allows an attacker to transfer encoded data inside of DNS queries. Attackers utilize this method to bypass common security controls (e.g., firewalls) and exfiltrate sensitive data.

SFTP/SCP Exfiltration

This allows for usage of SSL to hide details about the traffic. You will need a proxy that can break SSL to intercept and investigate this type of activity.

Consider what adversary techniques you want to focus on and research them as much as you can. Put yourself in the mindset of an attacker and determine how one would carry out each of these techniques if they were breaking into your network and attempting to access your critical assets. This will help you determine what data sets to look at and what techniques to deploy, which we will cover in the next section.

Do not be overwhelmed! Hunting is, in part, an exercise in prioritization. For every insidious tactic that an attacker can use to compromise a system, there is a technique that a stalwart defender can use to repel them. As in all battles, the advantage of being the defender is that you are on your home turf. It is your network, and if you know it and know how to patrol it, you can foresee and stop even the most sophisticated adversaries.

For a more comprehensive list of example hunts that look for various TTPs, check out threathunting.net.

CHAPTER 7

Four Primary Threat Hunting Techniques

Okay, you have your hunting procedures and you have your tools squared away. Now let's talk about the actual practice of hunting, what techniques are in your arsenal, and some examples of how you can proactively find the adversaries lurking in your network. This is by no means an exhaustive list, these are just a set of general techniques that can be applied in different ways.

Techniques

Searching

The simplest method of hunting, searching is the process of querying data for specific results or artifacts, and can be performed using many tools. Searching requires finely defined search criteria to prevent result overload. There are two primary factors to keep in mind when carrying out a search: searching too broadly for general artifacts may produce far too many results to be useful, and searching too specifically for artifacts on specific hosts may produce fewer results than may be useful.

Clustering

Clustering is a statistical technique, often carried out with machine learning, that consists of separating groups (or clusters) of similar data points based on certain characteristics out of a larger set of data. Hunters may use clustering for many applications, including outlier detection, due to the fact that it can accurately find aggregate behaviors, such as an uncommon number of instances of a certain occurrence. This technique is most effective when dealing with a large group of data points that do not explicitly share immediately obvious behavioral characteristics.

Grouping

Grouping consists of taking a set of multiple unique artifacts and identifying when multiple of them appear together based on specific criteria. The major difference between grouping and clustering is that in grouping your input is an explicit set of items that are already of interest. Discovered groups within these items of interest may potentially represent a tool or a TTP that an attacker might be using. An important aspect of using this technique consists of determining the specific criteria used to group the items, such as events having occurred during a specific time window. This technique works best when you are hunting for multiple, related instances of unique artifacts, such as the case of isolating reconnaissance commands that were executed within a specific timeframe.

Stack Counting

Also known as stacking, this is one of the most common techniques carried out by hunters to investigate a hypothesis. Stacking involves counting the number of occurrences for values of a particular type, and analyzing the outliers or extremes of those results. The effectiveness of this technique is generally diminished when dealing with large and/or diverse data sets, but it is most effective with a thoughtfully filtered input (such as endpoints of a similar function, organizational unit, etc.). Analysts should attempt to understand input well enough to predict the volume of the output. For example, if you are given a dataset containing 100k endpoints, stack counting the contents of the Windows\Temp\ folder on each endpoint across an enterprise will produce an enormous result set. Friendly intelligence can be used to define filters for your input.

Machine Learning Techniques

In addition to standard hunting techniques like those listed above, you will find that many investigations and procedures can be enhanced and carried out using various machine learning or data science powered techniques. These techniques can involve creating frameworks of feedback given to automated classification systems. This is known as supervised machine learning, which uses labeled "training data" to condition algorithms to make predictions about unlabeled data. This new, unclassified data is what you want the machine to label correctly (based on the training data).

It's not an absolute necessity that you be able to leverage machine learning techniques in your hunting, but you should be aware of the role they can play, as you will see them referenced in many places including in the rest of this guide. The best hunting tools, such as Sqrrl, should be able to provide you with prebuilt machine learning techniques you can leverage as part of an investigation workflow.

Datasets

The techniques that you use are only a part of planning out your hunt and knowing what you can have at your disposal. You can't hunt if you don't have the right data, but what is the right data? The answer to that question will depend on what you're looking for, but below is a general list of datasets that lend themselves well to hunting and security activities in general:

Endpoint Data

Process execution metadata Contains information on processes run on specific hosts. Critical metadata associated with process execution includes command-line commands/arguments and process filenames and ID.

Registry access data Contains data related to registry objects, including key and value metadata.

File data Information on stored files and artifacts kept on a local host. This can include when files were created or modified, as well as size, type, and storage location information.

Network data Identification of the parent process for a network connection.

File prevalence Information on how common a file is in your environment.

Network Data

Network session data Contains information on network connections between hosts. Critical metadata associated with network connections including the source IP address, destination IP address, destination port, start time of the connection, and end time/duration of the connection. This includes Netflow, IPFIX, and similar data sources.

Bro logs A widely recommended network monitoring tool that collects connection-based flow data and application protocol metadata (HTTP, DNS, SMTP), specialized for security application.

Proxy logs HTTP data that contains information on outgoing web requests, including Internet resources that internal clients are accessing.

DNS logs

Contains data related to DNS domain resolution activity, including domain-to-IP address mappings and identification of internal clients making resolution requests.

Firewall logs

Connection data that contains information on network traffic at the border of a network, focused on blocked connections.

Switch and Router logs

Internal netflow, also known as east/west traffic, in your environment that shows what is going on inside the network behind your perimeter security.

Security Data

Threat Intelligence

A broad category of information that includes the indicators and TTPs used by attackers, as well as the operations and campaigns they carry out.

Alerts

The automated warnings or notifications created by correlation engine tools like a SIEM or IDS, indicating that a given rule set was violated or certain pattern identified, which might indicate a potential incident.

Friendly Intelligence

Another broad category of information about an organization's own IT infrastructure, security ecosystem, critical assets, employee information, and business processes. Friendly intel helps hunters orient and understand the environment in which they are hunting and contextualize their investigations.

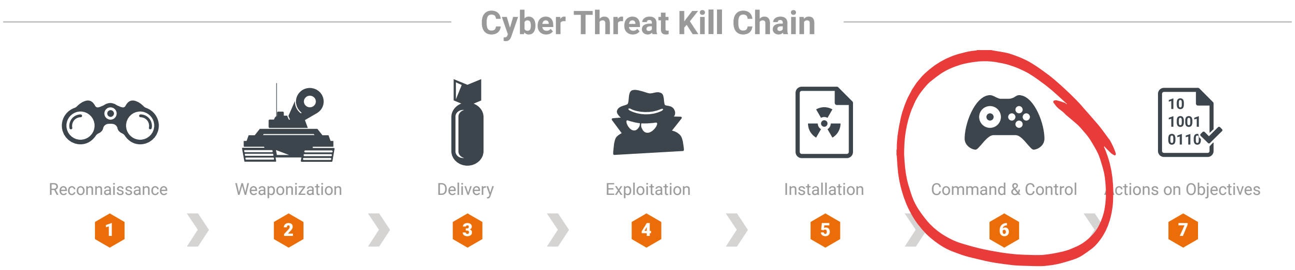
CHAPTER 8

Walkthrough: Hunting for Command & Control

By now you're determined how to map out how you're going to be carrying out your hunts and what you'll be hunting for, and figured out what techniques and resources are at your disposal. It's time to finally dive in and start finding evil.

Even with all this information, the prospect of hunting might still be a little daunting. Perhaps the most effective way of learning how to actually hunt is to learn from examples of applied hunts. In the next two sections, you'll see 2 example hunt walkthroughs looking for 2 different adversary techniques that you can look through for guidance. Give these a try on your own! You should now have all the information you need at your disposal.

Hunting for Command & Control


Cyber Threat Kill Chain

Command & Control (C2) Overview

Attackers generally build Command and Control (C2) channels into common protocols (ComPro) or custom protocols (CusPro). This enables remote access for attackers into target networks. A few examples of common protocols include HTTP/S, SSL/TLS, or DNS. Custom protocols are harder to predict, but include techniques such as encrypting packet data with an XOR cipher. Just like with protocols, attackers generally use common network ports (ComPor) or uncommon network ports (UncPor) for their C2 channels. Examples of standard ports include 80/TCP (HTTP), 443/TCP (SSL/TLS), 53/UDP (DNS). Uncommon ports are difficult to predict, but they typically deviate from ports registered with IANA. Attackers can use any combination of protocols and ports, including:

- Common Protocol + Common Port
- Common Protocol + Uncommon Port
- Custom Protocol + Common Port
- Custom Protocol + Uncommon Port

Example Hypothesis

Hypothesis

Attackers may be operating on a C2 channel that uses a common protocol on a common network port

- Look for unique artifacts pertinent to the protocol you are interested in. For example, if you are interested in identifying C2 in HTTP traffic, then you might consider looking for anomalous domains/URLs/User-Agent strings.

Datasets to Explore

Datasets used to hunt for C2 depends on what you are hunting for. For identifying use of custom protocols, focus primarily on network session metadata, including:

- Netflow ("flow" data in general)
- Firewall logs (should log allowed / accepted packets)
- Bro Conn log

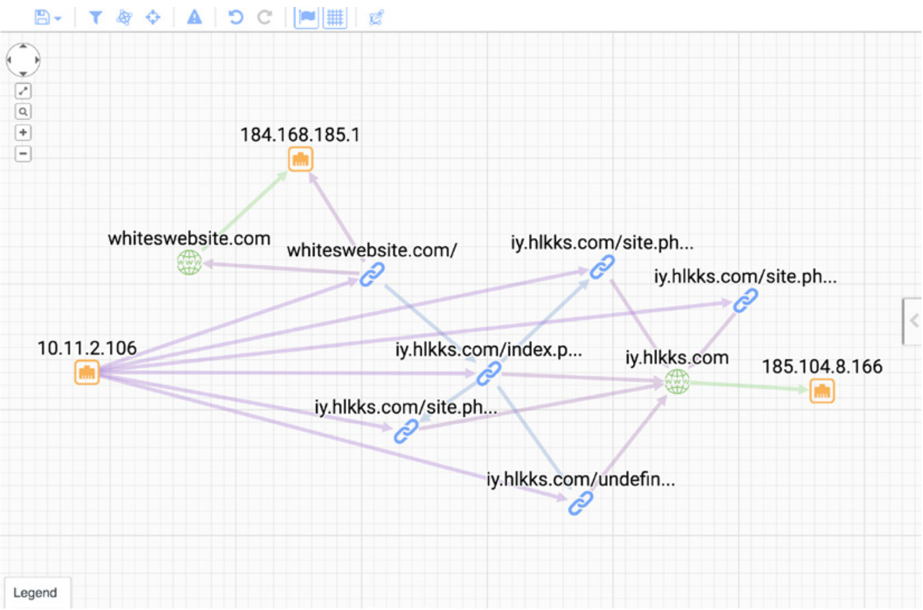
To identify use of common protocols, as in this example, focus on application protocol metadata, including:

- Proxy logs, IIS logs
- DNS resolution logs
- Bro HTTP, SSL, DNS, SMTP logs

Techniques to Use

Indicator Search

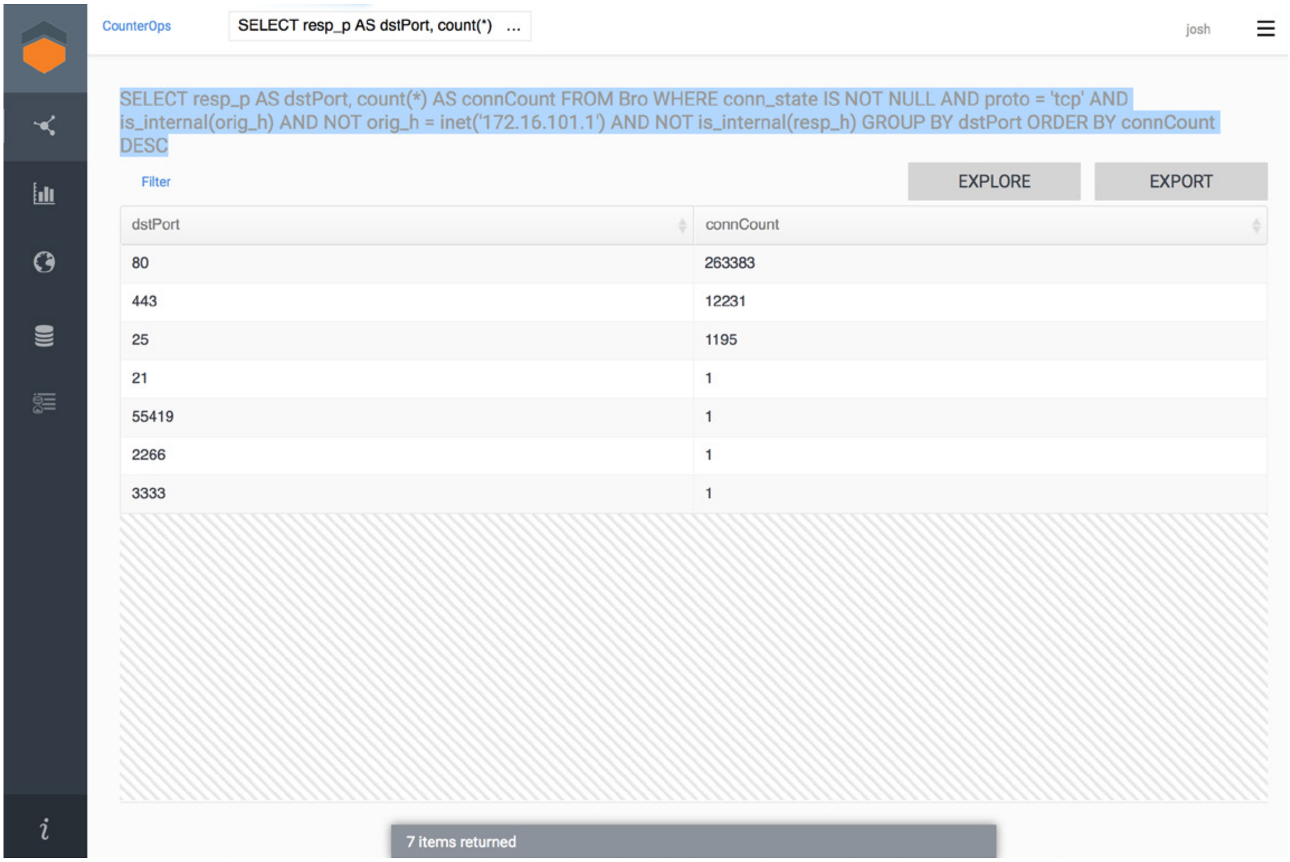
The value of this approach will be impacted by the value of the indicator. Locally sourced indicators will generally provide a high value because they tend to be timely and relevant to the network or systems you might be trying to protect. These can be gathered from previous incidents or by internal threat intelligence teams.



Common network session indicators: IP address, Port
Common application protocol indicators: Domain (HTTP, DNS, SSL), URL (HTTP), User-Agent String (HTTP), X.509 Certificate Subject (SSL), X.509 Certificate Issuer (SSL), Email address (SMTP)

Stacking

Stacking is a technique commonly used in many different kinds of hunts. In the case of hunting for command and control activity, a hunter will want to stack for anomalous instances of inbound or outbound traffic. Metadata types that can be used for stacking include: Ports, URLs, X.509 Certs



```
SELECT resp_p AS dstPort, count(*) AS connCount FROM Bro WHERE conn_state IS NOT NULL AND proto = 'tcp' AND
is_internal(orig_h) AND NOT orig_h = inet('172.16.101.1') AND NOT is_internal(resp_h) GROUP BY dstPort ORDER BY connCount
DESC
```

dstPort	connCount
80	263383
443	12231
25	1195
21	1
55419	1
2266	1
3333	1

Machine Learning - Binary Classification

This involves using machine learning to isolate malicious C2 activity. Supervised machine learning uses labeled training data to make predictions about unlabeled data. Given a set of known good and known bad examples, you can create a binary classifier capable of taking in new transactions and deciding if they look more similar to the good training set or the bad training set. After the classifier is trained, you can feed your HTTP (or other network logs) through it and get back much smaller set of records that require analyst attention.

Example Hunt

Hunting for Command & Control

1. What are you looking for? (Hypothesis)

Hypothesis: *Attackers may be operating on a C2 channel that uses* **custom encryption (uncommon protocol)** *on a* **common network port**
Look for:

- Anomalies in monitored network port channels, i.e. connections that do not have protocol artifacts related to the common port you are looking at. For example, look for connections that have no identifiable HTTP metadata over port 80/TCP

2. Investigation (Data)

Determine what datasets you are using:
For identifying use of common protocols, you will want to focus primarily on application protocol metadata, including:

- Proxy logs, IIS logs
- DNS resolution logs
- Bro HTTP, SSL, DNS, SMTP logs

3. Uncover Patterns and IOCs (Techniques)

1. Use a search to identify legitimate protocol connections on a common port you will be inspecting, by looking at protocol metadata
   - If looking at port 80, search for any HTTP protocol records that exist for a given time period
2. Use a second search to identify all network session metadata (e.g., Netflow, Firewall, etc.) on the common port for the same time period used in step 1
3. Using the output of steps 1 and 2, remove the legitimate protocol connections from the session data. This should leave uncommon protocol connections on the common port
4. Take the results of step 3 and stack the data for what is useful to investigating your hypothesis
   - For example: destination IP, bytes transferred, connection duration/length, etc.

3. Inform and Enrich Analytics (Takeaways)

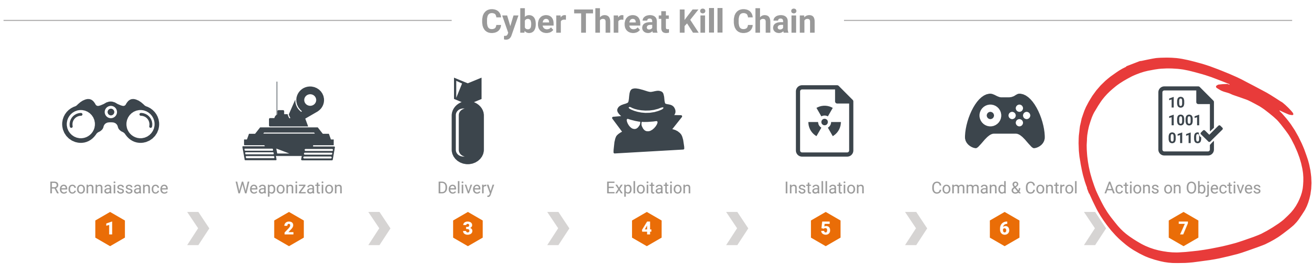
The destination IP addresses involved in the C2 activity you have discovered can be taken as IOCs and added to an indicator database in order to expand automated detection systems.

You can also create packet-level signatures to trigger alerts for cases where the custom protocol you have discovered may appear again.

CHAPTER 9

Walkthrough: Hunting for Internal Reconnaissance

Internal reconnaissance belongs to the 7th and final step of the kill chain: Act on Objectives. Internal reconnaissance is the process of collecting internal information about a target network, so that an attacker can more effectively move through the network and conduct further activities.



Process enumeration

After gaining access to a host or network, an attacker will use this process to attempt to establish what processes are running on the local host and the surrounding hosts. The commands used by attackers for process enumeration will depend on whether the attacker is looking for specific services (i.e. critical processes that run at startup and in the background) or general processes. Specifically on Windows, commands that an attacker might use for identifying services, running services, and scheduled processes include but are not limited to:

Identifying Scheduled Processes	Identifying Running Processes	Identifying Scheduled Processes
net start	tasklist	at
sc query	Get-Process (Powershell)	schtasks /query
gsv (PowerShell)	gps (PowerShell)	Get-ScheduledTask (PowerShell)
Get-Service (Powershell)	process (WMIC)	Get-ScheduledJob (Powershell)
service (WMIC)		job (WMIC)

Datasets to explore

For internal reconnaissance, there are two major data types that are useful in a hunt: process execution metadata and network connection metadata. In this context, critical metadata associated with process execution includes command-line commands/arguments and process filenames. This metadata should include the name of the host that the process was executed on and the name of the user who executed the process. Critical metadata associated with network connections include the source IP address, destination IP address, destination port, start time of the connection, and end time/duration of connections. For hunting network enumeration with this type of metadata, it's best to have data that includes internal-to-internal connections between hosts on a local subnet.

Process Execution Data Tools

- Sysmon
- PowerShell auditing
- Process creation auditing

Network Connection Data Tools

- Bro
- Netflow

Techniques to Use

Searching

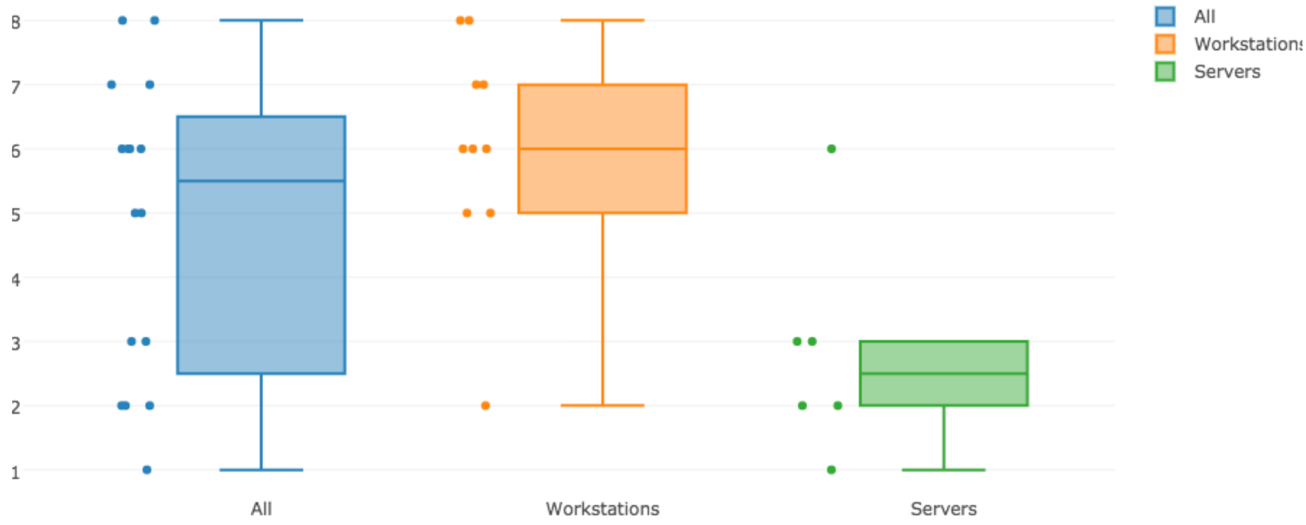
Searching for internal reconnaissance commands and patterns can be useful if the search includes thoughtful filtering, especially based on friendly intelligence. To make this technique effective at finding internal reconnaissance, it's best to have an explicit goal in mind such as searching for command execution of 'whoami' on across a particular class of workstations that should not normally execute the command (e.g., C-suite laptops).

Grouping

Grouping for internal reconnaissance commands is similar to searching, except you can review multiple artifacts across multiple assets in one result set. It's valuable to take commands related to a specific architecture (e.g., Windows), put them into a single group, and look for the execution of the group on a single asset. This technique works best when you are hunting for multiple, related instances of unique artifacts.

Visualization

Multiple visualizations may be applied to hunting for internal reconnaissance, but for this example, we will focus on one: box plots. Box plots visually describe distribution of data, with a box that represents median values and whiskers that represent high and low values (outliers). It may be useful to visualize the frequency and variety of command execution across hosts, as well as command execution across hosts across time.



Above is a box plot of the number of recon commands executed by workstations and servers. There are 17 hosts in the dataset and each host may have executed up to 8 unique commands related to internal reconnaissance. The three potential points of interest are the two workstation upper outliers and the one server upper outlier.

Example Hunt

Hunting for Internal Reconnaissance

1. What are you looking for? (Hypothesis)

Hypothesis: An attacker conducting internal reconnaissance would attempt to carry out host enumeration and automate these commands with a script

Look for these commands to be spawned by a script:

- whoami
- net user
- useraccount (WMIC)
- Get-NetIPConfiguration (PowerShell)
- hostname
- ipconfig
- nicconfig (WMIC)

2. Investigation (Tools and Data)

Determine what datasets you are using:

- Process execution metadata
- Process filenames
- Process file hashes

3. **Uncover Patterns and IOCs (Techniques)**

   Using grouping, search for the above artifacts in process execution metadata. Specify that the commands should need to be executed within a given time frame. Doing this, you discover a previously unidentified script that contains commands to enumerate host information and saves the results in a unique file.

4. **Inform and Enrich Analytics (Takeaways)**

   Taking the script and output files, you can now add those file names to your indicator database and automated detection tool's watchlists. In this way, if the attacker continues to try and use this script on another host it will be detected automatically. The indicators can also be used to identify other previously compromised systems.

https://www.linkedin.com/company/threathunting

CHAPTER 10

Parting Advice from Hunters

We've run through a lot of concepts and ways to hunt. But hunting is a practice like any other; you learn best by doing it, so don't hesitate to jump in. In the spirit of learning from experience, below is a collection of thoughts and tips from seasoned hunters to inspire and motivate you on your path to hunting down and stopping your adversaries.

If you would like to read an expanded collection of advice, tips, tricks, techniques and war stories from real hunters, you can check out the "Hunter Spotlight Series" at threathunting.org.

Samuel Alonso

KPMG, 2 years hunting

"I highly recommend understanding some basics such as the logs, devices and network you work with. Additional skills such as scripting and data analytics will also help, since the amount of data we have in the enterprise is otherwise unmanageable."

Stephen Hinck

ICEBRG, 5 years hunting

"Pick a protocol that you have a solid familiarity with, including what information exists within that protocol and should be logged. For example, in HTTP logs, we know there should be a domain, URI, user-agent, method, and other important fields. Start by stacking those fields to look for oddities, for example, only a small set of certain terms should appear in the HTTP method field – any deviation from that list may be cause for further investigation."

Danny Akacki

Fortune 100 company, 4 years hunting

"Learn your environment. This is so important I want create a cast iron mold of those words and smack you in the head with it. Use cases are great, but the thing about low hanging fruit is that it will always rebloom. Learn your environment so you can target your endeavors to catch the bigger fish."

Travis Barlow

GoSecure, 7 years hunting

"Question everything and assume nothing! One of the most useful things you can do as a hunter is to harvest as much data as possible and then examine it using multiple different lenses."

Alan Orlikoski

Square Inc, 3 years hunting

"Stay as organized as possible. The difference between a one time finding and a hunting program is organization."

Matt Arnao

Lockheed Martin, 3 years hunting

“Almost anyone can succeed in this field given a passion for the work, the right mindset, and foundational skills. Every successful hunter I have met shares these qualities.”

Josh Liburdi

Target, 5 years hunting

“Don’t get frustrated and don’t be discouraged. Threat hunting doesn’t always end with you finding the latest and greatest threat actor in your environment–– sometimes you find misconfigured servers, sometimes you find users doing odd things, and sometimes you find nothing.”

Chris Sanders

Applied Network Defense, 10 years hunting

“Success in hunting is really all about curiosity. If you see something that looks weird, find the motivation to run it to ground. The truth is that “gut feeling” is really not about your gut at all, it’s about your experience and the number of scenarios you’ve encountered.”

Jason Smith

Cisco 6 years hunting

“Don’t think that hunting is some sort of activity that only the NSM “elite” do. If you have accessible raw data, jump in it. Most importantly, keep a good notebook. Once you fill up that notebook, get another one. There is nothing more important than keeping track of your steps, and also keeping track of your methods. Also, if you discover a new method, don’t assume that everyone else already uses it. Share it!”

David J. Bianco

Target, 8 years hunting

“Don’t be afraid! Just get in there and get started. We are at the beginning of a Golden Age for threat hunting. We’re practically swimming in data, and a community of hunters is starting to come together to share their successes and failures. If don’t know where to start, or if you have a few hunts under your belt and want to expand your repertoire, check out ThreatHunting.net for practical tips from other hunters.”

Conclusion and Further Reading

This book gives you a solid foundation for beginning to practically delve into threat hunting, but there's always room to learn more. If you would like more information on hunting, some of the resources below are great places to start.

For a general repository of tried and true hunting procedures and techniques compiled by seasoned hunters, check out threathunting.net.

A full catalog of whitepapers, eCourses, hunter profiles and interviews, among other resources, can be found at threathunting.org.

- Generating Hypotheses for Successful Threat Hunting

Happy Hunting!