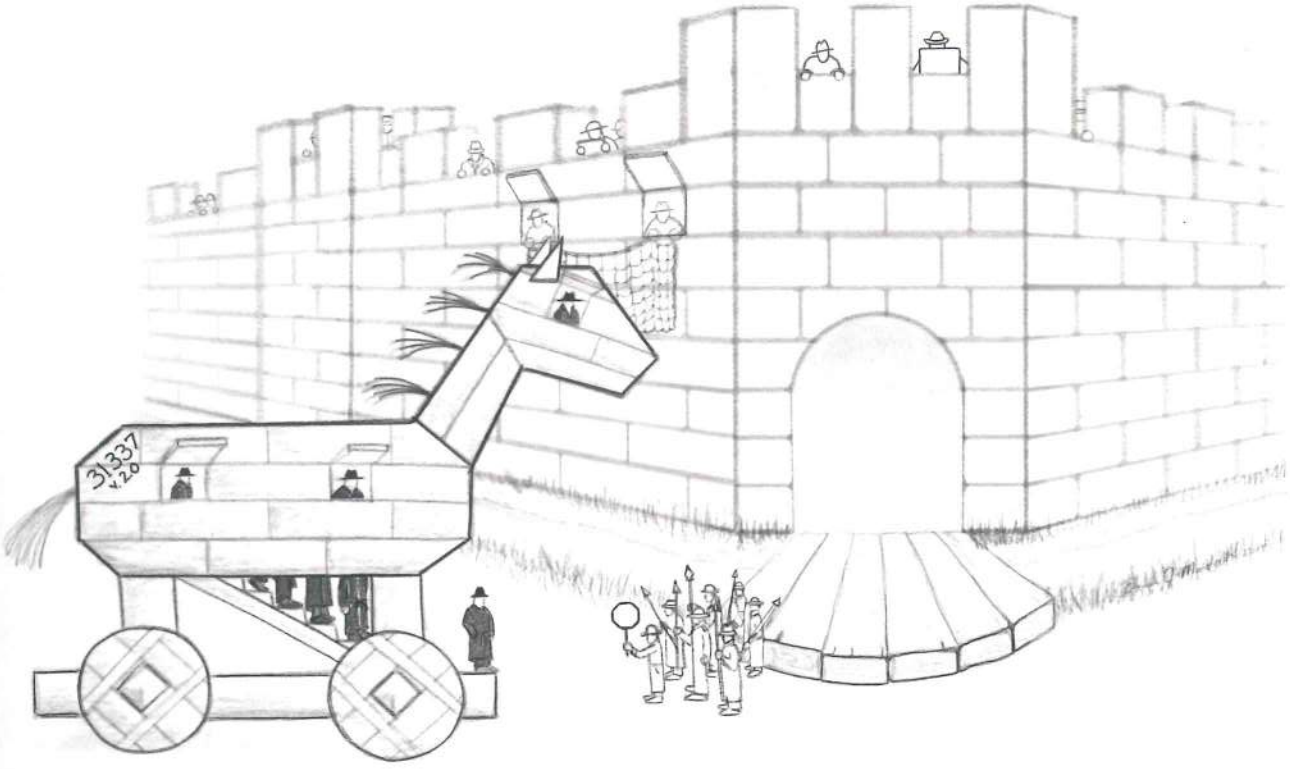
Blue Team Handbook:

SOC, SIEM, and Threat Hunting Use Cases

Notes from the Field (V1.02)

A condensed field guide for the Security Operations team.



Don Murdoch

Blue Team Handbook Vol 2: SOC, SIEM, and Threat Hunting Use Cases
Notes from the Field

A condensed field guide for the Security Operations team. (V1.02)

By Don Murdoch, GSE #99, MBA, MSISE
Illustrated by Bonnie Murdoch, BFA.

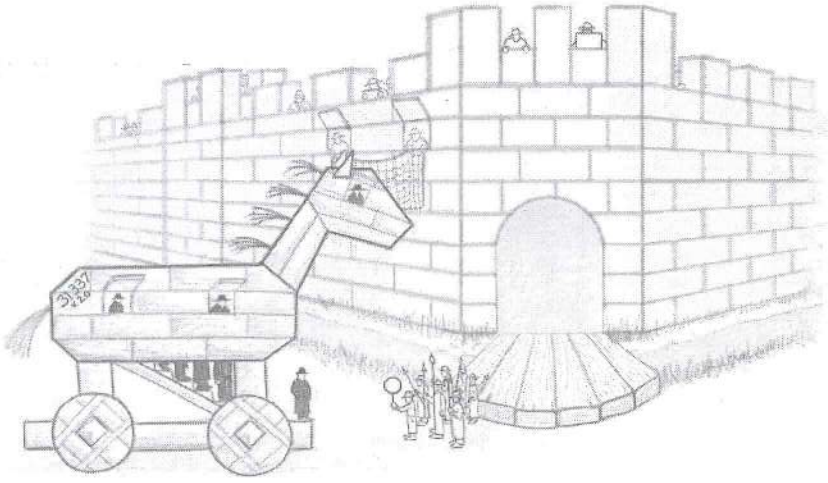


Table of Contents

This book is available at special quantity discounts to use as premiums and sales promotions or for use in academic and corporate training programs. Please contact the author through www.blueteamhandbook.com.

Version 1.00: Initial Printing September 2018.
Version 1.01: First Update: (v25) 11/10/18 - spelling, grammar updates.
Version 1.02: Second Update: (v27b) 3/23/19 - content updates.

ISBN-13: 978-1726273985 ISBN-10: 1726273989 V 1.0, 1.01
ISBN-13: 978-1091493896 V1.02

If you would like to get in contact with the author or illustrator, please use the contact form on the website at www.blueteamhandbook.com.

Art Notes: Art in the BTHb series is designed to be humorous and punny (yes, that IS a word!). We hope you enjoy it. If you would like to use the artist for your own books, presentations, or articles, please use the contact form on www.blueteamhandbook.com.

Table of Contents

List of Tables

Table of Figures

Preface

With the ever-advancing adversary, technology advancements, and a critical need for more skilled security operations practitioners, it is imperative for organizations to enhance their PDR cycle: Protection, Detection, and Response. This book attempts to answer that call by sharing experiences gained implementing five different SIEM technologies for more than a dozen organizations, running a MSSP division, and building several security operations centers.

Who this book is for: IT pros, cyber security pros, security operations staff, security consultants, SOC staff, SIEM designers and consultants, and line managers: those responsible for protecting information assets and teaching the next generation of security professionals.

About the Author: Don Murdoch, GSE #99, MBA, MSISE (GISF, GSEC, GCIH, GCIA, GPPA, GMON, GCFE, GCFA, GCPM, GPEN, GSNA, GPPA, GCWN, GCUX, TOGAF Enterprise Architect, SABSA Chartered Architect. CISSP, ISSAP), is a thirty-year veteran of information technology, with more than half a career devoted to information, computer, and network security. Don started his career with a boutique contracting firm located in eastern Virginia, writing COBOL and FoxPro code. For the next twelve years, Don took on role in a different aspect of IT as he grew his career: managing a small network, writing old school Perl/CGI software, developing international billing software for a startup ISP, and managing IT for an Application Service Provider right at the time of the Dot Com bubble. Don transitioned into information security where he started a DRP practice for an IT commercial spin off from a television production company. Things started cooking when Don entered his "digital combat training" phase in the "Wild, Wild, West" of academic computing for one of Virginia's largest institutions for higher learning. Don wrangled bots, tangled with well-equipped adversaries, and discovered what today are described as nation state grade attackers who were using the University network as a training ground. His University was the first in Virginia to implement a SIEM, user managed anti-spam technology, and an active countermeasures network based on Tom Listons' Labrea TarPit. After that experience, Don took on managing SIEM, conducting employee investigations, and security architecture for a Fortune 500 healthcare firm. That firm was acquired in 2012. His career took a security hiatus for a few years when he was an Enterprise Security Architect and then started running Infrastructure Strategy and Planning team for a Fortune 50 corporation. In 2016, an opportunity to develop an MSSP practice came up with a different boutique consulting firm. After two years, Don left to run the Cyber Range at Regent University, where he is today coaching the next generation of Cyber Defenders.

Don started working with the SANS Institute in 2002, taking courses, earning certifications, developing Stay Sharp courses during the mid-2000's, and currently teaches courses at the Community level. He earned the **GIAC Security Expert (GSE #99)** certification in 2014, as was later vetted as a Cyber Guardian: Blue Team in 2016 (#38).

About the Reviewers: Each of the technical content reviewers is a seasoned InfoSec pro with multiple certifications. The group represents a cross section of the community ranging from security operations and management, vendor product development and implementation, penetration testing, security engineering, and architecture. These reviewers are directly responsible for 42 pages of additional text from the original draft and collectively provided over 700 suggestions as a testament to their skills and passion for helping me to produce a well written book. I cannot thank them enough. In alphabetical order, the reviewers are:

- **Christopher Beiring:** Lead Security Operations analyst, network penetration tester, and all-around security engineer for a Virginia based consulting firm.
- **Chris Crowley,** Montance, LLC. Chris is a well-known information security consultant, a Principal Instructor with SANS, and a course author for two courses, including Management 517: Managing Security Operations.
- **John Hubbard:** John is a SANS Instructor and author, a SOC Lead for a large pharmaceutical company, and an all-around dedicated blue-teamer.
- **Seth Misenar, GSE #28:** Seth is a Cyber Security Expert who serves as a Faculty Fellow with SANS. Seth is a co-author for the bestselling SANS course SEC511: Continuous Monitoring and Security Operations. Seth provided the initial technical review for this book, when it was in its infancy.
- **Ryan O'Connor:** Ryan is an InfoSec security operation engineer for a leading security products company.
- **Phil Plantamura**: Phil is the COO for Security Onion Solutions. Phil has a 20+ year distinguished career in InfoSec working for defense, IR firms, and education.
- **Chris Sanders, GSE #64,** Applied Network Defense. Chris is a well-known security analyst, author, and educator. Chris reviewed the sections titled "A Day in the Life of SOC Analyst" and "Alarm Investigation Process."
- **Johanna Schafer, M.A.C.E.**: Johanna provided a layperson read through, checked it for readability, grammar, and punctuation for this book during the technical review process. My favorite error she found was the word "bacon" instead of "beacon", which for some strange reason, was a repeat occurrence in early drafts.
- **Peter Szczepankiewicz:** A long term colleague and SANS Instructor.

- **Martin Tremblay, GSE #80:** Martin has 20 years of combined red and blue team experience. He works for a leading international consulting firm and is based in Canada.

Update Notes

Major updates are indicated with (V1#) in the section heading.

Version 1.01: Corrected grammar, spelling,

Version 1.02: Expanded the project plan section beginning on 20, and in particular the EDIS discussion based on request from a state InfoSec team. Reviewed and added various Windows Event IDs*, corrected a few errors. Updated the list of suspicious EXE's.

Foreword

The choirs sang and the trumpets blared as a joyous parade marched down the avenue amid the blinding confetti thrown from the high-rise windows above. Sweet smells of cotton candy and funnel cakes permeated the air. The feeling of triumph flowed through us all...at least it flowed through a much younger me. I also may have been the only one who heard the marching bands and the angelic choirs. And, even though my excitement was palatable, my role in it all was merely tangential. But it was a turning point. Our SOC team's first (somewhat) successful SIEM deployment. From the ashes of web-based syslog, convoluted database exports to spreadsheets, and tools with names like ACID and BASE, arose the Colossus of Logs. From my little corner and my basic use case, I frequently paid homage to this wonder of the computing world, mostly through conducting searches that evolved over time, and crafting basic tools to help analysts.

Those crack analysts were applying some techniques covered in this book, but they didn't have a copy from which to work. At that time, our team spent countless hours thinking about our data, analyzing it, and determining ways to find evil. Our engineers built processes, tools, and dashboards so the SOC could work more effectively and empower the junior analysts. Over time, our SOC innovated, building more focused and custom dashboards for multiple use cases, ingesting intel and other content, writing scripts to pivot, and more, all to improve the analysis process. That team and toolset continually improved and never quite lived happily ever after, but I'll always remember how much my career outlook changed after that first experience with a good team and a decently implemented SIEM.

But, now that I've seen more SOCs and log management implementations than I can count, both good and bad, I have since realized that maybe the choirs I heard were a little flat and that the trumpets might have been blaring something other than fanfare. If our group only had a book like this at the time, we would have used the SIEM with greater success. The most effective SOCs with the most solid SIEM implementations got there through thoughtful strategy and skill, with seasoned pros who had spent years in the fight. They answered the right questions: Which information is important? Which logs produce that information? Who needs it? Why? Encapsulated in this book is a fifteen-year career building SOCs and implementing SIEM technology for finding evil every day. Many of the thoughts in here come through in my work today.

At Security Onion Solutions, we routinely consult with organizations of all sizes which use our well-known free and open source platform as a core component of their network and enterprise monitoring solution. During many of those

deployments and in classes, we are often asked, "What should we monitor? What makes a difference? How do I find the evil lurking in the network? Am I logging the right things?" Security Onion is one of many technologies available to help answer some (clearly not all) of those questions and, with hundreds of thousands of downloads and implementations across every industry, it keeps us pretty busy.

I met Don and many other thought leaders in our community at our annual Security Onion Conference in 2017, where Don delivered a very well-received talk on building security operations use cases. Don later asked me to be one of the technical content reviewers because of my experience as a technician, consultant, and leader. I was humbled and excited to participate. As I read through the draft, I found myself waxing nostalgic on my first SIEM deployment in the early- to mid-2000s. I later commented to Don that it felt like the words on the page were things I've been saying for a long time and were topics on many client engagements, but never written down in one place. In "BTHb: SOC, SIEM, and Threat Hunting Use Cases", Don covers how to use security focused data sources to their fullest, how to write a solid SOC focused Use Case, security metrics you can actually use, and how to build engagement plans and practices so you will be successful.

You might not hear choirs singing; but, if you're about to embark on the journey of building a SOC and/or SIEM, whether to implement in a green field, to validate your position, or simply to improve your security posture and capability, you have the right book in your hands.

Phil Plantamura, COO
Security Onion Solutions LLC
phil@securityonionsolutions.com

Introduction

This idea for this book actually predates the first book in the series, BTHb Incident Response Edition. In 2011, our team needed to replace our commercial SIEM platform. We headed down a path that lead to my fourth major SIEM implementation. We needed an outline to develop use cases, document all of the attributes of a use case and SOC procedures to fully use the new platform. I wanted our chosen vendor to have the best possible chance of bidding on the work and completing our use cases on time, and on budget. After vendor selection, we engaged a major firm, and set about replacing the legacy platform. The vendor estimated they could achieve 26 to 28 use cases. After 12 weeks, we exceeded project expectations. They implemented 35 of 37 fully defined use cases that totaled 497 pages in print once the paperwork was done. The vendor liked the use case template format that they adopted it, and they still use it today. We added fourteen new right click integrations. We even had a custom UI extension that pulled in a dozen account attributes for every user account listed in an alert when we opened an alarm. Lastly, we also went from 1.5 people monitoring the prior solution to four full time analysts.

As a result of all that work, the idea for BTHb:SOCTH was born, and I started collecting notes that eventually became the book you now hold. Along the way I started a MSSP practice with a good friend working with a consulting firm. We won 78% of our POC's in year one, and had a 100% renewal rate during year two, so several of those life lessons are incorporated herein.

This book will cover many topics related to the Security Operations Team from a "Field Notes" perspective. It is based on a log career implementing multiple SIEM technologies, building SOC's, conducting all manner of cyber investigation, developing and running an MSSP. The major topics are:

1. Building a Security Operations *functional unit*, including provisioning plan, budget considerations, thought habits, analyst skills, and tiering structures.
2. Deciding how to structure your Security Operations capability and the services it will offer.
3. An extensive discussion of security focused use cases organized by their respective data source. This chapter describes what to monitor from a given data source, as succinctly as possible. Many of these use cases have a threat hunt theme to them.
4. Building Security Operations Use Cases using my own Use Case Template, followed by a complete use case to use as a model in your own work.
5. Critical SOC analyst skills and investigation processes, which my own team used while I managed a MSSP operational unit for two years.

6. A discussion on applying modern Threat Hunting to the Security Operations team.
7. And a host of other topics that relate to security operations, SOC analyst skills, and SIEM.

It is my sincere hope that this self-published book delivers on the Blue Team Handbook motto: "a zero-fluff reference guide for the security practitioner, written with the intention of sharing real life experience". I trust that you will learn something useful as you read it as many readers of BTHb:INRE have shared with me over the years.

Thank you for our support,

Don Murdoch, GSE #99, MBA, MSISE

Security Operation Center Field Notes

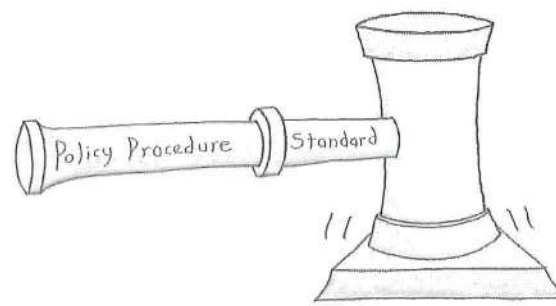
SOC Defined

A Security Operations Center (SOC) means different things to different people. Some say they "run the security platform", others say "they handle incidents", and still others say "they monitor the security of the network". The definition for a SOC used in BTHb:SOCTH is:

> "A centralized team in a single organization that monitors the information technology environment for vulnerabilities, unauthorized activity, acceptable use/policy/procedure violations, intrusions into and out of the network, and provides direct support of the cyber incident response process."

In a nutshell, the SOC is the *first line of defense*. This definition incorporates several important strategies for a successful SOC. First, a SOC must be under a single management and reporting structure so that it has a clear line of authority, funding, reporting, and accountability. Second, a SOC must have awareness of all aspects of both the business and the IT environment, from the smallest workstation to the largest supercluster in the cloud. Third, a SOC need to understand its area of operation (AO), how they will support the business, monitor business applications, and infrastructure. These criteria must be covered in the SOC charter. Fourth, SOC budget needs to be large enough to continually invest in people and support cross training instead of super sophisticated software. That concept leads to the fifth strategy: train and encourage analysts to be calm, correctly interpret alerts and their supporting data. This requires that SOC analysts are well trained.

One point deserves some elaboration. There are few different ways that the SOC team can establish its AO. The SOC can use the IT General Controls program, corporate policy/procedure, guidance from standards like the ISO 2700X series, or follow the Center for Internet Security's 20 critical controls. When designing, building, staffing, and operating a SOC, you need to develop a charter and mission statement.



In order to achieve these various strategies, the SOC needs to know the network, the application to server relationships, what is happening on the network,

and be able to determine if that activity presents a significant enough risk to the organization that the activity needs to be effectively dealt with. SOC teams don't solve security issues with complex SIEM software. They solve it with knowledge, skill, and ability. Complex SIEM tools help – but they are not a technological panacea.

SOC Charter

Every security operations center needs a "charter". The SOC charter defines how the SOC serves the business, mandate(s), and define governance and operational rules, what its areas of operation are, and how the organization needs to respond to the alarm conditions and monitoring the SOC performs.

Note that the SOC charter is not the same as a project charter. The SOC Project Implementation Charter is the formal document that authorizes the project to develop a SOC, possibly implement a SIEM, and empowers the project manager to apply resources and create the SOC.

The SOC charter is often developed in tandem with the SOC/SIEM project implementation charter. The SOC charter should be scoped properly, whereas an implementation charter is a Project Management Institute (PMI) defined document. The term comes from the Project Management Body of Knowledge (PMBOK) as a type "project artifact". Don't get the two confused.

Business Value Chain Tie In

One concept that IT people don't often embrace is the business "value chain". The value chain is the set of activities that take inputs and convert them into an output that brings a valuable service or product to their market. Value chains consist of: resource generators, inbound logistics, manufacturing or service operations, marketing, outbound logistics or service delivery, and after "sale" service and support operations. Today, there are very few aspects of the value chain that aren't dependent on some form of information technology, which must be monitored and fully secured following an IT General Controls Program.

In order for the SOC, and IT in general, to be relevant to and communicate with the business, they must understand how the business speaks, and the businesses' context and concept of operations[1]. Formally, a value chain should create some form of competitive advantage in the marketplace.

[1] These concepts are well defined in the Sherwood Applied Business Security Architecture (SABSA.)

Identify SOC Services

A Security Operations center can provide numerous services to the business and to IT. As you consider each of these services, be sure to incorporate them into your SOC planning process as well as the supporting skill, data sources, response patterns, and staffing to realize that service over the lifetime of the SOC. Further, as your organization considers the services it will offer the business, be careful to build out services which will be successful by only taking on a service that you can successfully deliver. The core services of a SOC *operations* team are listed out below. Your organization will certainly implement these services based on your own capabilities, funding, and staffing level.

Reactive Services	**Proactive Services**
Monitor Security Posture (Alerts)	Network Security Monitoring
Command Function (IR/Analysis)	Threat Hunting
Initiate & Manage Incident Response	Platform Health Monitoring & Support
Vulnerability Management	Cyber Threat Intel
Forensics/eDiscovery	Threat Intel Integration
Reporting	
Malware Analysis	**Other Services**
Intrusion Detection	Policy Procedure Support
Audit/Assessment	Internal Training and Support
Notification Refinement	

Monitor Security Posture: This is the primary role of the SOC: monitoring the environment for security conditions, alarms, health of the security platform, and responding through the organizations various technical solution(s).

Command Function: This may be a recurring activity, as the SOC coordinates alarm response, incident response, and forensic processes. Incident command can be a very intensive process. Incident command means that your SOC will identify incidents, work with handlers, coordinate containment operations, will assist in eradication efforts, take information from the incident and use it to better implement internal systems based on newly found intelligence, and may also support pushing out updates or other fixes.

Initiate & Manage Incident Response (identification and remediation support): A *significant portion* of the activities and instrumentation of a SOC focuses on finding and validating security incidents based on alarm and NSM work. The SOC may be empowered to initiate specific IR support from vendors, contractors, secondary business units – a wide variety of staff outside of the SOC and IR function. In these cases, an *operational process* needs to be defined with a set of *releasable data* provided to those outside of the SOC/IR team. Don't

freelance or make up these points on the fly – plan ahead. To start planning, review the application inventory and determine if IR support can be handled internally or if a third party needs to be engaged. Once you have planned, exercise your plan at least twice a year using a tabletop exercise format. Once that is stabilized, integrate various real data or activity components into testing the IR plan. Then graduate to engaging an external pen test team, outline an engagement structure, and put the blue team to the test.

Vulnerability Management: The SOC manager may be asked to assist, or even run, a vulnerability management program. The SOC manager should be very cautious not to take on tasking the SOC may not be able to handle: developing and deploying a round trip, full scope VA/VM program. Working through the process of safely finding, notifying, tracking, and attempting to identify the system owner and custodian, and *then gain system custodian and data owner support on remediating vulnerabilities in a timely manner can be a labor-intensive process.* Further, an effective VA/VM program needs to be executed within the business context and concept layer, meaning that the focus of the program should be oriented following a business criticality model. These are all complexities of running a program that can really stretch a SOC.

Forensics/eDiscovery: Depending on the size of the SOC, forensic support may be conducted in-house, or the SOC may coordinate and support forensic examinations with a third party. eDiscovery within an organization often uses the same or similar tools, requires chain of custody during the collection of case specific information, and will also analyze the results of data collection. A key difference is that eDiscovery is focused on collecting search specific information from live, in use data and information repositories that is generated and used by people. Forensics goes deeper, examining system artifacts from the file system that show intent for users to interact with files and data, malicious software residing in memory, or data deleted from disk.

Reporting: Run reports to support compliance requirements and IT General Controls monitoring. Run reports to support alarms, incidents, and other reporting requirements. Respond to additional data requests.

Malware Analysis: If a SOC analyst can safely recover a malware sample, then they may be inclined to perform some lightweight malware analysis using services like VirusTotal, JoeSandbox, or ThreatExpert. That advice was useful in ten years ago, and is no longer considered best practice. In 2017, the better course of action is to run samples through a local malware analysis engine built on Cuckoo sandbox *to prevent informing the attacker, who is likely monitoring online services, that their malware was found*. These tools allow a user to upload a suspect binary and then advise if it is known bad and provide varying

levels of activity analysis such as registry changes, new services, file system changes, IP addresses in use, or domain names looked up. If the analysis reveals something suspicious, then the SOC analyst would take that operational intelligence and be able to better search security data. More complex reverse engineering beyond this cursory level is a very specialized skill and requires environment setup for this purpose.

Intrusion Detection: There are several detection systems can be deployed on the network or on a host. These detection systems (Snort, Suricata, Bro, PassiveDNS, etc.) all require care and feeding in order to make sure they are operating properly. Winning budget to implement a NIDS platform that doesn't maintain the ruleset isn't an optimal solution.

Notification refinement and improvement: For alarm conditions that are deemed valid, create notification with sufficient supporting information for the recipient(s).

Network Security Monitoring: NSM is the collection, detection, analysis, and escalation of indications and warnings based on network level data that indicate an intrusion.

Threat Hunting: Threat hunting is a proactive process that inherently assumes that there is some form of intrusion or breach. Threat Hunting begins with generating a hypothesis of a compromise and then tests that hypothesis. It includes the systematic review of flows, account activity, and event review both from a longitudinal perspective and in the aggregate. Threat hunting sees to detect security threats, intrusions, misuse, and breaches by data mining.

Platform Health Monitoring: Monitor SIEM dashboards and alert stream, reviewing and acting on alerts following a priority basis. Monitor SIEM platform and other supporting data sources in order to detect issues and work with data custodians to ensure data survivability. Update platform definitions (assets, networks, privileged users, alarms, etc.) as the environment changes. Includes maintaining source data availability and quality by checking to make sure that events are parsed and creating new or refined alarms.

Cyber Threat Intelligence: This is the analysis of adversaries, their capabilities, motivations, and goals. Cyber threat intelligence (CTI) is the analysis of how adversaries use the cyber domain to accomplish their goals. When considering CTI, you should use multiple sources. Not all CTI sources are the same or offer the same degree of coverage. Also, CTI (in my opinion), includes understanding software vulnerabilities and ready-made attacker capabilities. For example, what are the new Metasploit exploits added this week? Metasploit makes the process of exploiting vulnerabilities significantly easier because exploits are

encapsulated into reusable code. How quickly does a new exploit appear after a vulnerability is announced in a technology you depend on? By keeping aware of attack tools, vendor announcements, and postings from major vendors from the IR community such as SANS, TrustWave, IANS, FireEye, CrowdStrike, AlienVault, and EMC/RSA, you can build a very low-cost CTI program and then make a purchase decision.

Threat Intelligence Integration: This is the process of carefully selecting and bringing in threat intelligence feeds into the system to improve alerting and better identify suspect or malicious sources, destinations, domains, and other patterns. Threat Intel sources and the information they provide should be on the detection roadmap.

Policy and Procedure Support: Many of the monitoring controls and capabilities should tie directly to established policy and procedure. As use cases are implemented, ensure that there is a tie-in to how the SOC will support PnP enforcement. More specifically, as this service area matures, ensure that SoP's are written to define how the SOC will properly engage with the user, supervisor, HR, and Legal in response to violations of PnP's.

Internal Training: Iron must sharpen iron, so the SOC management team must ensure that as the SOC changes the line staff must be trained and kept current. For example, as a new data source is integrated into the SOC, all members need a briefing on the data source and how to use it properly.

SOC Project Planning Outline and Field Notes (V1.02)

"If you fail to plan, you plan to fail. "
– Commonly attributed to Benjamin Franklyn

Instead of repeating any of that content in BTHB:SOCTH, a condensed outline for planning a SOC based on the PMI PMBOK[2]. Also, *do not shy away* from using the PMI PMBOK because "project managers are annoying", "project

[2] This section was significantly updated for BTHb:SOCTH V 1.01

management is useless", or "it's just not that hard". A solid PM that understands how to drive a project to completion on time and within budget is a *tremendous ally for anyone building a SOC or implementing a SIEM.* This section provides a no frills, just facts, discussion on SOC and SIEM planning. As you read through this section, many of the statements will become elements on a project plan as a "plan the item, conduct the item" line item entry.

Develop key business focused understanding of the organization and how the SOC can support its goals and objectives.

1. Understand the organizational need for a SOC, which means that you need to ***understand your organization's goals and objectives***. By being able to articulate how the SOC protects what the organization produces, sells, or the services provided to others, the SOC will have more credibility, be relevant to the business, and support your organization's mission statement.
2. Understand the business problem(s) the SOC needs to address and value chain resources that the SOC needs to monitor. You may need more of a "compliance" focused SOC, a tactical SOC, an Incident focused SOC, or some combination of these. The SOC will monitor several components of the value chain in addition to general IT resources. The SOC that intelligently targets the value chain for monitoring will be more successful and relevant to the business.
3. Identify the SOC sponsor. The sponsor may have an uphill struggle to initiate, build, and deploy the SOC. The SOC manager must be sure that the sponsor relationship is well maintained. The "customer" should want SOC services, and not have them dumped in their lap. The other operational roles will need to be well staffed. Evaluators and regulators are examples of "external stakeholders". These roles will be staffed by auditors with varying skill levels who are attempting to measure and report on risk and the degree of compliance within the organization. Understanding the questions stakeholders are likely to ask will inform use cases, reporting, and data sources that should be implemented to report to the SIEM platform.
4. Ensure there is an actual need for a SOC and its supporting logging infrastructure. Be ready to articulate that need, and explain how the staff and technical capabilities meet the need. Here, you should develop a formal business case. Be prepared to justify the staff, resources, access, and software needed to build a SOC.
5. Develop key "Security State" understanding (the "as is" versus the "to be" state). This understanding is technical in nature and corresponds to various use cases and monitoring needs from the traditional IT perspective. Wherever possible, connect a security state monitoring capability with a value chain component and the IT General Controls program. Refer to the

most applicable standard for your industry, such as the ISO 27002. See page 245 for more information.)

Build your initial business case, charter, project plan, budget request, and justification to support building the SOC.

This process will likely be two to eight months' worth of effort. Design the phases, identify the *key inputs and outputs per phase per the PMBOK,* and who will support each project phase.

1. Define the organizational ownership, responsibility, and SOC location. Attempt to locate a physical space that will accommodate twice the head count you will have in year one, so that you don't have to move in year three.
2. Identify the key roles for SOC: "architect", "engineer", "analyst", "manager", "customer[3]", "sponsor[4]", and "stakeholder[5]". Several of these are nearly identical to the roles defined by PMI's PMBOK (definitions in footnotes, more information on page 51).
3. Identify the relevant Policy, Procedure, and Governance - in place, or new PnP's that need to be written and adopted. Review existing PPG and determine if they support the SOC. Ensure that the SOC function is integrated into IT processes, particularly new application acquisition, server provisioning, and change management process. Also, the SOC will need to consume the forward schedule of changes, maintenance window updates, and notifications that changes were successful[6]. As a monitoring service, the SOC team needs to know about changes so that they don't over react during change failures or other OS and app changes that may seem suspicious.
4. Document necessary staffing levels, training, and educational process(es) (more information on page 45). Here, concretely plan for the first year. Once that's done, develop a three-year plan and assume that you will have above average turnover. SOC Analysts are in high demand, and incident response tends to burn people out. Note that a SOC of one person *isn't a SOC.* It is often a highly motivated person who will perform heroic acts and will eventually burn out, or a single person running a SIEM.

---

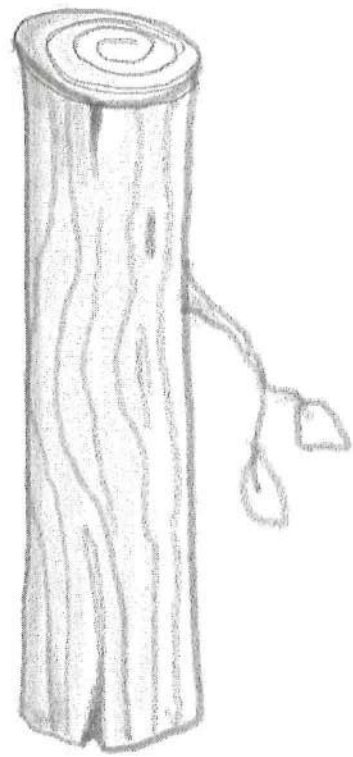
[3] Customers and users. Customers are the persons or organizations who will approve and manage the project's product, service, or result (from PMBOK V5).

[4] Sponsor. A sponsor is the person or group who provides resources and support for the project and is accountable for enabling success (from the PMBOK V5).

[5] Stakeholder: an individual, group, or organization who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a project (PMBOK V5).

[6] These are ITIL V3 Change Management terms.

5. Conduct a current data source survey. Identify the data sources, their logging configuration, assets, applications, application to asset mapping, data or logging suitability for the SOC. *You should not assume that every candidate data source is well instrumented and has the level of auditing your SOC will need.* As you prepare your data source survey, *preserve vendor and product documentation* that describes how the logs work and what their values mean. You will need this detail later. During this process, you will need to inventory how each data source can provide information to a future SIEM: syslog (UDP or TCP), file write, database table, SNMP traps, etc.



Conduct an Environmental Data Inventory Survey (EDIS). (V1.02)

Not only do you need source system data, you need metadata about the network, organization, users, applications, and a mapping of the business processes that depend on the organizations applications.EDIS[7] begins with developing an inventory of major business processes along with the business process owner. From there, define the applications that enable said processes, and then the servers that support the applications - similar to BIA, BCP, and DRP planning. The difference between SOC/SIEM focused EDIS is the depth of information. BIA, BCP, and DRP are focused on bringing an application, data, and servers back into service, whereas SOC/SIEM is focused on enabling monitoring, understanding who to contact for an incident, establishing baselines, and being able rapidly investigate an incident Both processes collect similar data sets, and can complement one another. Data includes users and their demographics, network maps, address ranges, applications in use, app to server mappings, app to RDBMS (or other data storage), input/output streams, web services that the application uses, and the overall organization chart. Many of these data sources will provide information to the SOC and the SIEM through automation, so ensure to get at least "read

[7] The steps are nearly the same done in the Business Impact Analysis (BIA) phase of a traditional Business Continuity Plan (BCP), and then the Disaster Recovery Plan (DRP). If your organization as a BCP, DRP, or TOGAF[7] style EA team, then consult with them for the application and server inventory

only" credentials for the systems that house this information such as a Configuration Management Data Base (CMDB).

From a Project Management perspective, the major steps for the EDIS process are outlined below.

1. Identify and develop an inventory of major business processes and departments. Note that this information may be readily available from a BCP and DRP plan.
2. Review the asset and network attributes necessary to best populate the target SIEM in order to maximize the data collection process.
3. Identify the applications which support business process, along with the data owner and system custodians, and from there document an application to server model, and thus the inventory of technologies in use. In many SIEM platforms this relationship will be implemented in an asset model, which supports more accurate alarm rule development.
4. Develop an inventory of every security focused or IT support technology. A sample list is shown below
    a. Network devices: Firewalls, IDS, VPN, DNS, DHCP, NAC, WiFi, WIDS, switch logs
    b. Technology Support systems: Mobile device tracking, Anti-Virus, Enterprise Detection and Response (alerts), Vulnerability Scanner, Password management system, web proxy, Email, virtualization platform, database systems
    c. Windows focused event logs: Application, Security, System, Sysmon Operational log. Note that as a subproject, the SOC implementation may need to spin up a separate project to implement WEC/WEF.
    d. Application logs: these usually require a database query or some other method of data collection
    e. Other relevant tools: Email security tool, Insider threat tool, System Backup logs
5. With the list of applications and security technologies, a line item for each item can be created in the project plan.
6. Estimate the number of hours to incorporate the data source for the use cases - these items will expanded in a subsequent phase, following the "progressive elaboration" model.
7. *Include a project specific line item* to develop a briefing for the SOC team that explains each data sources field set and field values.

For each of the identified data sources, you will need these planning and implementation elements:

1. Determine how the data source will be collected. Consult SIEM Data Collection Methods and Considerations on page 207 for more information on SIEM data collection methods.
2. Review the current auditing and logging configuration for fit.
3. Estimate to the extent possible the volume of data. For this point, try to get an average daily volume over at least a five-week period, which should catch any surges that naturally occur across a month boundary.
4. Determine if data can be trimmed, meaning review the data to find out if there are low to no value records provided by the data source that can be pruned or dropped either at the collection point or the arrival point.
5. Inventory the data fields from the data source, and develop an internal SOC training program so that all SOC staff understand the data source.
6. As needed, implement the necessary change control to configure the data source to report to the SIEM.

Plan the Technology provisioning process to support the SIEM, and another identified SOC services (see p. 17). Plan for twice the data you think you will need in year one.

1. Hardware: Including disk and disk controller architecture, as influenced by logging requirements and SIEM platform.
2. Virtualization Layer: Modern virtualization technology makes virtualizing your SIEM a very attractive option. When considering this option, it is critical to articulate the data speeds in terms of IOPS necessary for databases and/or data storage – don't assume that this will be handled by your infrastructure team.
3. Log storage architecture, scripting, and long-term storage requirements. For long term storage, you really need space over speed, because you rarely go back to data past a 90-day threshold. Reliable, safe, and large long-term storage is more important that blazing speeds. Blazing speed is needed for the past three days' worth of logs.
4. SIEM and supporting software. Note that most major vendors have their own predefined project plans for implementing their software, which you should leverage to the fullest.
5. Spend time on the Budget Process. A SIEM is actually a major enterprise wide application, and it deserves the same budgetary rigor as with any enterprise project. This means build a first-year model to get started, 3-year projection, and then a 5-year projection model. A significant component of the budget development process is developing the Total Cost of Ownership model. You will need to know your organization's technology refresh mode to plan for system replacement. You should assume 50% log storage growth year over year.

6. Application and IT resource data provisioning and possible development. This phase is where you will design how each application and data source will be integrated into the SOC and SIEM. It usually involves some significant custom development efforts. Each data source brings its own capabilities and will need some form of alert support.
7. In order to make this process work well, *find the gaps* in the security posture of your organization and work to *quantify* risks. To get this done, find your risk management subject matter expert (SME) or Point of Contact (POC) and partner up with them.

Build your log architecture, source data collection delivery, and SIEM and logging deployment plan.

There is more information in the SIEM Deployment Checklist section beginning on page 198. Also, Briefly:

1. Perform software and vendor selection based on a scoring model built from use cases that correlate to your business model, unique data sources, compliance requirements, and InfoSec program.
2. Review the auditing stance and build out the Events Per Second (EPS) rating for each of the systems in the environment that will provide data. Then plan for a 50% increase so that your solution can weather an "event storm". Its critical to determine the EPS *after* the source system has had its auditing level configured!
3. Provision the hardware and storage platform and implement.
4. Monitor your data feeds, reporting, and system response time.
5. Build your data integration plan for commodity sources, and carefully select customized sources. For example, an ERP application is not likely to be supported, so you will likely need to develop a database query to pull data from the audit table, implement auditing, test, develop a method to archive current data to a historical table, and monitor to ensure that the query process has minimal system impact.

Build out Use Cases.

1. There is an entire chapter in this book on building out use cases. Review all of that material and compare it against the use cases in your chosen platform.
2. Plan how to implement the vendor-defined use cases as these should provide baseline coverage.
3. Forecast the effort required and data sources to implement your own custom use case.
4. From there, prioritize the implementation so that you will have project measurement that supports defining earned value.

Build your response processes.

Response processes are enabled by the variety of data arriving, applications, business processes, and your requirements.

1. Response processes will be driven by your security program and the applicable standard you are following.
2. This part of the planning process should answer incident resolution questions like this: "When we get condition A from system B, what does the analyst do and what data is necessary for the system custodian to resolve the incident?"
3. In effect, the process of pulling data into a SIEM will provide the SOC function with dozens of scenarios that need to be worked through. As you build these processes, ensure that you are *outcome focused* – what objective needs to be achieved based on security condition X, Y, or Z presenting itself?

Build your SOC Metrics, as defined in Metrics for the SOC on page 39.

Many technical platforms have reporting and measurement that supports SOC and SIEM metrics. There are many organizational metrics that need to be developed and collected. This aspect of developing a SOC and SIEM implementation plan will evolve over time.

Build, and implement your continuous training program.

Training is a constant. SOC skills need to advance as the attacker's skill and determination advance. Ensure that there is budget for at least two tiers of education. Provide premium education for the more senior tier, and then develop OTJ training for the junior tier. OTJ should consist of knowledge transfer, short course, and job skills focused education. Investigate your local community college work force education program and capabilities in area.

There are several open source or very low-cost options. Consider ENISA, SecurityTube, SANS Cyber Aces, local BSides conferences, DerbyCon, and the annual Security Onion conference as more inexpensive education options.

Consider, and be Prepared, for Tough Questions

In order to fund SecOps, SIEM, and a SOC, you will undoubtedly face many questions. Here are a few of them that I have been asked over the years, condensed for publication. Determine the answer when building your funding request.

1. Nothing has happened yet. Why do we need to do this? How can you be sure that nothing has happened yet? As a possible answer, try these out: "It's not if, it's when."
2. How will the team detect and respond to security issues, incidents, and data breaches? How did we do this before? Isn't that what the sysadmins do?
3. Did the organization incur any costs from an incident last year? Virus outbreaks? What costs incurred from our peers and competitors?
4. How many users at what "level" were negatively affected (as in lost productivity) from an incident?
5. How are you going to measure yourselves and get on the IT Balanced Scorecard? As a possible approach, ask if you can be on the *business scorecard during the SOC charter development process.*
6. How will the team determine what alarm conditions are prioritized over something else – who wins? (Hint: asset value tied to critical business process and revenue stream protection).
7. I thought we spent X on Security last year. Why do you want more?
8. I know We don't have anything worth stealing. Why do we need to do more and more of this security "stuff"?
9. Don't those things cost millions?
10. We are doing vulnerability assessments. Isn't that enough? If you are not doing active and timely remediation, then no, it isn't.
11. Those security people keep saying no, so I'm going to say no to them this time. So there.
12. We can successfully outsource that for 1/8 the cost, right? After all, that's what the vendors say. Why do you disagree with them? Aren't they experts?
13. How will this SOC solve business problems for us?
14. What does this SOC thing look like year 1, year 3, and year 5?
15. I don't want to buy more expensive security people only have them quit. What are you going to do about staff retention? Burnout? Attrition? Internal transfer? I recently heard at a security conference when people take a SOC job they plan to quit in 18 months.
16. How will you know when you have had a success? What does success and failure look like for a SOC? (or a major security purchase?)
17. Have you been talking to the auditors again? They said something about this last year. I bought a new firewall.
18. Show me a playbook first – can you do that? Come back when that's done.
19. IT Is outsourced, it is "company X's" responsibility, not ours. We have no liability because that rests with the outsourced vendor and it's in the cloud/vendor contract[8].

[8] I would encourage not to blurt out in response to this question that that is "A Guaranteed Orange Suit Acceptance Posture". It does not go over well.

20. What can you do with a third of that? Because that's all we have.
21. We spent $3.5M on SOX last year. No more!

Collect the Bread and Butter Data Sources

There are many baseline systems that need to be monitored because they represent key data sources that you need in an incident and support compliance. This is part of the EDIS process. At an absolute minimum, these information systems and data sources to collect are:

1. DNS activity, with a focus on internal to external activity first (about 8% to 10% of your networks' DNS request/response traffic).
2. Windows Domain Controller security log.
3. Most, if not all, Windows member servers.
4. *Account life cycle, process execution,* and *presence* indicators from workstations. This item is best accomplished using Windows Event Forwarding and event subscriptions *because this is a native* built in capability in Windows, and prevents the need to deploy yet another agent.
5. Perimeter firewall. At a minimum, any outbound 'denies', accept and deny traffic to the DMZ, and platform changes. If you have capacity, collect outbound accept events as well, provided you cannot get a better data source for the communication flow. For example, if you have a proxy, you can consider not recording firewall data to/from the proxy if you can get the proxy logs. Proxy logs are superior to firewall logs as they are application aware and are user attributable whereas firewall data is not usually user attributable.
6. Database Account activity and account management.
7. For Linux, the minimum to collect are the sudo, auth, and authpriv logs.
8. Antivirus centralized console data.
9. Forward (outbound) proxy data. For the proxy, validate that the system records the user agent, referrer, the URI query string, and the allow/deny decision. If the proxy understands the site type, that is also useful.
10. Document editing "in the cloud", such as Google's GSuite or Office 365. This means who touched which file and how.
11. Shared Storage file system activity, as in who touched which file and how for user and process exposed shares.
12. VPN activity.
13. DHCP transactions.
14. Network device authentication which usually arrive through RADIUS or TACACS+. Further, network change detection, which usually comes from Syslog events.

Once you add your own "must have's" to this list, your next task is to get the daily data volume for each source. Volume has three factors: events per day, average event width, and the typical peak or spike times. From there, you can estimate the capacity you will need for your platform.

Useful MBA Concepts: SWOT and PESTL

There are two business management concepts that help when designing, planning, and building a SOC: SWOT and PESTL.

SWOT Analysis

SWOT is a strategic planning technique used to help an organization identify the Strengths, Weaknesses, Opportunities, and Threats that every manager should understand, and be prepared to use in strategic planning exercises. Building a SOC is an *internal business venture*, which is affected by both internal and external pressures. SWOT analysis will improve your business case for your SOC, will also help you plan, *and if done well can help identify adversaries that will launch attacks against the organization*. Below is a very brief example to give you an idea what a SWOT analysis for a SOC project could look like.

Table 1 An Example SWOT Analysis

	Traditional Business Mgmt.	Security Operations Example
S	Characteristics of the organization that provides advantage over others	Technical controls and monitoring capabilities; strong perimeter controls; Policy/Procedure in place and followed; valuable IP in place is protected (and targeted).
W	Characteristics at a disadvantage to others (competing projects)	Moderate Funding; staff and skills; volume and quality of log data; some log sources or logging capability missing from critical applications.
O	Items in the business environment that can be exploited (does not mean technical exploit!)	Improving security software; Implement awareness training; local University has Cyber program; Current VA program is semi structured.
T	Items that can cause trouble or thwart the project	Increasing stealthiness of malware; trusted insider can defeat controls (accidental or can be disgruntled); ownership; resistance to response from alerts; management expectation to "find the bad guy" is high.

PESTL Analysis

PESTL (**P**olitical, **E**conomic, **S**ocio-cultural, **T**echnological, and **L**egal) analysis is a framework of the macroeconomic and macro environmental factors pushing against the organization. It is used by strategic management, marketing, and business development teams who need a solid understanding of how the organization will perform given a particular business venture. PESTL analysis will help SOC planning along two dimensions: the change and pace of technology that the organization consumes or produces, and the legislative or regulatory environment where the organization operates *and has a presence* that requires monitoring.

Funding the SOC

Always remember that an organization has a specific mission or goal, and it articulates a set of objectives to achieve its goal in a mission statement. Since the SOC team is rarely a profit center, it should ensure that funding requests are aligned to the organization and the applications that enable the business.

Understand the App Stack. The SOC leadership team needs to understand how the organization is funded by its value chain, and in turn how much of the IT spend that SOC may receive in both Capital Expense (CapEx) and Operational Expense (OpEx). Practically, this means that your team needs an inventory of the business applications by their criticality, and then a map of the servers, database(s), storage, and network connections that enable and support these applications. With this model in mind, you can then work to align your monitoring controls, capabilities, and instrumentation to support the availability, integrity, and security of those applications. Your Disaster Recovery Planning / Business Continuity Planning (DRP/BCP) team can be highly valuable as they have a criticality-based view of the application stack. If you don't have a DRP/BCP team, the build the app inventory yourself with the intention that the work can be used for SecOps, as it will assist IT when there be a need for a recovery event.

For example, your organization is likely to have an eCommerce presence. There are several components on the eCommerce chain: digital storefront, order processing, messaging to the customer and supplier, the WAN link(s), back-end data storage, staff that manage all of these IT components, and protecting credit card transactions. In that list alone, there are dozens of servers, technologies, people, and processes. Therefore, if the security team can put

monitoring and incident response in place so it can detect violations in baselines and provide assurance that the components are working, it is supporting the company mission and the ability of the business to sell goods and services.

Finalize the Reasons to Fund SecOps: There are many, many reasons to fund security operations and the logging infrastructure that SOCs and SIEM platforms require.

1. Regulatory compliance (HIPAA/HITECH, Sarbanes Oxley, and others).
2. A prior incident may initiate a funding event.
3. Management Directive out of a genuine desire or a fear response.
4. Your chosen standard that defines how IT is structured, and are therefore audited against, may require SOC and a logging platform.
5. Logs within a given system are volatile. Some systems, like a Windows domain controller, only hold log data for a few hours and then the data rolls and is lost forever. Some systems hold data in memory, in a buffer, and when power is lost or the power is lost, so is the data.
6. Without Logs, you have no ability to go back in history and find issues – security, operational, or change related.
7. It is, in fact, "the right thing to do".

Security Operations Centers Cost Components

There are numerous cost components to consider when building a Security Operations Center. Below are many of the common cost components. For each of these costs, carefully analyze your current environment as you develop a "build vs. buy" analysis.

Direct Costs - There is more information on this below this list.

1. **Internal Staffing Level**. A 24/7/365 requires at least 5 whole people, using the barest bones staffing model. Target 9 staff people and one SOC manager.
2. **Vendor neutral Training**. Some examples are SANS, Security University, CompTiA CASP, and ISC2 SSCP.
3. **Product Training:** SIEM solutions have training provided by the vendor.
4. **Tools:** Desktop Infrastructure, OS, Office, SIEM license, and investigative tools.
5. **Subscriptions:** SOC's will have several subscription services, and in particular, threat intelligence services.
6. **Hardware:** Server, storage, and network Infrastructure. Be aware or the typical hardware refresh cycles – usually 4 years.
7. **Forensic Hardware:** Forensic hardware has unique requirements because these systems are usually isolated onto a small LAN in a locked room. For

example, a customized forensic workstation known as FRED can cost $5K and up, storing images on central storage can take many terabytes, and write blocker kits can easily cost $1K and up.

8. **Software licensing costs which includes annual maintenance:** Software includes SOC support, additional licenses for various management consoles, SIEM platform, ingest costs, forensic packages, PDF generation applications, BI[9] tools, and eDiscovery capabilities.
9. **Facilities:** SOC Room, furniture, shared large format monitors or projectors, and proximity card or possibly biometric door control. Also, the forensic analysis space should have its' own separate locks and proximity card control.
10. **Upgrades:** Annual upgrades – often handled on a SoW basis with the vendor.
11. **Vendor assistance:** Over time, you will need vendor assistance for new content, improved reporting, more training, and possibly upgrades.

Indirect Costs:

1. Recruiting costs such as a portion of building, recruiting, and staff pay increases.
2. SIEM Selection costs for the initial project, which includes labor expended to specific, review, and select the primary SOC toolset.
3. Developing on the job training for your SOC staff. Note, this will tie up key SME's and the time commitment cannot be underestimated.
4. Integrating new data sources, which may require customized data parsers, alerts, and reports to be created within your platform.
5. Periodic internal or external audit support.

Staff Cost Considerations: A Security Operations Center needs to be staffed by skilled Information Security Analysts. Period. Shiny SIEM solutions don't solve cases, educated and seasoned people do. ***After participating in InfoSec since 2001, I can confirm It's just that simple. Highly skilled people can produce more accurate and timely results with a moderate product set than novices with a super expensive shiny toolset.***

Base pay, benefits, and the inevitable staff turnover disrupts the cost model. The US Bureau of Labor Statistics lists Information Security Analyst base pay at $92,500 in 2016, and $95,510 in 2017. If your internal overhead load is 30% for administrative costs, a loaded position costs $124,163/year. At this rate that is $620,815 for five people per year in 2017 dollars. A 2016 study published on glassdoor.com can help to understand the hiring climate: Companies spend $4,000 to fill a position through open recruitment with a 52-day vacancy period,

[9] For example, Advizor Analytics and/or Tableau.

47% of candidates decline an initial offer. Further analysis found 50% of employees left a position due to their manager. These numbers help to define how much a temporary contractor would cost if you cannot find FTE's or need to replace an FTE. To minimize this cost, concentrate on getting SOC staff through the investment zone period and into the return zone period by onboarding them into handling specific SOC services and IR tasks as rapidly as possible.

The staffing cost is further influenced by the *coverage model*. If the team operates with 24/7/365, that requires 4.52 people, or 5 FTE's *at a bare minimum to staff the SOC with a single person staffing the facility. A lonely job indeed*. This value is based on 8,760 hours per year, two weeks paid time off and 8 holidays which yields 48.4 work weeks, or 1936 hours of coverage per person. In reality, any 24-hour operations team should plan on at least nine people to accommodate vacations, sick time, and staff turnover. Five people will cover the shifts, and the remaining three will provide additional coverage during high activity hours, such as 7AM to 7PM and some portion of Saturday. Missing from this estimate is the percentage of "admin" time. Admin time consists of all other company required tasking that detracts from conducting actual heads down job duties.

Incident response has a very *high burn out rate* compared to other technical professions. Therefore, to compensate rotate your SOC front line through different SOC services so that they have variance in job duties. Also, look for analysts that aspire to move up and not stay doing the same thing every day. SOC managers should always evaluate opportunities to vary the job duties in order to retain people.

Facility/Space: Most organizations have a per square foot rate for office space that should be in the cost structure. As an example, the average 2016 cost per square foot in Atlanta, Georgia was $20.01 for Class A space and $16.36 for Class B space[10]. If you assume 90 square feet for shared workgroup areas with two workspaces available for a five-person team on rotation, the monthly office space cost is $2,944.80 for a two-person SOC. There are other single purchase costs, however. For example, you may want two large format monitors and PC's to run them with everything mounted on the wall. That could easily be a $4,000 single event cost item.

[10] From Offices.Net, August 2017.

Vendor Neutral Formal Education: Assuming you can find security people, you will still likely need to train them in SOC operations. As of August 2018, one of the best courses from SANS Institute is "SEC511: Continuous Monitoring and Security Operations" with its corresponding GIAC GMON certification. This course covers the practical skills needed for every SOC staff member. I can attest that there is a measurable improvement to the quality and speed of each analyst who completed this course and the corresponding verification. The August 2018 cost weighs in at $6,939 USD. Also, ensure that the travel and hotel costs are included when looking at the cost of training, and estimate $1800[11]. Stay for Day Six and compete for the SEC511 coin[12]!

Product Training: Every major SIEM vendor has a series of product training course. These are usually included in the initial proposal and implementation. There will be a cost for new staff as they come onboard. To defray the cost, I've had success by asking for a training credit with an upgrade, system enhancement activity, or adding in new product component.

Organizational specific training: On the job training will be a continual process. There are numerous studies that quantify the cost to develop robust and reusable training. Data from The Association for Talent Development[13] listed the time to develop an hour of instructional delivery (a formal class) between 43 to 185 hours for stand-up professional instruction, with numerous factors affecting the time. Don't count this lightly, and don't ignore it. For concise insight into the learning organization and how valuable it is to the company, review Chapter Seven in Paid to Think by David Goldsmith.

Desktops: Analyst hardware, monitors (the more, the merrier!), monitor arms, client-side software, analyst licenses for dozens of applications, furniture, and lighting. Think Quad 24" or 27" displays, and an Ergotron type quad arm. Nice!

Vendor Support: Dedicated vendor support for the security product suite, use case implementation, and continual upgrade processes. These support relationships are usually priced on a per hour basis with a minimum number of hours per week. If your SIEM vendor charges $225/hour and sells this support arrangement with a minimum of 4 hours, the annual support cost is $46,800.

Infrastructure: Back end servers, sensor platforms, network instrumentation such as a TAP, and multi-tiered storage. Plan for a per server hardware refresh

---

[11] I have had good luck shaving a bit by staying in the hotel next door.

[12] Coin Image provided by the SEC 511 course authors and is used with permission.

[13] https://www.td.org/newsletters/learning-circuits/time-to-develop-one-hour-of-training-2009

at 3-4 months before your server maintenance period to ensure you are not paying a premium for keeping a server online outside of its maintenance window. If you need six servers at $8,000 each, that's an initial capital outlay of $48,000 just for the hardware. Plan for 20% annual maintenance, and technology refresh at the four-year mark. Most of the SIEM implementations I've done were virtualized using VMware 5 and 6. This model can be quite successful – but you will need to add in the cost for the Hypervisor. When it comes to actual capacity, spec 2 more CPU's and 4 more GB of memory that you think you need and virtualize your platform on just that server. The long-run benefits you will gain are tremendous. These include volume snapshots, copy over to the new platform during tech refresh, and the ability to more easily mix and match drive configuration.

Integrating new data sources: To minimize impact as new systems are brought online, incorporate the SOC engineering staff in the IT provisioning process. The objective is to ensure that new systems or major system updates can provide relevant log data to the SIEM/SOC team. *This labor charge should be assigned to the application or system, not SecOps, and is a recurring cost item for the life of the SIEM.*

Content Development: Developing new and refining current *use cases* within the SIEM solution. Current use cases will also be updated based on improvements from the threat hunting team. The better you define the input data, content needed, analysis, rules, notification, SOC actions, and outcome desired, the more accurate a cost you will have and the higher the opportunity for success.

SIEM Software: SIEM platform licenses are most often driven by a sizing factor. Typical factors are the "ingest" rate in events per second (EPS), GB per day, or monitored device counts. You will find there is a class of non-security relevant events that arrive at the SIEM along with the data that's really needed. Depending on numerous factors (event cost, processing horsepower, log storage, event width) you may want to develop a tiered logging method. Costs can vary widely here, from $20,000 per year and on up. The better you define your environment, the better an estimate you will get from a vendor.

SIEM Software Upgrade: Some upgrades for enterprise systems can be performed through an update process, and some cannot. Experience with five different platforms leads me to advise that a complex upgrade, such as a major upgrade, may be better off outsourced to the SIEM vendor. A typical SoW will be 40 to 80 hours at the vendor's rate plus travel and expense. Ensure that you investigate this fully with your vendors and integrate at least one annual upgrade event to your platform.

Audit Evidence Support: The SOC is often asked to support reporting on security event data and incidents in direct support of an internal or external audit. Staffing this specific role is closely related to the regulatory environment and how often auditors make requests. To estimate this, determine how many audits your organization responds to per year and the reporting needed to support those audits. The SOC team should always record their time to support audits, as this is a service to the business. If you have recurring audits tied to a particular unit, then ask for accounting charge codes to document costs for the SOC to support operating units.

In House vs. Outsourced vs. Hybrid SOC

Now that you have a structured outline of the costs and most of the long-term factors involved in building a Security Operations Center, you are in a better position to consider the pros and cons of outsourcing part or all of the SOC function. Some empirically based observations on engaging outsourced Managed Security Service Provider (MSSP's) are listed here:

1. Startup time will have an impact. MSSPs in effect deploy a partial to full SIEM solution on your network. Each data source needs to be integrated into the platform, hardware will need to be deployed, and your organization will still need to define your own incident response process.
2. An MSSP will only be able to go just so far when investigating alerts. If you are fortunate, the MSSP can cover 50% to 70% of the alarm conditions well and will engage your organization on 15% of the observed alerts.
3. MSSPs will *never know your network like you do, and you cannot easily quantify this impact to their quality of service delivery.* MSSPs also are unlikely to know what changed on the network, as they rarely participate in change control.
4. MSSPs work with you through a defined SLA and reporting relationship. They cannot replace your own staff who can reach out directly to a system custodian – this is an invaluable benefit to having in-house staff functioning in a security operations role.
5. Their opinion on alarm sensitivity and configuration *is not your opinion* because they tend to look at "genuine threat" conditions, and will ignore or tune out many other conditions. Your use of SOC should include policy issues, threat hunting, audit reporting, and gleaning operational value from the mountain of data the SIEM will consume.
6. There are some tasks that should be outsourced *if they are infrequent,* like system forensic analysis. However, the battlespace of today tells us that we need a memory image more than a disk image. This is nigh impossible for a third party outsource MSSP to collect but may within their ability to analyze.

7. You cannot delegate responsibility to a third party for the security of the assets under your organizations care, no matter how much someone tries to convince you that you can. You may be able to delegate authority to operate, but not the responsibility of system security.
8. 7/24/365 monitoring by a third party will cost you less for the labor component than building your own 6-8-person team. There is no getting around that fact. If you you are being pressured to outsource, realize this argument and devise ways to respond to the argument.
9. MSSP's may also be able to perform system upgrades and very likely have done more upgrades than you. Factor in the cost of your deployment a week or two to perform an annual upgrade.
10. Lastly, you get out of a MSSP relationship what you put into an MSSP relationship. If you do not invest time, then don't assume that they will give you stellar results.

Getting into the Hunt

Historically, SOC would monitor a variety of prevention-oriented systems and respond if one, or many, of these platforms alerted the team. Then they would spend time validating the alert, communicate with the system custodian, owner, or the end user, and if the situation were an incident, they would respond.

The "reactive or detective only model or posture" from the 2000's is no longer effective today. Today's SOC teams need to change their focus, assume that there is a likely compromise, become detection oriented, and *proactively mine* the vast amounts of data coming into their systems and actively look for patterns of intrusion and misbehavior. Proactive threat hunting is an ideal career and skill development path for SOC analysts. Once they understand all of the organization's data sources, know how to handle alerts, and demonstrate that they have established research skills, capitalize on that and get them involved in threat hunting. Depending on how the SOC team operates, you could have a SOC analyst can perform hunting one day a week, one week per month, or take a particular hunt pathway. Threat hunting is further defined on page 171, with numerous use cases beginning on page 61.

SOC Directly Supports the CSIRT Function

Today, the need to develop some form of Computer Security Incident Response Team (CSIRT) function can't be ignored. In many organizations or industries, a CSIRT is mandated by specific regulation. The need for a CSIRT is especially true with modern malicious software running rampant, automated ransomware, industrial espionage, criminal elements, nation state grade hacking teams, and host of other aspects of digitally based asymmetric warfare. Sounds sensational,

doesn't it? Today's cybercriminal will exploit any weakness they find to extract untraceable digital crypto currency from any potential victim. Regardless of the degree of sensationalism, there are several reasons to advocate for and build a CSIRT function in your organization, which is in turn supported by the SOC function.

1. The SOC provides an active detection capability that should enable early response and limit the long-term impact from an incident. The CSIRT can then coordinate responding to an incident with the goal of identifying, containing, and reducing incident impact. Further, the CSIRT function can return Indicators of Compromise or IoC's back to the SOC so that the SOC can perform historical data analysis, such as searching for internal systems that communicated with a found IP or domain name. Thus, a more mature CSIRT and SOC team can capitalize on a Threat Hunting capability in order to seek out and find a malicious agent that was recently on the network.
2. The CSIRT influences, supports, and fully leverages the security spend. It helps to ensure that the SOC tooling is in place will support incident response and Security Operations. Or put a different way, having a single CSIRT should ensure that the best tool for the environment is in place, can be leveraged, and others are decommissioned in order to maximize the dollar investment.
3. Maintain objectivity when interacting with internal staff, classifying incidents, and prioritizing the response process. One of the challenges that a CSIRT will need to deal with is staff relationships and maintaining objectivity. Like the HR function, which is charged with enforcing company policy and procedure in a uniform and consistent manner, the CSIRT needs to be objective, perform its work to protect the business, and avoid playing favorites.
4. Lastly, regulation. There are numerous aspects of the business environment that mandate an incident response capability. These include, but are not limited to: HIPAA/HITECH, PCI DSS 3.2, and Sarbanes Oxley compliance.

Metrics for the SOC

"What cannot be measured, cannot be managed."
- W. Edwards Deming.

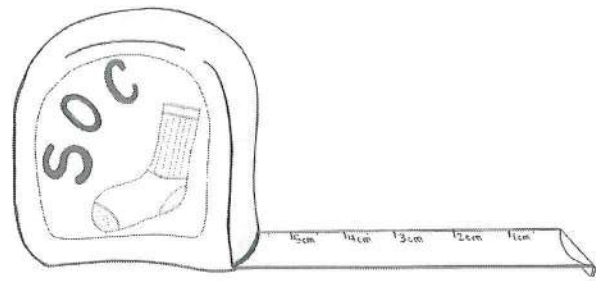
"Not everything that counts can be counted, and not everything that can be counted counts.""
- William Bruce Cameron

Mature business operating units and enterprises utilize various methods to measure the operating units effectiveness. The SOC is no exception. The question is how do you get there and avoid toxic metrics that demotivate your staff?

In the book Pragmatic Security Metrics, W. Krag Brotby and Gary Hinson make several key points about metrics. Some of their key definitions are listed here from chapter 1.6:

1. Instrument: short for "measuring instrument," that is, a device for measuring.
2. Measure: (verb) to determine one or more parameters of something.
3. Metric: a measurement in relation to one or more points of reference.

In Section 2.6, they state "Having valid metrics enables business managers to make rational, sensible, and, for that matter, defensible decisions about information security. No longer must they rely entirely on advice from information security professionals or generic good practice standards, laws, and regulations."



In Section 3.2, they state "Metrics are primarily a decision support tool for management. Good metrics provide useful, relevant information to help people—mostly, but not exclusively, managers—make decisions based on a combination of historical events (the context), what's going on right now (including available resources and constraints), and what is anticipated to occur in the future (the change imperative)."

There are numerous metrics and measurements that can be developed and applied to a SOC. When it comes to designing a metric, there are several criteria.

Take these criteria into account as metrics are developed.

1. Metrics should be relevant to the goals, objectives, and the mission of your SOC as a business unit. This means you should be able to describe what you do numerically, and also explain what you don't measure.
2. As you evaluate your data, security use cases, and metrics be sure to develop a roadmap of what you will measure, how those measurements will be captured, the tie in to the IT General Controls and security program, and the business value chain where possible. Metrics can then provide guidance for decision making.
3. Be sure that what you are measuring will drive towards a a measurable outcome. Don't measure for the sake of measuring. Every metric should inform a consumer and seek to either to change behavior or demonstrate

that current behavior is operating well within established procedures. Here, you should be clear on any action that you expect a consumer of the metric to take based on the measurement.

4. A metric should support a "control", and therefore should match up to your ITGC program. If you do not have one, look to standards like the ISO 27002.
5. Bad data is marginally better than no data at all. If you are not collecting any source data for what you need to measure, start there, but do not stop. Use good data to its fullest.
6. Avoid burdening the analyst with the need to record an excessive amount of using some artificial means to track what they do like a complex spreadsheet. Instead, develop methods to mine the SIEM platform, the workflow system, opening investigation coding, and alarm closure codes as they work through alerts. These methods provide an *economy of mechanism (EoM), because the analyst is using their native tool*. Also, following EoM principles pushes you to consistently leverage an internal capability of the SIEM platform, the less likely you are to cause mistakes.
7. Tell your story in terms of your business / organization. When telling stories, it is very important to remember the audience and use terms and definitions that they will understand.
8. Two key acronyms come to mind:
    a. Be SMART: Specific, Measurable, Achievable Relevant, Time-bound
    b. KISS, or Keep It Simple, Sam/Susan. This isn't actually meant to be cute. Rather, ensure that the name of the metric and the measurement scale is obvious. A metric the requires explanation is not likely to be an effective metric.
9. Determine how you can build a score card that measures the information and technical security posture of your organization. Whenever possible, build a tool to demonstrate how effective the technical tools work. You may not show this tool to management – but you should be ready to.
10. Work hard to avoid any toxic metrics, which are one that can end up punishing someone who "doesn't close alerts fast enough" or "only works on five cases per day". Instead, focus on metrics that

Table 2 Example General SOC Metrics

Metric[14]	Definition and Notes
# Unique Data Source Types providing SIEM data	Defines how many different information system data source types are consumed and available for analysis. This measures how many of your technical systems and applications are instrumented. There is a corresponding percentage of coverage in unique sources / total sources.

[14] In the table, MTT means "Mean Time To".

Metric[14]	Definition and Notes
MTT Detect Data Source Issue	How long does it take to detect that a data source is not functional, which is affected by the volume and velocity of data arriving from the data source?
MTT Correct for Data Source Issue	How long to resume data delivery once an error is detected in data delivery?
Time to sweep the enterprise	The Security operations function should be able to check every host in the enterprise for IoC's, or the host's security state. Ideally the average time to interrogate the enterprise should decrease over time, and the percentage of completeness should increase (# investigated / total #).
% of apps under ALCE [15]monitoring (Non-AD Integrated)	Measures how many shared applications report account life cycle events. Metric assumes that the app inventory is known. Applications that defer the central directory either through native integration or LDAP query *are counted*. Desktop apps are not normally included in this metric – only shared, server based, or SaaS applications. This metric has an *implicit assumption: you don't need to measure applications that are AD integrated for ALCE events. You will need to validate this premise for your own organization.*
% of SaaS under periodic ALCE monitoring	Integrating SaaS into a SOC or a SIEM can be problematic. As a compensation and at a minimum, all SaaS applications should be periodically reviewed to ensure that users defined in the application both have accounts in the central directory or can be accounted for, and that there are no disabled organizational accounts which are enabled in the SaaS application.
MTT Close an alarm by Close Category	Measures the decision-making process by the SOC analyst to close an alarm when it can be explained, processed, escalated, is non-reportable, or non-actionable (example close codes). **WARNING**: This metric and ones similar to it can easily become TOXIC to your staff. Be very careful in adopting this metric. Instead, search for alternatives to show that the staff can effectively respond to a portion of alarm conditions on a daily basis.

[15] ALCE: Account Life Cycle Event

Metric[14]	Definition and Notes
MTT Forward an alarm up Tier	Measures the lowest level analyst response time to identify that an alarm requires further action or resolution by the next level analyst or application SME.
MTT Open a formal Incident	The SOC function may open up a formal incident at any point in the alarm review process. This measures overall SOC and Incident Response team's capability to detect that an alarm or another condition is indeed a "security incident".
MTT Implement a use case	Measures how long it takes to define, document, instrument, and train on a specific SOC Use Case (see **Security Monitoring Use Cases by Data Source** beginning on page 61 for more information on SOC Use Cases).
# of Implemented Use Cases	Each organization will have a defined set of security conditions that SOC is able to handle with a supported use case or a response playbook. This metric measures SOC capability and coverage. Cautions: do not count the number of "detections" from your SIEM platform, or "alerts". Rather, take those into account and define your SOC playbook. This metric also guides how much training is required for new analysts and how well you are doing at mining your source data. Also, "# of "new" event conditions converted to Alerts can count here, but may not warrant a full use case.
# of Use Cases (rule) that never fire	While it is true that you can create, and test, a notification process for a rare condition, the SOC should keep an eye on use cases that never cause an alarm or don't prove out over a reasonable period, say one month. Try to keep use cases that never fire minimized.
# of Events Received	As a raw number, this metric isn't tremendously valuable. A better number is to measure events by severity, priority, or criticality – and don't get confused here.
# of Alerts by Severity	Based on a combination of your SIEM platforms.
# of high severity alerts not reviewed after 8 or 24 hours.	Measures how well SOC does at putting attention on all "high severity" or "high priority" alarms, per shift, and per day. If the frontline analysts are not capable of keeping up with alarms, then consider adding more staff, improving automation, and put attention to defining "close/escalate" criteria on these alarms. Once done and under control, this metric/measurement would push down the severity levels.

Metric[14]	Definition and Notes
Rules Tuned to Minimize False Positives (per week/month)	Every SOC should have at least one staff member who spends some time improving the notification rules within the platform. The better a SOC tunes the platform, the better it is demonstrating understanding of the environment.
ATT&CK Coverage by Phase	The MITRE ATT&CK matrix (page 174) is a knowledge base for understanding adversary behavior and the attack life cycle. This matrix can be used to evaluate how well the SOC and current instrumentation can identify presence of an attacker on the network.

Incident Response Metrics are not the same as SOC metrics because they pick up where alarm processing ends in many cases. In other words, when the SOC identifies a true incident, they will turn that over to an Incident response function.

Table 3 Example Incident Response Metrics

Metric	Definition and Notes
Cost per incident	There are two dimensions to cost: One is an accumulation of non-FTE costs contributions, such as paying for credit monitoring during a data leakage event. The other is the number of FTE hours lost.
MTT to Detect a Security Incident	How long does it take for the SOC to review an alarm and determine that it is, in fact, some sort of incident?
MTT for Detect to Contain	Once a security incident has been verified, several steps are taken to determine how to "stop the bleeding". Some issues can be easily contained, such as removing an infected single computer and replacing it, some cannot such as changing the codebase for a complex application.
MTT to expel an intruder	Once a true intruder is identified, meaning a real adversary, how long does it take for the security team *as a whole* to push the intruder out of the network. Be careful to analyze and report this correctly for your environment to ensure that consumers understand that there are more decision makers than just the SOC involved in this metric.
Incidents opened and closed	These should be trailing numbers, meaning as incidents are opened there should also be incidents being closed. These are measured per day, week, and month.

Metric	Definition and Notes
Avoidability of an Incident	Incidents should conclude with some form of Lessons Learned function, meaning that as a result of an incident the security posture of the organization is improved to the extent possible. If it is determined that the incident was avoidable if a common security practice was in place, report it.
Thoroughness of eradication practices	Measures whether or not the original compromise event, or one that is substantively the same (like a remote exploit) is observed subsequent to the first occurrence.
MTT Notify Principle, System Owner, or Custodian (Incident metric)	The recipient of an alarm condition may be a little hard to track down. There are at least three possible recipients – a principle within the organization, such as an operational director responsible for the affected system(s), an actual designated system owner, or a custodian such as a system administrator. SOC should define its escalation points and determine how to measure how well and quickly it communicates to the designated recipient.

SOC Training, Skills, Staffing, and Roles

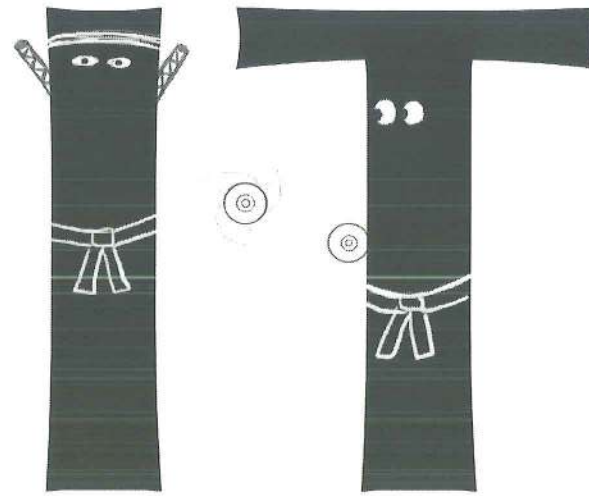
Effective security operations teams require technical skills, should possess certain personality traits, and require product training by role. Staffing our SOC team is critical to success. It is important to understand that you cannot "make" ninja grade incident handlers and SOC analysts who can synthesize a dozen data sources in real time and find "the bad guy" after completing a one-week course and passing an exam. That type of skill only comes with "time in the game". What you can do is develop people, provide them the opportunity to grow, and develop strategies to keep them in the Analyst seat. You can train people to respond to specific alarm types, handle specific cases, and work specific processes. There will always be "task driven" work that needs to be done by some level of SOC analyst. Playbooks make this level of staff effective, which in turn gives them success, and that leads to staff who want to do more. Those are the people you want to identify and grow.

SOC Onboarding and Initial Training

Training for SOC will come in several flavors:

1. **Product skills**: These are actual vendor solution training. This is achieved through vendor offered on the SIEM platform itself or a major technology vendor such as the Cisco CCNA Cyber Ops program, because that course material targets the actual products in use.

2. **Vendor neutral skill development**: These skills should cover the job role, tasking, and concepts that the analyst needs *to do the job regardless of the technology platform.* There is no shortage of education available such as: college courses, certification courses like EC-Council Certified Security Analyst, ISACA's Certified Information Security Manager, CompTIA Network+, Security+, and CompTIA Advanced Security Practitioner; SOC focused training by SANS; CyberAces; and various Q courses offered by Security University. For readers in Europe, look into European Information Technologies Certification Academy (EITCA) and European Union Agency for Network and Information Security (ENISA).



3. **On the Job**: Internal training relevant to the position and team, internally developed and delivered.
4. **Success and Failure**: there's nothing quite like "getting it right" and stopping the bad guy or missing the bad guy and finding out afterwards.
5. **Cyber Range Operator training**: intensive exercises offered on a dedicated large-scale lab with a focus on hands on skill development.
6. And many more.

Your training program should include a mix of items from the items above, with the goals of ensuring that your analysts develop all of the skills listed in the next section. As new staff are onboarded onto the SOC team, you should follow a well-defined structure in order to ensure that they will have the best possible opportunity to succeed. Below is a sample onboarding model for a SOC analyst I've used in several organizations. The primary goal of your orientation program should be to develop the analyst so that they can be self-sufficient at their particular level after four to five weeks through an onboarding program.

Week	SOC Analyst Orientation
1	• Organization onboarding • Operator Level product specific orientation (usually 2d) • Side by side observation by new person with a current analyst • New staff reviews a "good" and "bad" write up (report) of each major alarm type that they will work at their level (OTJ)
2	• Side by side alarm review and analysis – new staff works through alerts and is partnered with senior staff • By the end of the week, new staff should be able to handle several alarm types on their own
3	• New staff is introduced to more complex case types by reviewing "good/bad" reports, starts handling more of them • More advanced product training focused on system health, uptime, and data flow monitoring • Schedule time w/ each next level analyst/case lead to expand knowledge of various SOC areas
4	• Shift and responsibility rotation – work various shifts • Longitudinal "weekly" report contribution
5	• Skills acquisition testing, which should consist of scenarios that support assessing how much the analyst has learned

SOC Analyst Skills

Historically, successful IR and SOC people need to have a diverse IT background, should have a few years in the IT game, do require continual training, and have a variety of skills in order to handle the breadth of cases a SOC will face. In particular, an analyst needs to understand how to "connect the dots". Connecting the dots is a skill developed over time. Analysts can be educated on understanding a given data source, but must acquire the skill to understand one event in context of another over time.

Analysts also need to learn how to efficiently preserve case relevant information as they work through an alarm investigation. An effective technique is to capture relevant data while they write a ticket or draft an incident report, instead of attempting to reconstruct data after an incident is over.

Today, in order to address the skills gap, find people who are: naturally curious, can think abstractly, often took things apart and put them back to together during their formative years, have a ***strong attention to detail***, can "connect the dots", and lastly who can perform research.

Below is an Analyst Skill Development Recipe which came from surveying over 30 SOC and Security Managers during Q1/2017, and then refined as I implemented that advice. SOC Analysts need to understand:

1. **The "Attack" process and phases:** Recon, Scan, Initial compromise, Establish Persistence, Command and Control, Lateral Movement, Target Attainment, Act on Objectives, Exfiltration, Covering Tracks, Leave without Trace. The MITRE ATT&CK framework is invaluable learning tool in this space as well as the Cyber Kill Chain as described on page 184.
2. **Ethics:** Ethical behavior means that they work within their limits, ask for assistance, do not overstate conclusions, never fabricate an opinion on weak facts, keep confidential all of the data they use, and other professional responsibilities.
3. **Organization specific data familiarity:** Familiarity with your organizations source data (event types and fields) so analysts can tease out fact data that can "connect the dots" to the alarm event in the overall context of the event stream for the suspect machine or user. *This is one of the hardest skills to develop.* As a new data source is added into a given system, engage the EDIS process: the senior level analyst must prepare an overview of the data for the remainder of the team.
4. **System:** Technical system access control capabilities and how a technical control can be applied.
5. **Firewall:** Firewall Principles such as log types and what the log row records; rule attributes; actions such as block, allow, reset, drop silently; interface meaning as it relates to flow; SNAT/DNAT, byte sent/received; and understanding security zones.
6. **Security zones:** In particular, security zones are specific to each organization. One can infer the meaning of a zone, but should not because a zone term may not be consistently applied by the various admins or engineers. As firewall rule sets are built, the firewall team must document what they mean by a given zone and their underlying assumption about traffic flow. Active Directory admins need to do the same thing for organizational unit definitions. In this way a firewall zone named "eComDMZ" can be connected to an ADOU named "WebSalesDMZ" when the definitions state "severs used to support web sites used for ecommerce".
7. **PCAP collection and analysis:** Network data collection and the use of tcpdump as the data collection to and then WireShark/TShark as the analysis tool.
8. **Forensics:** Forensic principles
    a. Gathering data on system following the order of volatility
    b. Establishing and maintaining Chain of Custody

   c. Documenting actions taken, data extracted, time shift, and timeline reconstruction
   d. Locard's Exchange Principle
9. **Hardware:** Hardware platform: PC, Server, Router, Switch, TAP, disk boot, MBR, and identifying volumes.
10. **Incident Handling:** Incident Handling process: Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned. Once someone understands these points, they need to be able to write an executive summary of a case.
11. **Investigation:** Investigative processes as described in the Alarm Investigation chapter.
12. **NAT Issues:** NAT translation and the complexity of locating the true source IP.
13. **Network Application Protocols:** DNS, SSH, HTTP/S, SMB, NTP, DHCP, FTP, SSL, SMTP, POP3, IMAP, and SIP.
14. **Network Protocol understanding:** ICMP, QUIC[16], IP, TCP, UDP, GRE, BGP, SCTP, and ARP.
15. **NIDS:** Network Intrusion Detection System (NIDS), ruleset categories and how to read events generated by an IDS using rulesets such as the Talos Snort rule set (formerly known as SourceFire VRT) and the Emerging Threats Pro feed.
16. **Application Portfolio:** Organization Application inventory, application behavior, and PoC's
17. **Policy:** Organization Policy/Procedure *and the ability to politely articulate and enforce PnP.*
18. **OS proficiency:** Windows and Linux: services, startup, log files, network connections, registry, and process identification, and how rights and permissions are applied to both Windows and Linux.
19. **Scripting:** Programming, focused on admin and data reduction scripting skills using PowerShell, Python, shell, Perl, or a similar utility focused scripting language
20. **Report writing**[17], including spelling, grammar, Word usage, and the ability to write a summary that answers "Whom, What, When, How, and Where", and make a reasonable assessment of "Why".
21. **HTTP, HTTPS, and Web Browsers:** Understanding how the most dangerous application, a browser, works is critical to analyst success. The application itself more commonly attacked than the perimeter, because it is more successful. As evidence, consider the rise in phishing tools and attention on socially engineering the end user to entice them to click. The ability, and

[16] Quick UDP Internet Connections, a protocol promoted by Google that supports multiplexed connections between endpoints and is supported by almost 1% of webservers (August 2018).
[17] Chris Sanders has the only course the author knows of on this topic.

consistent habit, of identifying the source user when a proxy server is involved in an alarm is essential. When an alarm occurs or a proxy is identified as participating in an intrusion, the analysts need to always attempt to find the user, user agent, or system that generated the traffic.
    a. Understand user agents, HTTP status codes, URLs, and browser redirection.
    b. Research the URL data revealed through the proxy.
    c. Discretion when researching user browsing habits.
22. **OWASP Top 10:** Following on the need to understand the browser is the need to understand how an Internet facing application is attacked.
23. **All of the skills measured** in the CompTIA Security+ certification and many of the Network+ skill areas so that a SOC analyst is terminology compliant.

SOC Analyst Traits

SOC Analysts need to have specific personality traits in order to be effective at their jobs, as listed out below.

1. **Natural curiosity:** SOC Analysts will be faced with an ever-changing array of problems, situations, and new data sources. Find people who took things apart and put them back together when they were young, especially if it worked when they were done.
2. **Organizational skills:** An ability to perform "rapid research" that allows them to separate wheat from chaff, and in particular the ability to determine if an alarm is likely to be real, based on the alarm and data surrounding the alarm. For example, if a NIDS rule fires from an attack that was popular three years ago, what conditions must exist today to permit that attack to succeed?
3. **Abstract thinking:** In particular, the ability to read intrusion events like alerts from a Snort/Suricata system and a Palo Alto firewall and correlate them in near real time to other data sources, visualizing activity patterns in their minds.
4. **Contextualize large data sets:** The ability to reduce a larger volume of data down into information, in context. More specifically, when faced with an alarm *right now*, determine if that alarm is relevant to a larger context.
5. **Communication:** Perform data summarization and commonality detection such that a group of original facts can produce information, and then articulate how the information proves or disproves a hypothesis.
6. **Ego:** A small ego, but not small enough that they don't take pride in their work.

SOC Roles

There are several roles that need to be staffed in a security operations center. Depending on the size, scope, and budget, a SOC may have more roles defined. Roles will also have defined interactions with other key roles, because a SIEM platform and NSM platform *actually* have user community – the SOC analysts – who are supported by the engineering, architecture, and process support side of the SOC team.

Table 4 SOC Roles and Functions

Role	Duties and Responsibilities
Analysts	As defined in SOC Layered Operating Models on page 52, this role is the primary SIEM, NSM, and log management system user. Analysts may function as incident handlers or may directly support the CSIRT function.
SOC Developer	There will be a need to write "utility" software, such as a log parser, a monitor script, an add on component for a dashboard, or a lookup tool. Many SOC's would benefit from being able to utilize development talent and skills.
Shift Lead	A senior analyst who ensures that all shift responsibilities are met and is a resource for SOC analysts. Shift leads also handle communications external to the SOC, and therefore should have well developed people and communication skills.
SIEM Engineer	Engineers install, maintain, and upgrade the SIEM platform and its operating systems. They also implement use cases, provide troubleshooting, and configure device support. Advanced SIEM engineers can also build parsers for unsupported devices, which usually involves knowledge of regular expression parsing and SQL queries. Lastly, a SIEM engineer needs to document the EDIS process and prepare internal OTJ training for all SOC staff so that analysts properly interpret event data.
Security Process Engineer/ Analyst	A process engineer has a security focused system analysis role. They perform analysis to develop, define, and test use cases, train the analysts on supporting use cases, and help to develop reports.
SOC Manager	This is a day to day managerial role for the SOC team. The primary customer is the CISO. The SOC manager implements strategy and process defined by the CISO.
CISO	The CISO owns the information security management program. The SOC team, SIEM, log management, and NSM platforms support many aspects of the program, such as incident response, continuous monitoring, and log management. CISO's

Role	Duties and Responsibilities
	must understand business requirements and expectations (without this understanding, they are likely to fail).

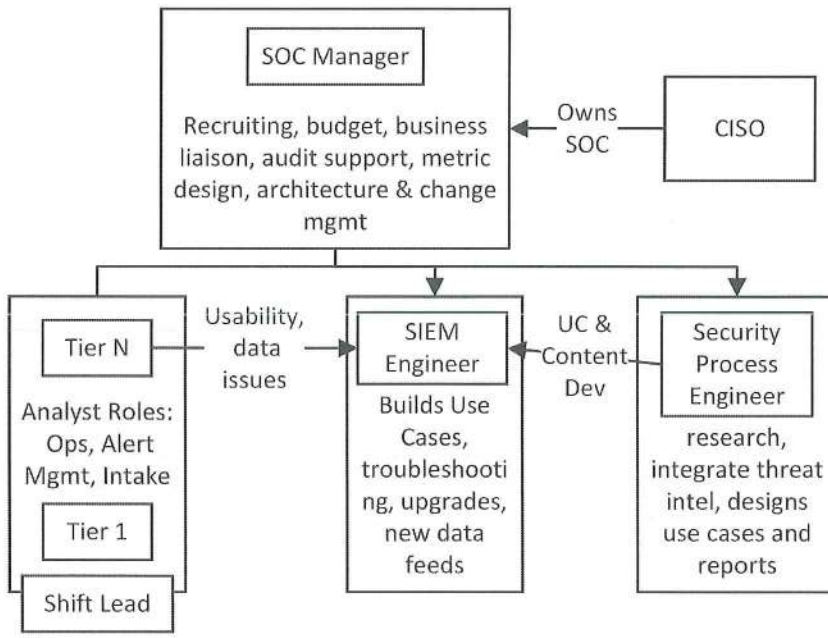


Figure 1 SOC Roles and Relationships

SOC Layered Operating Models

A small security team of just a few people rarely follows a hierarchy. The team just work each other to monitor the environment and respond to alarms. As a Security Operations Center matures and grows in size and breadth, it usually develops into some form of a tiered analyst job structure that reflects staff skill base, alarm conditions that are worked, pay level, SOC service areas they support, and training commensurate with their job level.

This section describes a two and three-layer approach, with the word "layer" being used as a placeholder. Job titles may be Tier 1 Analyst, Tier 2, and Tier 3 Senior Analyst, or titles like Junior Analyst, Analyst, Senior Analyst, Lead Analyst, or SOC Shift Lead. Regardless of the actual title, the essential concept is that there are layers that relate to the analysts' skill base, comfort level, and their ability to respond to alarm and event conditions which in turn ties to pay and responsibility. Layers models also provide a way to measure progress and provide job advancement by title and pay commensurate with the skill base.

Regardless of the model and title, there is almost always a formal management layer for the Security Operations team.

What is essential to whatever stratification model your SOC uses is that each analyst must be willing to ask for help if they find themselves outside of their depth when working an alarm or handling a ticket.

Two Layer Model

In a two-layer model, the front-line staff is most often staffed with seasoned security analysts while the second layer is staffed by engineer level staff who handle more complex cases or require more complex analysis. Above these two operational layers the SOC management. Each of these levels will have end user and management interactions at some point.

Table 5 SOC Two Layer Model Roles and Responsibilities

Layer	Example Duties and Responsibilities
1	• Real time event and alarm monitoring that follows a standard operating procedure for a wide variety of alarms • Phone intake for initial case support (phone, email, webform) • Run reports • Monitor SIEM system health, data feed checks, and keep an eye on the system(s) as a whole • Gather key data, feed many case types to the ServiceDesk, handle and process more straightforward alarm conditions. and escalate more difficult or complex cases to the next layer *after* they have collected some initial data • Close some cases based on well-defined criteria. • Certain analysts may be asked to perform "longitudinal analysis".
2	• In depth analysis of alerts and events escalated by Tier 1 • Perform complex analysis and research on alerts and events, such a previously unseen alarm condition from a new data source • Take a longitudinal view of event patterns, searching for longitudinal security issues • Coordinate incident management • Tendency to specialize for certain alarm types, systems, or areas of the business • Synthesize vulnerability data • Performs daily or weekly threat hunting activities

Three Layer Model

In a three-layer model, responsibilities are more stratified as the organization attempts to respond to increased need for more coverage and balance that with staff costs and skills or a need to provide more coverage like adding in overnight and weekends. Therefore, a higher separation appears to enable junior analysts to be effective for the SOC.

Table 6 SOC Three Layer Model

Layer	Example Duties and Responsibilities
1	• Real time alarm monitoring, using a well-defined SoP or other standardized Operational Guidance document • Phone intake for initial case support (phone, email, webform) • Monitor system health and data feeds • Gather key data, escalate most cases to the next layer if they can't resolve the case quickly or close some cases based on well-defined criteria.
2	• Handle escalated alarms • In depth analysis of alerts with a determination whether to forward or not to Tier 3 • Review the inventory of daily event types received by day, looking for patterns that may indicate a security issue • Take a short-term view of event patterns in support of alerts (a higher degree of alarm awareness) • Support incident management data requests • Tendency to specialize for certain alarm types, systems, or areas of the business • Perform data feed monitoring and basic system daily checks • Synthesize vulnerability data • Perform daily or weekly threat hunting
3	• In depth analysis of incidents and cases, individually, by day, or longitudinally • Manages incidents, may function as an incident coordinator/commander, or may be a second in command • Take a longitudinal view of event patterns in support of alerts and areas of the business or a client • Has operational and longitudinal responsibility for specific areas of the business, or a client area/business unit • Performs specific daily threat hunting activities

Layer	Example Duties and Responsibilities
	• May perform memory or dead disk forensics, or may supervise outsourced forensics

SOC Maturity Curve Using the CMMI

Security teams go through a variety of growth phases and stages, which often happen very organically. SOC's usually start when management hires someone either deliberately or as a response to an event because they need to "get a handle on security". That first hire then hires or pulls in a few people from IT to form the security team, and as a natural outgrowth some form SOC team is formed. Muddled in these organic steps is a focus on "buying and implementing a SIEM", which gets funded as a project[18]. Then you have a SOC in someone's shared office, or a couple of people get some space in the Network Operations Center (NOC). All of this happens while the people involved are doing their "day job".

These types of organically grown teams often miss a critical factor in their formation: the SOC function wasn't deliberately created following a needs assessment or a formal model. As a result, varying aspects of SOC services described on page 17 operate at different levels. The SOC staff members write different reports in response to the same alerts so results are not predictable, everyone has different skill levels, the operational structure is not internally consistent, and staff may be frustrated.

How is this problem solved? One well respected method is to apply the Carnegie Mellon Capability Maturity Model Integration (CMMI). There are five maturity levels across in the CMMI, with each level representing an evolutionary plateau in terms of process improvement. Normally, CMMI is applied holistically in an organization across up to twenty-four process areas, so using it needs to be adapted and focused for a Security Operations organization. *It may not be necessary, or even desirable*, to push all capabilities and processes to level 5, because each maturity level is measured by meeting a set of objectives for the process.

[18] Remember – a project is a one-time event while running a security operations team is an ongoing business and IT function. They are very, very different.

Table 7 CMMI Five Level Maturity Model

Name	Characteristics[19]
1: Initial	Processes are ad hoc, chaotic; success depends on heroics as opposed to following a defined process. Services often work but exceed budget, time, schedule. Success is not repeatable, and heroic action frequently saves the day.
2: Managed	Workgroups (the SOC) define processes, create work plans, monitor their processes, and meet a set of "contract requirements" with its customers (the business). Configuration management is implemented with quality assurance.
3: Defined	Defined processes for managing work are used, well understood, services, procedures, tools in place. Sound project management is implemented into each process set. Difference in L2 and L3: process/procedure can be quite different in each instance of the process, whereas L3 procedures are tailored for the workgroup.
4: Quantitatively Managed	There are quantitative objectives for quality and process performance, which are used to manage processes. Measurement statistics are collected on specific subprocesses. Difference in L3 and L4 is focused on predictability.
5: Optimizing	Focused on continuous improvement.

There is also a formalized model available to assist in assessing a SOC's maturity level. The SOC-CMM was created by Rob van Os, MSc. as part of his Master's thesis work[20]. Rob's methodology, tooling, and spreadsheet is setup to evaluate the SOC across five dimensions and twenty-five aspects. The five domains are Business, People, and Process, which are evaluated for maturity. The other two are Technology and Services, are evaluated for both maturity and capability. You can use this tool to evaluate your current state and develop a road map of

[19] These points are adapted from the SEI CMMI for Services, 1.3. Note that as this book was written, 2.0 was just released, so you should review 2.0 material. 1.3 URL: https://resources.sei.cmu.edu/asset_files/Webinar/2010_018_101_22253.pdf (8/16/18)

[20] The SOC-CMM is available at Rob's website: https://www.SOC-cmm.com/

how to mature your SOC. Rob's site is https://www.SOC-cmm.com/. Rob's tools are gaining quite a bit of traction.

Measuring Data Source Integration Maturity Levels

To apply the CM CMMI, the SOC can consider how it integrates a data source into the SIEM and its operational process as a process that must be well managed and repeatable. As a primary requirement, in order to move that ad hoc process to a mature well-defined process, the SOC needs to systematically accept data, mine that data, and ensure that the SIEM platform is maximized to the fullest.

Data Input as an Initial Process (L1) is characterized by:

- Getting data into the SIEM can be very ad hoc. A well-meaning system custodian sets up a syslog feed and lets SOC know that new data is arriving.
- Someone in SOC works with system custodian to make sure that the systems data can be gathered into the SIEM.
- Someone else in SOC works with the SIEM vendor to make sure that data is parsed.
- Data survivability baseline is established so that if the source system stops providing data, it can be detected "quickly enough."

Data Input as a Managed Process (L2) is characterized by:

- Data input goes through a consistent process. Source system instrumentation is well defined, if there is a need for custom by the vendor it is planned, and a synthetic transaction is setup (see p. 193).
- The source system is fully exercised so that all of the security and operationally relevant events are logged.
- Once the data arrives, SOC builds source specific alarm conditions based on that data source.
- The SOC is trained on the breadth of event types so that the team can fully utilize all of what the source system can provide.

Data Input as a Defined Process Characteristics:

- Organizational policy and process artifices require that data source input is integrated into the organization. For example, policy requires logging to the SIEM and/or log management platform is required, and a check to see this is setup properly is integrated into the Configuration Management process.

- The SOC manager ensure that all users have completed onboarding for each data source and confirms that there is equivalent understanding how to use the data.

Measuring Alarm Processing Management Maturity Levels

To apply the CM CMMI, the SOC can analyze how it handles Alarm processing. Alarm processing is another good example of a service that should be consistently delivered. This service as it is the result of processing data input into the SIEM and then acted on by the SOC analyst. In order to realize this service, all aspects of the SIEM and SOC are part of the service delivery: data input, parsing, health monitoring, alerts raised in response to events, analysis of alerts to validate that they are true positive, severity is responded to, based on the impact to the organization, incident response is activated to the degree needed, and the resulting incident is tracked through to resolution.

That is quite a mouthful! Below is an example of how alarm management can move through a set of maturity steps, going from an ad hoc maturity level to a well-defined level. The primary requirement is that the SOC read, review, and respond to alerts as rapidly and completely as possible.

Alarm Processing as an Initial (L1) Process has these characteristics:

- SOC Analysts review alerts as they arrive, with some notion of priority and impact. High priority alerts which are likely to be true positives receive attention such as reporting for resolution, data owner and system custodian notification, or routed to Tier 2 for further investigation.
- SOC analysts may or may not be consistent with each other in the decision-making process on alarm resolution. During times of high stress, alerts are not consistently managed. Critical alerts may never be evaluated in a timely manner.
- Alerts may occasionally be reviewed daily in the aggregate such that critical alerts always receive some form of treatment as a "stop gap".

Alarm Processing as a Managed (L2) Process Characteristics:

- Event data is reviewed in order to find other alarm conditions in order to improve detection capabilities.
- Alerts are tuned so that false positives can be minimized. This part of the process should influence the source system in a feedback loop.
- Alerts are treated consistently by all members of the SOC, with most of them mapped into incident playbooks or other supporting processes.

- Alerts are resolved, routed, or investigated as they arrive within a certain defined time frame – say 30 minutes for critical, one hour for medium, and within 4 hours for low.
- The alarm board is reviewed every shift to *ensure* that all "Critical", "High", and "Medium" alerts receive attention.
- Alarm analysis and processing enter a tuning process so that lessons learned and system custodian feedback improves the alarm management process.

Alarm Processing as a Defined (L3) Process has these characteristics:

- As data sources are integrated into the SIEM and SOC processes, they are mapped into the taxonomy, the review processes are defined, and the SOC staff is fully trained on the data source and the alarm conditions it brings to light.
- The SIEM platform is augmented with orchestration and automation in order to improve analyst responsiveness and put better data in the hands of the analyst.
- Alarm and Event data is used to guide a threat hunting program, and the threat hunting program in turn guides and improves alarm generation capabilities with the corresponding SOC processes.

Example SOC Turnover Shift Check List

Start with this list for your SOC turnover. Cover turnover at the beginning of each shift. Each shift needs to summarize running alerts, incidents, and follow up items for the following shift as these are critical for situational awareness. Review this list and define what analysts do per shift, develop an organization specific model, and then update the end of shift turn over. Some SOC teams aren't staffed enough to provide an overnight function, so at the beginning of the morning shift there should be an immediate review of overnight alarms. Determine how to incorporate this point in your environment.

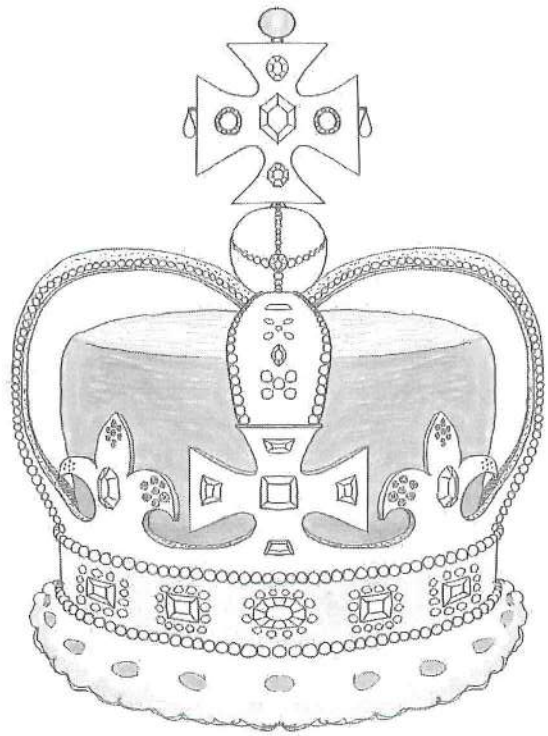
1. Turnover from prior shift, which should include:
    a. Key events
    b. Status for ongoing incidents
    c. Staff outages
    d. Major data issues
    e. Any system stability issues
    f. Relevant communication topics
2. On the normal maintenance day, review the scheduled changes so that the SOC won't over react for alarm conditions that can be explained by a change running through change management.

3. Quick review of yesterday's alerts so that any recurrence or repeat conditions which may reveal a repeat event are easily identified.
4. Review the daily briefing, which should cover topics like:
    a. New alerts instrumented in the system
    b. Data sources leaving the system
    c. New data sources coming into the system
    d. OTJ training gaps

Security Monitoring Use Cases by Data Source

As you read through this chapter you should:

1. Think about common scenarios for attacks and adversaries who target your organization. To aide in this process, this chapter introduces a scenario, an attacker plan, and a defense plan.
2. Ensure you know how to get the data to support the use case points.
3. Gather vendor and product documentation so analysts can easily look up event detail as they use event data when researching an alarm.
4. Determine the risk that the use case is designed to address for the organization.
5. Decide and document how the analyst will respond to the use case.
6. Review the enterprise data inventory survey in the SOC planning chapter.



Above all else keep looking for "evil" because it is looking for you (or, rather, your data that makes up your organization's Crown Jewels).

The Scenario

Every organization has a wide variety of types of data that they can collect. As the SOC considers collecting data, evaluate that data against the actual needs of the organization, how that data will be used to detect an adverse security condition, how close the data is to the end user focused application, and how user attributable that data is. To illustrate these concepts, this section will walk through an example of systematic intellectual property theft conducted against VictimCo.

The Setup

First and foremost, VictimCo is in possession of valuable data. After surveying its business environment, value chain, and identifying its valuable data, VictimCo makes several key determinations:

1. There are well-known users with a public presence, ripe for targeting. Of this user population, a small number of them frequently travel and speak at conferences. There is also an active traveling business development group.
2. Critical Intellectual Property is spread out on the network, meaning it is not consolidated. Further analysis here indicates that there is a mix of storage platforms. The list includes SAN, NAS, OS based shares, and web-based collaboration sites. A significant portion of IP is "Trade Secret" and not necessarily well marked.
3. Even though there is a web proxy, it is not configured with an overly restrictive policy.
4. More than 30% of the user population have elevated rights on their workstations, which is implemented by adding the user's primary login account to the "Administrators" group.
5. The firewall has a default by deny stance and logging is pretty good.
6. There are easy methods of data egress because outbound FTP is not restricted from the desktop and websites that allow users to paste data and then get a URL to that pasted data are not blocked.

The Attackers Plan to Find Data and Exfiltrate

VictimCo's team develops an outline how an attack would progress. An attacker who is after valuable intellectual property will need to build and execute a plan similar to the one outlined below.

1. Reconnaissance: Website scanning, performing Internet based "doxing" type research, and gathering candidate email addresses for a targeted phishing campaign.
2. Weaponization: An attacker would use a variety of tools to craft malicious payloads, such as PDF documents with droppers, macro enabled Office documents, or a website with malicious content.
3. Delivery: The most likely chance for success for an attacker is to send email with a convincing pretext. The pretext is written so the message appears to come from a professional colleague or someone with interest in working with them. For example, a phish campaign would indicate that they read a book or article by the recipient, express an interest for some collaboration, or suggest that the recipient may be interested in other articles, books, or sites. The links for these sites would include malicious sites. Often malicious sites would contain a link to a login site that looks like it belongs to the recipient's organization – but in fact it didn't – with the goal of capturing real credentials.

4. Exploitation, Part One: Once credentials were captured, an attacker would use them on any exposed site or may even attempt to login to a common VPN service.
5. Exploitation, Part Two: While gathering the username pattern from email addresses, an attacker may attempt to brute force or "password spray" any potential account against any exposed web interface that VictomCo has.
6. Installation (Remote Access): Once the attackers had a minimal foothold they were capable of establishing persistence on several users' workstations, gather higher level credentials, move laterally within the network, and gain access to shares and sites with trade secret data.
7. Command and Control: With a foothold, the ability to run services and enable persistence, proxy service aware command and control agents that communicate out to C2 nodes can be installed on target systems.
8. Act on Objectives: Once sensitive data is located, it can be exfiltrated in numerous ways. Examples include uploading to file share sites, FTP on non-standard ports, email, direct transfer to using netcat are but a few.

The Defense Plan

The primary underpinning of the defense plan is to prioritize capturing user attributable data that matches likely attack vectors. Gone are the days of collecting millions of firewall records in hopes of analyzing that data. Modern data collection should be user attributable, be as close the application as possible, and provide execution context *because the modern attacker must live off the land,* which means they need to change the OS and use scripting languages present on the system. To quote Alissa Torres[21] from SANS, "Malware Can Hide, But It Must Run." Prevention technologies are certainly valuable tools and they do protect the network. However, we must assume that they will fail in the digital arms race so we must instrument for post exploitation detection. Furthermore, no advanced preventative technology can counteract a user who knowingly, willingly, clicks on a link in an email and ignores all the warnings.

Instrumentation	SIEM and Security Architecture
Use the dnstwist algorithm to develop an inventory of domains that look like VictimCo. These domains can be checked historically for site visits, email communication, and blocked in the proxy with a redirect. Twist domains can also be DNS sinkholed.	A search can be run to see if any system that uses a FQDN from the dnstwist list is hit. Examples – user made a DNS request, attempted to visit a website, sent or received email to one of these domains.

[21] https://digital-forensics.sans.org/blog/2016/10/29/malware-can-hide-but-it-must-run

Instrumentation	SIEM and Security Architecture
Confirm that email activity is logged and recorded at the SMTP conversational level, and configures these data elements to be fed into the SIEM from message platforms: Sent, From, Return Path Domain, To, CC, Subject Line, Attachment attributes (name, size, date), type, and Bayesian score	Should a user succumb to a phish mail, the SIEM/email system can be queried to see if other users also received the same mail and then the email can be removed from their inboxes.
Implement sysmon on high value user's workstations in order to collect process invocation (parent, child, path, command line). Registry Change – Sysmon Event ID 13	Create alerting that will fire if an office application or the email application opens up a scripting tool or a command prompt, which is a high value alarm and indicator that a user clicked through and opened a malicious email payload.
Update Windows domain auditing to collect command line path from 4688 events (detailed Tracking). Collect same into the SIEM using Windows Event Collection and Forwarding as a first phase approach.	Nearly the same functionality can be accomplished with command line analysis which is provided by the 4688 event. The threat Hunt team can perform long tail analysis of software executed and over time build a whitelist of known good applications and command lines. After that is done, they can then monitor by exception.
Update all Internet facing site to minimally log access attempts (success and failure) to the SIEM. (custom development in many cases).	Access attempts can enable several account management use cases – a new account successfully logged on, a brute force was successful, brute forcing is being attempted, and a spray attack (try one password for all possible accounts) can also be attempted. If the source machine is on the inside, it can be thoroughly investigated for C2.
Always on VPN (enhance security architecture)	Traveling users can be redirected back into the company network so that security controls and monitoring is actually effective.

Instrumentation	SIEM and Security Architecture
Update the Windows Workstation Presence Indicators as a second phase approach, once it is proven to work and the issues are resolved. New Service: Event ID 7045 Scheduled Task: Event ID 4698 Local Group Changes: 4731,4732,4733,4734	Windows has a native capability to centrally collect audit logs. At a minimum, several event types need to be collected which are known as "presence indicators": login, screen lock, reboot, screen unlock. After that: local group management. Finally, service state changes. These event types can be used to detect when workstations are used outside of normal business hours and for unauthorized changes, new accounts, and service installation – all underpinnings of persistence and lateral traversal.
Persistence detection: Autoruns (daily, for all workstations and servers).	An advanced detection technique is to consume the output of "autorunsc" into the SIEM, sort the data using Long Tail Analysis (or stacking) in order to detect any new persistence entries.

Once operating system data is collected, then focus on valuable network level trace data. Network level instrumentation should focus on chokepoints, flow data, and application support intelligence such as DNS activity, web browsing activity, and network flows between network segments. For example, workstation to workstation traffic is highly unlikely in most corporate networks.

Defining the SOC Use Case

After working through this scenario, the security team and IT work to jointly define use cases. This is a process that, if done well, will pay huge dividends down the road.

First, let's level set on the phrase "use case"[22]. A use case is "a set of **actions** or steps which define the **interactions** between an **actor**, which can be a person, a system, or a service, to a system in order to achieve a particular **objective**." A full-fledged use case template, tuned for Security Operations and focused on instrumenting the environment presented in this book on page 133.

[22] Adapted from the Wikipedia article on Use Cases

For a SIEM and a SOC team system, building a use case has several requirements based on the definition from above:

1. You must be able to describe the *observed* condition that is relevant to your *security posture and realizes the use case* (the **objective**).
2. The system providing data to the SIEM must be capable of *actually auditing* the desired behavior (observe the action by person or system **interaction**).
3. The system must provide the event record *with sufficient fidelity* to the SIEM (as defined under Log Record Data on page 223.) (define the **action**).
4. The SIEM must be able to process and present the event at the necessary level of granularity for the Sec Ops function (to measure or observe the **interaction** for the **actor**).

Briefly, the security focused use case development process is:

1. Understand how the use case maps to or supports a *Business Capability or a Requirement*: Uptime, brand protection, dozens of compliance requirements, fraud prevention/detection, IP theft, or minimize disruptions.
2. Design the question that the use case should answer. How would the attacker gain needed access, cause damage, exfiltrate data, or what accounts would they need to use?
3. Determine and test the data sources and the data elements that provide the visibility needed to answer the question.
4. Evaluate the data by establishing normal baselines and other analysis dimensions. Characteristics to understand include volume, peaks/lulls, outliers, averages, frequencies of types of data or specific elements, duration of normal behavior, and how do you find something "new".
5. Establish the SOC guidance and processes that will be used to filter out false positives from the baseline data. Ensure that guidance is developed to support identifying malicious use or operational issues.
6. Various techniques exist to visualize data such as bar chart analysis, graph analysis, simple timeline presentation, and other summarizations.
7. Many data sources naturally lend to building correlation rules where one data source compliments another.

With this definition and these criteria in mind, I'll define dozens of SIEM use cases that should be built out along with the corresponding event data that makes up the use case. Again, as in the rest of the content of BTHb:SOCTH, most of these use cases are based on experience and were implemented to one degree or another. This inventory of use cases is not exhaustive, and should not be considered "the definitive list". Hopefully as you look through these they will

assist you in implementing your own use cases based on your data sources and value chain.

Example: Web Presence Attack

Many organizations operate a web presence. According to the January 2018 Netcraft survey, there were 213,053,157 unique domain names, 7,228,005 web-facing computers on the Internet[23], all hosting 1,805,260,010 sites.

These systems can be as basic as an externally hosted public information sharing site to a full-fledged eCommerce and customer engagement platform. The illustration below provides a high-level view of the layers arranged horizontally, with the Cyber Kill Chain steps and common avenues of attack cutting across the layers vertically. At the business conceptual and contextual layer, the organization operates a web farm with multiple web servers. Frequently there is a custom developed application, which may or may not include packaged software or libraries as a component. For example, a site may have a product catalog, a blog site, and a contact page, each of which can use its own subcomponents. On the private, or login protected portion of the web presence, the organization can host customer order request, acceptance, and processing, customer engagement, and support ticketing. The organization may also host some hybrid applications, such as a forum which requires some form of registration that may be separate from the customer engagement account database.

Cutting across all of these technologies is an exploitable attack surface emerges. Some of the best guidance to understand this attack surface is the consensus driven OWASP Top 10 Most Critical Web Application Security Risks list, which is updated every few years. SOC and IT can leverage this list to determine what type of monitoring may reveal an exploit, what type of security monitoring capability needs to be deployed to protect the web presence. From a Cyber Kill Chain perspective, the attacker needs to perform reconnaissance against the site and its numerous components. Not shown is the weaponization step, as attackers develop attack capabilities using other environments. Once the attack technique is available and an exploit is developed, the attacker can successfully exploit the site, a component, or an underlying library and install their own persistence mechanism such as a backdoor shell. As mentioned elsewhere, this scenario shows why it is necessary to keep informed about updates to Metasploit.

[23] https://news.netcraft.com/archives/2018/01/19/january-2018-web-server-survey.html

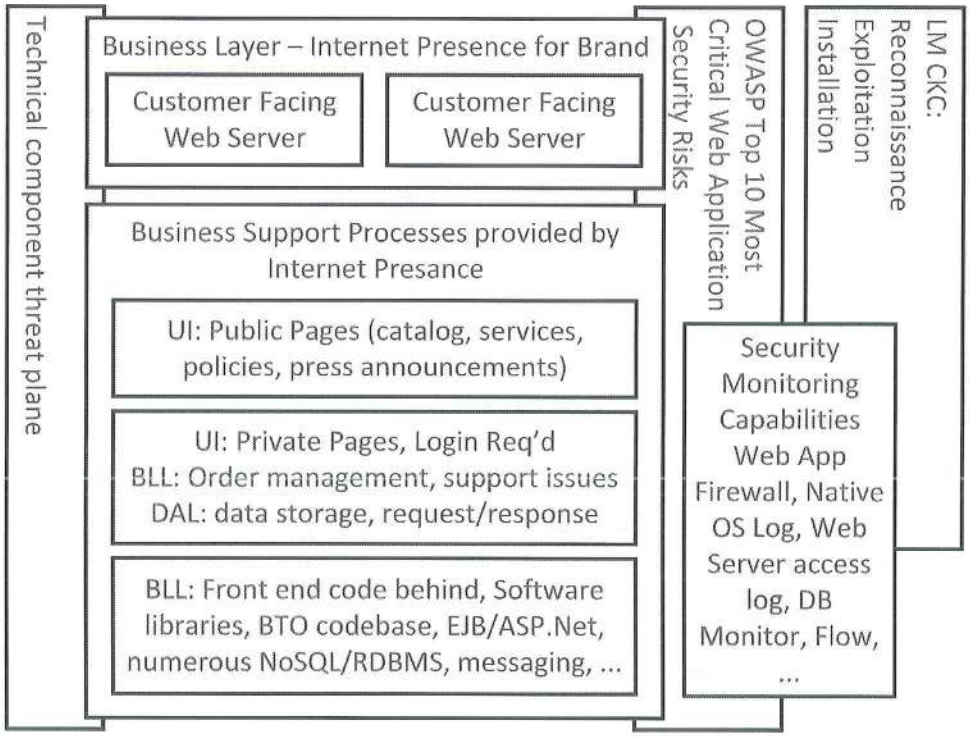


Figure 2 Web Presence Attack Components and Attack Surface

Example: End User Payload Focused Attack

Today, the most likely avenue of attack is against the end user through some mechanism that entices them to visit a site, download a file, or click a link in an email and then ignore security warnings. Collectively, these attacks all fall under the umbrella term "social engineering" which are leveraged through email, email attachments, watering hole attacks, and manipulating text stored in forum posts.

At the top. the attacker exercises the cyber kill chain to deliver some content by some means that interacts with a user application. In the case of a malicious email, the user may be enticed to open an attachment that could be a malicious PDF file or an office application with a macro that has malicious script code in it. Once the payload is opened, or successfully delivered *and the user interacts with the payload*, the attacker can begin to leverage a wide variety of techniques found in the MITRE ATT&CK matrix.

The elements illustrated in the next figure along with references to the Cyber Kill Chain and the MITRE ATT&CK Framework.

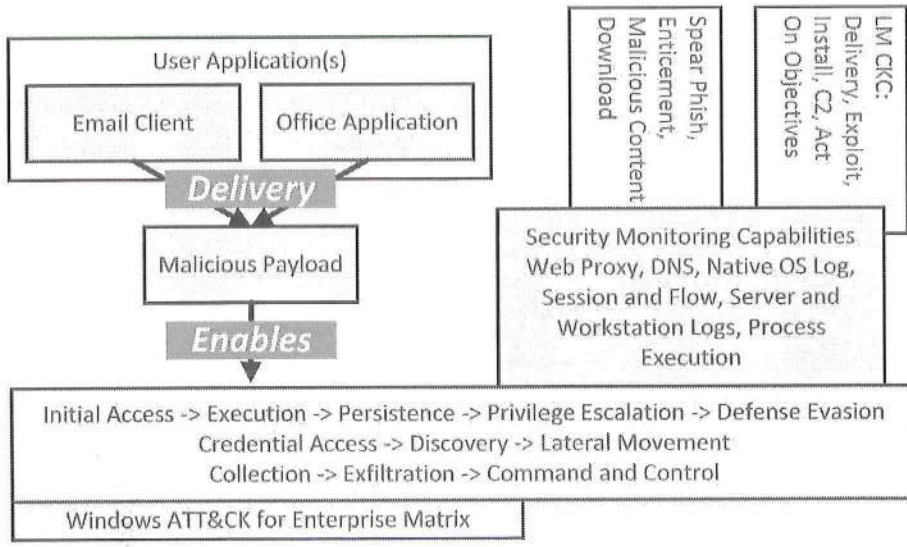


Figure 3 Example: End User Payload Focused Attack

One of the first actions is to establish persistence so that the attacker can return. After that, attackers vary their actions. They may attempt to crack passwords, gather hashes from the SAM database or from an inbound authentication token on a connection, search the network for an opportunity, or if they are lucky, pillage the system they compromised because it was just the right one. In any event, an attacker has numerous opportunities that can be used to take advantage of Windows and Microsoft networks. The older the operating system and the less out of date it is where they establish the first foothold, the better (for them, not the blue team).

Organizational Considerations for Use Case Development

Before we dig into dozens of data source specific technical use cases, security operations really need to understand several key factors and answer several questions. This chapter will lay out several models of the modern attack landscape so that security operations can determine where to engage, what logging sources will protect the business, and how to instrument data collection.

Questions to Answer:

1. Is there a security minded business analyst available to help develop security focused use cases?
2. What is the business operating environment?
3. What is the application and data resources the resources the business depends on to achieve its mission objectives?

4. Is there logging enabled for the applications and the platforms they depend on?
5. Who is the application and platform owners and custodians SOC will need to engage with?
6. What will the SOC do in response to an alarm?
7. How will you maintain your use case library?
8. At what point does a use case produce change control items?

"Top Ten" Security Operations Use Cases

This section is here to provide a starting place for your SIEM and SOC efforts. When formulating your organizations top ten use cases, consult standards like the Australian Signals Directorate Strategies to Mitigate Cyber Security Incidents[24]. The ASDSMCSI is designed to assist organizations to help security professionals to mitigate incidents. There are various other sources that can be consulted. Below is a list of the top ten security use cases that a SOC team should implement as early as possible.

1. Privileged Entity Monitoring.
2. Brute force Authentication failures.
3. Authentication Anomalies
    a. Service Accounts used for interactive logon.
    b. Service Accounts used from non-authorized source systems.
    c. User logon locally (on LAN) within a short window of a VPN logon.
    d. User logon more than an hour before or after normal work periods.
    e. Interactive User authentication from multiple source systems.
    f. Shared account usage (which should not be confused with accepting the idea that using shared accounts is acceptable).
    g. Default account usage (same caution).
4. Session Anomalies. There are numerous examples in this area.
    a. The typical end user should have a session beginning and ending with ten (or less) hours from each other.
    b. Significant profile change in web browsing habits.
    c. Spike in outbound firewall denies.
    d. Workstation network to workstation network communication.
    e. What is a reasonable clipping level for sessions?
5. Account Anomalies
    a. Accounts used before the user's start date.
    b. Accounts used after the user's end date.
6. Data Exfiltration indicators

[24] https://acsc.gov.au/infosec/mitigationstrategies.htm (8/18/2018)

   a. HTTP(S) Send/Receive mismatch. Data received from a site is often many times data sent to a site, by byte volume, as most of the time the browser is downloading a file and rendering it for the user.
   b. File transfer protocol(s) used from end user sources or systems that don't require these services such as outbound FTP from a print server.
   c. Use of file storage sites (Dropbox, Box, Microsoft OneDrive, GoogleDrive, SugarSync, Leapfile, etc).
   d. Use of websites that allow for 'easy information sharing or text storage', where users can cut/paste information in an unregulated manner.
7. Signature Matches to known Vulnerability Scan Results.
8. Any excessive 'service failures', such as A/V agents that repeatedly fail or backups that fail. Note that outage detection also provides operational value as well as security value.
9. Insider Threat Indications -
   a. Accessing "security research" sites.
   b. Use of USB drives.
   c. Authentication baseline violations.
   d. Authentication failures against file shares, applications, servers, internal SharePoint sites, etc.
10. Security Log Data failure conditions.

AntiSpam and Email Messaging

If there is one system people use *every day,* it must be their email system. Typical email is person to person or person to a small group, with a low ratio of email with attachments to those without. This is a good starting place to define "normal" for your organization.

Use cases based on email are easier to implement for locally hosted email systems than cloud email systems, because it is easier to get logs from on-premises systems. Free email systems are highly unlikely to provide meaningful data export and SIEM integration. Paid cloud email systems may not support SIEM integration or may only provide user activity through a downloadable report.

1. **Email with attachments that have spaces or multiple periods**: Attackers like to obscure their malicious content. Two methods are to add several spaces and include multiple extensions like ".docx.exe".
2. **Email burst or flood**: A rash of inbound email can easily be a phish, or some other campaign you may not want.

3. **Infected sent mail**: Internal users *sending* virus-laden messages internally or outbound. This condition indicates a failure in the local A/V client or malicious software on the system.
4. **Spam sent mail**: Users who sent email that is identified as spam is not normal. If the organization has an upline anti-spam system, the user's messages were likely blocked. It may be advisable to let the user know, *assuming* there isn't a further negative condition.
5. **Non-authorized systems sending mail**: The only systems that should be communicating to any one of the messaging TCP ports should be well known and understood (such as the internal messaging systems). Below are common messaging ports for email systems. For continuous monitoring, the system should create an alarm for traffic *outbound* on these ports from a *non-authorized source*. For Threat Hunting, a report that includes these ports and the senders should be developed and periodically reviewed.
    a. 25/TCP - Simple Message Transfer Protocol (SMTP)
    b. 110/TCP – Post Office Protocol (POP version 3). POP2 is on 109; but the likelihood of seeing this is minimal.
    c. 143/TCP - Internet Message Access Protocol (IMAP)
    d. 209/TCP & UDP - Quick Mail Transfer Protocol (QMTP)
    e. 220/TCP & UDP - Internet Message Access Protocol (IMAP), V 3
    f. 465/TCP - Authenticated SMTP over TLS/SSL (SMTPS)
    g. 587/TCP - e-mail message submission (SMTP)
    h. 993/TCP - Internet Message Access Protocol over TLS/SSL (IMAPS) – Apple systems use this port.
    i. 995/TCP - Post Office Protocol 3 over TLS/SSL (POP3S) – Apple systems use this port.
6. **Messaging IoCs**: Intelligence sources do list known email addresses as an IoC. Messages to and from these email addresses are suspicious. You cannot necessarily prevent a malicious user from *sending email inbound, but* you can monitor and communicate to a user who received it. Any traffic to an email address identified based on a feed from a threat intelligence source should definitely be investigational, whether it was blocked or not.
7. **Significant volume changes:** From a threat hunting perspective, the team should look for volume-based changes, such as a user who rarely sends attachments suddenly sends a large number of attachments to a competitor may indicate intellectual property theft or industrial espionage.
8. **Autoforwarding:** Users who send large amounts of data to their home email addresses may expose the organization to an unacceptable risk.
9. **Email with competitors:** Most, but not all, organizations do not routinely send a large portion of email with a competitor. This pattern is also subjective, but it may reveal an insider threat.

10. **Users generating numerous Non-Delivery Reports:** This condition may indicate their account is being used to probe for valid email addresses at a particular domain or some operational issue.
11. **Constant email transmission**: Users sending email every hour of the day, which may indicate something on their system is attempting to use an email capability for covert communications.

Email and Web: Interactions with Look a Like or Doppelganger Domains

Phishing scams sometimes use domains that look very close to your domain and are changed by single letter replacement, fuzzing, omitting a character that the mind will naturally fill in, or transposing two letters. For example, blueteamhandbook.com can easily be changed to b1ueteamhandbook.com, bluetaemhandbook.com, and numerous other variations that look like the real domain name, with the goal of attacking at least one book author. Also, domains can be registered with Unicode characters to support foreign languages and lead to a homograph attack. Most browsers attempt to mitigate the exposure – but nothing is perfect.

You can either develop an inventory yourself, or use a tool like "dnstwist" by Marcin Ulikowski[25]. This tool is written in Python, and does have some dependencies required for it to work properly on the command line.

The goal in using a tool like dnstwist is to generate likely look-a-like or domains that a phishing campaign is likely to use against your organization. The dnstwist tool can also be used to check and see if a twisted domain is registered.

DNSTwist Domain Enabled Use Cases:

1. **Investigate Domain Name:** Pull out the domain name portion of email sent to/from the organization and compare it to a twist domain. If you have a match, then drop the email. Even better, do not accept email from a twist domain.
2. **Check twist against browsing:** Pull the FQDN portion of a domain name from web server logs, and alarm if a user visits a twisted domain name.
3. **Alarm on DNS Lookups:** DNS lookups against twisted domains should also generate an alarm. Truth be told, DNS blackholing twisted domains is actually a solid protective measure, so that if a user attempts to visit a suspect site it will be directed to 127.0.0.1 (there is no place like home).

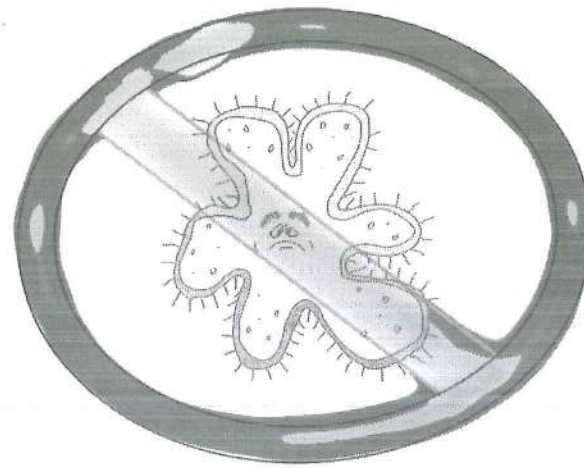
[25] https://github.com/elceef/dnstwist

Implementing this type of a DNS sinkhole not only protects your users, it also allows for an alarm condition.

Antivirus (A/V) Systems

Antivirus systems are bread and butter security component, even though traditional A/V has a tough time keeping up in the digital arms race. Nonetheless, maintaining a desktop security suite that includes A/V and other HIPS components should be part of your defense in depth architecture. There are several conditions should be monitored.

1. **A/V detection**: Finding and cleaning a piece of malware on a system, a USB drive, a CD, an email attachment, download from the web, etc. should be investigated. Just because the A/V system did its job doesn't mean that the PC is "clean." The file name provides significant clues to the issue. For example, if the A/V system removes a file with "drop" in the name it may mean that a "stage two" tool successfully got onto the system. The immediate web, email, and USB device history right before malware observance should be checked. A ".scr" file is most often a screen saver, which are often malicious. Realize that a screensaver, by its very nature, captures the username and password. *Remember, even though the A/V protected the user the malicious file got on the system somehow*, so work to answer that question and examining quarantined files in context.



2. **A/V Potentially Unwanted Program** (PUP) detection. The names of executables that are classified as PUP's may be early warning indicators, may reveal insider threats, uncover unauthorized software installation, or may provide a clue that an attacker tried one tool and then moved onto another when the first tool failed.
3. **Password Dumpers**: In particular, if a password dumper was seen and subsequently quarantined means that someone tried to extract the hashes from the SAM database, extract passwords from memory on a running system, or gather passwords through some form of network extraction.

Example names are PWDUMP#, cain.exe, wce.exe, mimikatz.exe, mimidogs.exe, and gsecdump.exe.

4. **A/V *agent status and update failure* conditions**. A/V agents should run successfully, should receive their updates, and not crash. Example events can range from an inability to scan a drive, inability to quarantine a file, an agent that will not start, the agent starts and crashes, it cannot retrieve updates, or is shutdown unexpectedly. A truly advanced use case can be developed around the number of active systems in a Windows domain, end user authentication from a given machine, and successful updates to A/V. Here, the use case would focus on detecting that an end user machine was "recently updated" as users authenticate (by name, not IP – DHCP and portable computing would make IP addresses less reliable). If not, then you could have a machine out of compliance.
5. **Repeat A/V offenders or reinfections**: Users who routinely get an infection notice need more attention. This use case is longitudinal in nature, meaning that you want to know if a machine is being re-infected *over time, say 14 to 60* days. You can further differentiate this by end user logged on and virus name or family. Same user means the user is causing higher risk, different users may mean a kiosk or loaner is the victim PC, and the wider variety of malware means the greater likelihood that the compromise is more severe. Here, company policy should address what happens when users cause repeat security threats to the enterprise. If a user keeps getting the same type of infection (virus name or class), investigate likely causes: USB drive insertion, particular websites, or maybe even a particular network share.
6. **Multiple Infections in a short time frame**: A host that has multiple *different infections based on the binary and/or the virus name* may warrant more rapid attention than an asset with a single virus.
7. **Escalation based on Asset Value**: Assuming that your SOC team has valid intelligence on the value of an asset, an infection on a "critical" asset such as one that falls under the Payment Card Industry Data Security Standard (PCI DSS) warrants more rapid attention than an asset with a "medium" value.
8. **Anti-Virus Infection notifications in close proximity to spyware, malicious site, email attachment file open events**: The A/V system may be the indicator that gets your attention. Here is where an EDR application can be very informative and assist in a rapid MTTD. An analyst should be able to locate the file name and then determine how the user triggered the malware. Common methods are opening an attachment, inserting an infected USB drive, or visiting an infected website. If the source is email, then make an immediate search for other users who received the same email by subject line and sender, then seriously consider forcibly removing this email. In the case of a particular site, block access.

9. **A user with Elevated Access logging into an infected system**: This use case requires that you maintain an inventory of users with elevated access, or that all of these users have a particular naming convention so elevated access accounts can be more readily detected. If they login to an infected machine, then at a minimum an automated notification advising them to change their password is in order.

Windows Defender has its own set of Event ID's. In many cases, Windows Defender also includes a suspicious file name, a unique threat ID, a severity rating, file path, error codes, and other useful details. These details are representative attributes to use when building dashboards, alarms, and reports.

Table 8 Windows Defender Application and Services Logs\Microsoft\Windows\Windows Defender\Operational and System Log

Event ID	Name
1000	An antimalware scan started
1001	An antimalware scan finished
1002	Scan stopped (canceled) before finished
1005	Scan terminated due to error
1006	Detected Malware
1008	Action on Malware Failed
1010	Antimalware could not restore an item from quarantine.
1115, 1116	Malware detection
1117	Malware remediation or action taken
1119	Remediation error
2001	Failed to update signatures
2003	Failed to update engine
2004	Reverting to last known good signatures
3002	Real time protection failed

Application Whitelisting

The security posture of the endpoint is more important than ever before. These systems are designed to monitor executables running on a system when running in detection mode. When running in prevention mode, they will stop running an executable that doesn't match an approved policy. Application whitelisting can identify several adverse conditions. Further, since this capability records end user activity, it can be very useful in an employee investigation case.

1. **Unauthorized Installation**: Recording a setup or install process should provide the identity of the user installing the software if it set up services, and the directories where the software installed. The installation process

can be checked against an authorized change in the change management system to determine if it was authorized. Also, application whitelisting can help detect if the system's integrity was violated.

2. **Unauthorized drivers**: When a user inserts a USB device into a system, the OS will respond to that notification event and attempt to install the appropriate driver software. Removable storage devices can be an avenue for data exfiltration or can provide an avenue for malicious software entering the environment.
3. **First Observed Binary**: Introduction of a new binary in the environment may indicate an adverse condition. A user running several new binaries on their system may also be of note. An application whitelisting suite may not cleanly detect this specific item, unlike an EDR platform. Alternately sysmon can be deployed because it will generate a file hash as each process is invoked. Note that first observed binary analysis needs to be done by file hash, not file name because while the name may change, the hash will not. You may need to instrument some other method, such as manual review to realize this condition.

Windows has two facilities of note: AppLocker for Windows 7 and above, and Software Restriction Policies for Windows Vista and below. Both technologies record control binary usage. AppLocker events in the AppLocker event log and can be enabled using group policy. There are at least sixteen different events recorded. EventID's 8020 to 8027 are focused on package deployment issues, so they are not listed here.

Table 9 Windows AppLocker: Application and Services Logs\ Microsoft\ Windows\ AppLocker

Event ID	Level	Name
8000	Error	Application Identity Policy conversion failed. This condition indicates issues applying policy to the system.
8002	Information	FileName was allowed to run.
8003 Audit Only.	Warning	FileName was allowed to run but would have been prevented if policy enforced. (EXE's).
8004	Error	FileName was not allowed to run.
8005	Information	FileName was allowed to run.
8006 Audit Only.	Error	FileName was allowed to run but would have been prevented if policy enforced. (Script/MSI's).
8007	Error	FileName was not allowed to run (by policy).

Command and Control

There are several methods to detect command and control, aside from using an IoC list, an IDS rule, or a domain block list. Rather than duplicate information here, please refer to the Threat Hunting chapter for a discussion on command and control as described on page 178.

Data Loss Prevention (DLP)

DLP systems come in two broad types. Data in Motion systems are deployed to monitor traffic as it moves through a system and the network. Data movement can be email, FTP, copied to a USB or CD, saved off to cloud storage, copied to a network share, and therefore DLP needs to be integrated into an OS level service. For example, DLP software that analyzes email is plugged in to the messaging pipeline as a Message Transfer Agent (MTA). Data at Rest systems (or agents) find files of interest-based searching, such as a a file share or a web repository like SharePoint.

Once alerts from a DLP system reach a certain threshold they may need to be investigated by the proper *internal team*. Realize that in some organizations, DLP events can be quite normal. For example, sensitive patient data is routinely handled at a hospital or an insurance company. If a user emails ten spreadsheets and the DLP system intercepts them and encrypts them for delivery to the recipient, the user may be aware this is normal and wanted that action to occur because they are trusting the DLP system.

Before going too far down the road with DLP, the SOC team will need to determine if there is a more applicable internal team who are a better business fit than the SOC (there should be...). For example, an insurance company likely has a Member Privacy team, or HR may perform investigations using the DLP system. One argument in favor is that by sending in alarm data to the SIEM, an end user activity report will have a more complete picture to present during an employee investigation. Further, since the DLP system identifies potential IP loss, an analyst can incorporate these alerts to gain a better picture of end user activity and notify the appropriate internal team if warranted.

Regardless, in motion DLP systems identify data exfiltration, whether intended or not. At rest DLP systems identify where valuable data resides. Today, attackers are interested in the data, because that's where the money is.

Domain Name Services (DNS)

Gathering DNS data presents a few data collection and data reduction challenges that you will need to work through. DNS detection requires detecting

name queries that are outside of the norm *and being able to detect the true source IP address if at all possible.* One issue that will prove to be difficult is a lack of internal reverse DNS lookups and stale DNS entries. If you can't reliably lookup an IP to a name, there will be a small impact on alarm processing time. The situation can be a bit worse when an IP comes back to multiple systems.

Collecting DNS: Collecting DNS from a DNS server can be problematic. For example, Windows DNS requires that you enable "debug logging", and then fully parse that data through a either a local or remote file reader process. Another problem with DNS is that most (90%+) of the traffic on the network are local queries. Local queries are normal. When considering how to collect DNS, focus on collecting internal to external queries, find where those queries are resolved, and collect data at that point using network extraction as the collection method. If there is a mirror port available at the perimeter, DNS query and responses will be logged *from the internal DNS server(s)*, as they are forwarding queries on behalf of the end user. If you collect DNS traffic via a mirror port on the same switch as the DNS server, you will collect a significant amount of normal query traffic for the internal network that will have low to no value for identifying attackers. There are at least 30 defined record types available for use, with the more common being A, CNAME, PTR, SPF, AAAA, NS, and MX. TXT records are seen, but in low volume. There are at least two well-known tools to collect DNS: PassiveDNS and Bro IDS.

DNS Monitoring Use Cases and Detection Patterns:

1. **Young (< 7d old) or recently registered domains (and thus, websites):** Malware is increasingly using sophisticated DNS lookups and query types to *signal* their command and control network. Attackers, and in particular Phishers, are using recently registered domains as spreader points. Techniques vary in exactly how recently created domains are used for an attack. Domains that are less than a week old are more likely to host malware than established domains. If the "Created on" or "Creation Date" field from a whois lookup is less than seven days, look very closely at the domain registration details. As an example, on 11/05/17, a check of domainpunch.com found 85,794 dot-com domains registered on the prior day. There are also several sites that provide lists of newly registered or expired domains, every day, usually for a charge. Examples include whoxy.com, whoisxmlapi.com, domainlists.io, domains-index.com, etc.
2. **Names not in the Top 1 Million List:** As described on page 112.
3. **Long, misshapen, or weird second level domain names**: Most second level names should be less than 24 characters. DNS names have a maximum of 255 characters in total. In practice, some analysis should be performed on DNS names that are 72 characters or longer. Really long names (>128

characters total) and continued query/response is most likely DNS tunneling or a DGA. You will need to establish these two thresholds for your environment.

4. **Hexadecimal Domain Names**: Domain names should be readable by people; after all, they are designed to help people locate resources. Hex is not usually human-readable[26]. Malware uses Hex values as beacons, may have Base32 encoded commands disguised as a name component, and usually require specific query and answer resource records set to specific values. Examples include FrameworkPOS, FeederBot, Morto, etc. Based32 encoding is used because the characters in a DNS name are effectively limited to 37 possible unique characters.
5. **TXT Records/Lookups**: DNS can provide freeform lookup information from a domain. Historically, the most common uses for TXT records are to help validate email delivery with Sender Policy Framework (SPF). Other normal uses are DomainKeys (DK) and DomainKeys Identified E-mail (DKIM). Query/response outside of these purposes is not normal, and further, illegible data in a TXT query or response is suspect. In contrast to names, the data returned from a TXT response can be Base64 encoded.
6. **SRV Records**: Server Resource Records are used to define a network location for a server that provides a specific service. They are actively queried by internal Windows systems within AD for many resource types. From the Internet, they are commonly used for communication-oriented services like SIP, email, some games, Session Traversal for NAT (which, in turn, support real-time audio/video/messaging), among other services. Again, you would want to establish a "normal" baseline and then be advised of "new" services queried. Also, a high volume of different queries to a particular DNS site where the request/response types are *not the same* type of lookups would not be normal.
7. **Private IP addresses returned**: Name server queries to Internet sites should rarely return private (RFC 1918) IP addresses. NetGear's "routerlogin.com" is one of the few examples of a private IP returned from your local DNS.
8. **TXT without A Records**: A direct query for a TXT record without a preceding A record lookup is not normal. Further, domain names that don't have A records that support their TXT and SRV records is also not normal.
9. **Long TXT record queries**: Assuming that you can monitor for query types, excessive queries or long queries returned from an Internet server may be used for command and control. Look for Base64 encoded data. TXT records are used for SPF, so they do occur. Tools known to use TXT records include dns2tcp or DNScapy.

---

[26] Note, though that you may see DE:AD:BE:EF:CA:FE on the network. And there are a few humans who can natively read hexadecimal network traffic.

10. **Look-a-Like or fuzzed domains:** Review the section Email and Web: Interactions with Look a Like or Doppelganger Domains on page 73 when working through DNS use case development.
11. **DNS queries *not from* authorized servers**: An enterprise should only have a small number of internal DNS servers that can forward queries to servers on the Internet. *Any* DNS query outside of this boundary should be investigated, if for no other reason that ensuring the sender is properly configured in order to provide operational assurance.
12. **Volume and volume profile changes**: Establish a baseline profile for DNS traffic. These indicators can become alarm conditions once baselines are established. Examples are:
    a. Average queries per hour during working hours/off hours.
    b. First time use domain queries (new domain name seen).
    c. Volume of SRV RR, TXT, and MX queries.
    d. Internal failures – lookup for domain fails.
13. **Name analysis**: High volume queries with hostnames that are random for the same 2nd level domain *and* the same length indicate a DNS tunneling tool is sending data to the attacker's site, because the DNS server is consuming the host name as encoded data.
14. **Foreign countries**: You should study your organization's communication and operating model to determine how much communication occurs to countries outside of your own country. For example, a University with a varied foreign student population would consider this normal, but an insurance company that operates in a few states in the US would consider several queries to foreign countries abnormal. Note that if you are reading this book and you are in a foreign country, queries to name servers in the US and several European countries may be very common and may make this analysis more difficult.
15. **Queries to Dynamic DNS providers**: There are several dozen dynamic DNS providers operating today[27] who provide nearly free or inexpensive name to IP DNS resolution. A common model is for a home user to register their IP address and allow certain services through, with a name unique to them that their ISP would not provide. For example, a VPN client. Attackers can easily use these services as an avenue for hosting malicious services such as C2 DNS service because DDNS providers allow for rapid changes of a name to an IP address and can be used at nearly no cost.
16. **Abused Top Level Domains (TLD's)**: Spamhaus maintains an ever changing, evidence-based inventory of the top ten most abused domains names[28], which is expresses as an aptly named "badness index". Integrating this

---

[27] Lists include : http://dnslookup.me/dynamic-dns/, GitHub: Nate Guagenti / neu5ron, and http://mirror1.malwaredomains.com/files/dynamic_dns.txt (3/26/18)

[28] The interactive list is available here: https://www.spamhaus.org/statistics/tlds/ (8/18/18)

functionally into the SIEM may not be practical, but integrating a check of the domain TLD into the incident response process and the analyst checklist certainly is. As of June 28, 2018, there are 1,503 TLD's.

17. **Traffic to external IP without DNS query**: Direct HTTP, HTTPS, FTP, SSH, and likely other protocols directly to an IP address is suspicious. It is not common for an end user to type in https://#.#.#.#/. With whatever method you have, review which end systems are communicating outbound directly to an IP without a name. A caution: a reverse lookup *could* be performed, with some risk of alerting the site owner that you are trying to get a name for an IP. It is best to use an intermediary, like a call to any site that offers a NSlookup function. (Root DNS servers don't count!)
18. **Use of non-authorized DNS**: There are several free DNS services available on the Internet other than the DNS that the sites ISP provides. Queries to these DNS servers, such as Google's at 8.8.8.8 and 8.8.4.4, *may* indicate a condition that needs resolution.

End Point Detection and Response

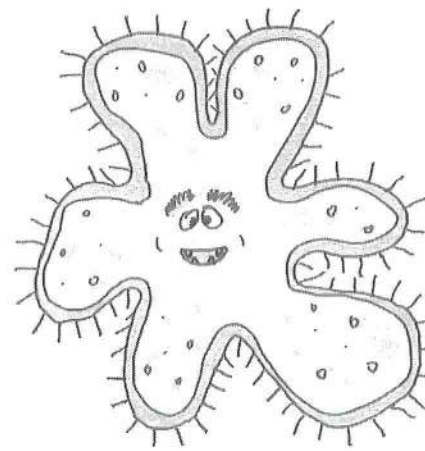
Entering the desktop protection field are highly capable software platforms focused on threat hunting for the endpoint. Vendors include FireEye Endpoint Security, Carbon Black Cb Response, Guidance Software EnCase Endpoint Security, Cybereason Total Enterprise Protection, Tanium, CrowdStrike Falcon Insight, and CounterTack Endpoint Threat. The Gartner 2018 magic quadrant for this space lists more than twenty vendors.

Here, alerts from an endpoint protection system can have a high "signal to noise" ratio because those alerts are pre-validated. These events occur because a binary matched an entry on a known feed list or matched a detection criterion. By consuming and analyzing endpoint protection system alerts *only (and not all endpoint activity)*, you can actually *perform threat intelligence and assessment* on the user population. If users are predominantly becoming infected and then bringing their notebook back into work, or if users insert USB drives in their computers and there is infectionware on the USB, then SOC has a better-defined path for security awareness training and remediation.

Microsoft's EMET is a free tool from Microsoft. It can perform a similar detection and mitigation, but it logs locally and doesn't have a comprehensive console like the vendors listed above. According to the EMET 5.5.1 User Guide, EMET reports to the local event log, so if there isn't a method of consuming that log then these events would stay on the system. The application whitelisting policies, which are also configured with Group Policy are excellent candidates for forwarding to the SIEM.

In order to get data into the SIEM platform for SaaS EDR, some form of encrypted syslog service or some sort of REST API needs to be configured to send data into the SIEM. *Do not send all endpoint data that the EDR platform collects to your SIEM.* Rather, send the "condition detection" data to the SIEM.

End Point Detection Use Cases:



1. **IoC hit:** IoC hits when the EDR system detects a connection to a suspicious or nefarious IP address or domain name, or a file that matches a known bad by hash value.
2. **Binary first observed:** A "first occurrence" of a binary, never seen before in the environment, *once baselining is done* can detect unauthorized software installs, malicious software, unauthorized downloads, or software executing from removable media.
3. **Hash Checks:** A locally configured alerting list, such as a hash value of a particular binary. These don't have to be malicious. For example, you could have a deception system or practice in play to detect if a user opens or copies a "Top Secret look alike" file (this is an example of a HoneyToken).
4. **ASEP Registry Key:** Modification of a specific registry key used to establish persistence, such as the Run, Run Once, or RunOnceEx.
5. **Printer:** Also, the printer key can be configured to load an arbitrary DLL: (HKLM\SYSTEM\CurrentControlSet\Control\Print\Monitors).
6. **Specific directories:** Modification of directory or file within a file system.

Data Islands or System Snowflakes

For Windows, there are at least four security contexts that need to be considered for Microsoft Windows systems. Domain, domain member server, domain member workstation, and standalone workgroup systems. Each of these systems need to provide data for the SIEM. For Linux, most systems are their own source data island, which usually provide data via syslog. For an application its own localized and application specific audit system is yet another data island. those systems represent a challenge – little to no standards, they may not even generate enough logs, and they will require greater customization. However, this last group contains the item the attackers want most – your Crown Jewels.

Windows Account Life Cycle Events (ALCE)

These events record new accounts, account modifications, deletions, disable, and enable changes. These events all have discrete event IDs, record the account that was changed, and the user who made the change. Account management should be done by a specific account management team with a supporting request and authorization process. Also, realize that *any Windows system* other than a domain controller can have local accounts defined and therefore used. This is a common requirement for most IT General Controls programs (ITGC).

Table 10 Security Log: Account Management Events

Event ID	Name
4720	A user account was created.
4722	A user account was enabled.
4723	An attempt was made to change an account's password. **
4724	An attempt was made to reset an accounts password. **
4725	A user account was disabled.
4726	A user account was deleted.
4738	A user account was changed.
4781	The name of an account was changed.

** These events should be monitored differently.

Account Lifecycle use cases:

1. **Short cycle account create and account delete events:** This use case catches accounts that are created and removed within a very short time window. As a bonus, the severity would be raised if the account was used, such as a logon event between the create and delete event.
2. **Short Cycle elevated group add and group remove events:** This use catches accounts added to highly privileged groups like "Domain Admins" and then quickly removed from the group. Extensive damage can be done in a short time.
3. **Accounts created/modified/disabled by staff *other than* designated account managers:** This condition helps identify policy violation, rogue admins, or attackers who gain access to a domain admin level credential. In more mature organizations there will be a few IT that manage user accounts and group changes. When this situation exists and others manage accounts there may be a policy violation, a social engineering event, or true maliciousness.
4. **Accounts managed by users *other than* the designated *service account:*** A service account is used by a mediated access application such as NetIQ DRA,

SailPoint, PeopleSoft AD integration, and CyberArk. Mediated access applications handle account and group life cycle events by processing a request through a workflow system and using a specific account to implement the change on a domain and/or member system. This alarm assumes that one service account used by a specific application is used to manage accounts and any ALCE outside of that realm is unauthorized.

5. **Accounts that do not follow an established naming convention:** Detection can be accomplished through regular expression pattern matching or account length checking. In the weakest case, simple account name length checks may work, or a daily human review of accounts created, enabled, disabled, or removed from the network and the AD forest. In the more sophisticated case, an organization will have a naming convention that supports a regular expression check to determine if accounts follow a pattern such as AA#######, SVC_*, U########, or DA######t. Note that this should also be generalized to workstation additions to the domain, where only workstations that follow a naming convention are allowed to be added.
6. **Account deletion:** Some environments may choose not to delete account. Rather, they change the password to something very complex, remove *all groups (including Domain Users),* and disable the account permanently. This method allows for Security IDs in the NTFS file system to resolve to an account name, but effectively prevents account usage. Also, this method allows for a re-enabled account for an account held by a user who terminated can be more easily identified when these accounts are added to a tracking list.
7. **Accounts created and disabled or deleted this week for new users and users terminate employment:** This use case can be satisfied with a daily and a weekly report for all systems where user accounts are created, like an active directory domain or a constituent application.
8. **Accounts used prior to the authorized use date**: It is common for organizations to create accounts ahead of a new users' arrival. These accounts should not be used before the user arrives. This use case implies that the organization has a method of connecting account creation, account names, and the relevant dates in order to conduct this analysis.
9. **Accounts created outside of the domain context**: There are two cases here. One case is when accounts are created on a workstation – this should be a very rare occurrence. The other use case is when accounts created on member servers. Local accounts may be needed for a specific application. These use cases look for ALCE's *not* from the set of domain controllers.
10. **Observed default accounts/credentials**: Default accounts should not be observed, because use of a default is inherently not attributable to a person, and a default account starts its life with a default password. There

are several websites with lists of default accounts and passwords[29]. Default account usage has been in the OWASP Top 10 for several years.

11. **Local account creation and elevated access**. One of the key tenants in an Active Directory domain is *centralized authentication.* In reality, practices vary widely when it comes to local accounts. For example, an organization may grant administrative access to a workstation by creating a local account for the workstation user to accomplish an administrative task, or a domain level account and then add that account to the local administrators' group. The SOC should understand how elevated access is applied and be able to detect elevated account usage.

Advanced Monitoring Rules and Alerts Which May Require External Scripting

1. **Accounts created in a *constituent system* which do not match an account in the *primary directory*.** This type of rule requires that systems managed with local accounts need to be cross indexed with the primary directory. In order to have this rule function, some sort of lookup in AD would be required and a constituent system will need to push an ALCE events to the SIEM. This use case will mature into having an artificial identifier added to all constituent systems and the directory itself, such as an employee ID so that each account can be attributed to a single account holder.
2. **Post ALCE events to an account tracking database**. For example, an Identity Management system which has more metadata about an account than a directory holds. Since the SIEM gets the actual event as they occur, it would be a better use of system resources to push that event to the IdM rather than have the IdM deploy yet another monitoring agent to the domain controller.
3. **One over One[30] manager notification on creation, modification, or disable**. This notification would normally come from an IdM, because while AD user accounts do have a manager field, that field may not be populated. If and IdM is not in place, *and the manager attribute is populated at the time of account creation in the directory*, then a notification would allow for the user's manager to know when the account was actually created.

Windows Group Life Cycle Events

Windows has two group types, security and distribution, and four group scopes local to a system, universal, global, and domain. The scope defines how the

---

[29] For example, the aptly named www.defaultpassword.com, or defaultpasswords.in.

[30] If you haven't heard of this term, it means that each person's direct supervisor, the one who is responsible for annual review and pay action is informed. It does not mean a dotted line relationship.

group is used in an AD forest, while the type defines the intended usage. This model translates into dozens of event IDs that need to be monitored. In practice AD groups are often used to control access to a resource that must be monitored, such as a directory where financially significant data is stored in a publicly traded company[31]. Further, there are several groups within Active Directory that provide elevated access. Even worse than that, groups can be *nested.* For example, an organization may have an "Administrative Service Accounts" group embedded within the Domain Admins group. If you are only monitoring the Domain Admins group, you will miss a user being added to or removed from a nested group which has Domain Administrative privileges. When monitoring group changes with a SIEM, a subset of changes may prompt a notification. There are many other means to support the intent of a control, such as creating a report for group membership through scripting out a report or using a purpose-built application.

AGDLP is an abbreviation for "Account, Global, Domain Local, Permission that summarizes the recommended method by Microsoft to provide Role Based Access Control (RBAC) with any resource that can leverage Windows authentication and Active Directory. User accounts should be members of Global groups, which are then assigned to Domain Local groups. The DL group should describe the access permission and be applied to the specific resource. Depending on where the resource group is in the forest and the group's scope within the forest, a different security event is created on a domain controller within the forest. Given that there are so many event IDs from the same source, the IDs are summarized below in a table rather than listing them out once per line.

Table 11 Windows Events: Group Changes (Security Log) (V1.02)

	Security			Distribution		
	Local	**Global**	**Universal**	**Local**	**Global**	**Universal**
Created	4731	4727	4754	4744	4749	4759
Changed	4735	4737	4755	4745	4750	4760
Deleted	4734	4730	4758	4748	4753	4763
Member Added	4732	4728	4756	4746	4751	4761
Member Removed	4733	4729	4757	4747	4752	47620

[31] In the United States, this requirement is derived from Sarbanes Oxley controls.

Group Based Monitoring Alerts

NTFS access is normally managed by applying a domain local group to the resource on the directory itself. Also, a group can be used at share level itself to apply permission. If you compare permissions to your front door, share permissions are like a screen door and NTFS permissions are like a bolted security door.

Changes to a select set of NTFS and application control groups may be in order. When you create a SOC alarm or an email notification for a resource owner *make sure that* the message explains what the group controls access to - meaning the resource Itself or the application right managed by the group. Don't automate a notification that a user was added to "NTFS_g45_Direc_RW". Instead, find out the purpose and path of the directory and us that instead. for example, "Shared Drive, Monthly Financial Summaries, path name \\Storage\NY\FinRep_Monthly" for the monthly financial performance reports for the NY business unit.

Applications can also use Active Directory groups to control access by mapping an AD group to an internal role within the application. Following the same example, a notification that states a user was added to "PeopleSoft Production Admin Members" is better than "AppCtlFinPplAdminsPRD".

Special Group Changes

Windows 2008 introduced a new monitoring capability called "Special Groups" which is used to record when someone who is in a set of defined groups logs into the network. The event ID is 4964. In effect, this event ID records when accounts that are members of any *administrator defined* group logs in. This function can be used to monitor service accounts, members of elevated access groups, users who have put in their notice, or any other security focused valid reason. In order to implement this, assign the Security Group ID field to the registry key HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\Audit as a semi colon delimited list. The default administrative groups should be added to this key, such as Domain Admins (S-1-5-21domain-512) and the local Administrators SID (S-1-5-32-544). For more information, see Microsoft article 947223. If you couple this feature with Windows Event forwarding on the workstations, then there is an audit record when *any user who is a member of a group that provides some form of elevated access* logs in.

Account Usage Events

Standalone System: When a user logs into a *standalone* system, there are a distinct 4624 Type 2 and 4776 events written to the security event log

(assuming that the policy is set or the system is Windows 10 or Server 2016.) The "Security ID" field is set to the local system name and the local user account. The account domain is set to the local system, meaning that the SAM database used to authenticate the user is one resident on the physical system. By default, a *standalone* system has the default value "WORKGROUP" in the Account Domain field unless the NetBIOS workgroup name is changed. For a standalone system where someone logs in with a Microsoft account (a cloud account used to connect the system to the Windows Store and other identity), such as a home user's system, the process is similar to a standalone login.

Domain System: This process is different when a user logs onto a workstation and the user is authenticated from the domain. There is a 4768-event written to the DC that authenticates the user, a 4624 event is written to the local security log with the domain name in the Security ID and the Account Domain fields. When users terminate their session, there will be a 4647 followed by a 4634 event. As users authenticate to Windows file shares, there are 4624 Type 3 events record on the *serving* system. As users authenticate to other Kerberos integrated services, there are 4769 events registered on the DC.

Table 12 4624 Logon Types

Event ID	Level	Name
4624	Informational	An account was successfully logged on. The Logon Types are: 2: Interactive (keyboard/screen on system) 3: Network (shares) 4: Batch or Scheduled Task 5: Service (services applet) 7: Screen Unlock 8: NetworkCleartext 9: New credentials such as using RunAs 10: Remote Desktop or Terminal Services or Remote Assistance 11: Cached credentials (off domain)

Table 13 Other Logon Events

Event ID	Level	Name
4740	Informational	A user account was locked out
4624	Informational	An account was successfully logged on The process ID in a 4624 event can tie into 4688 events.
4634	Informational	An account was logged off
4625[32]	Informational	An account failed to log on. Note: your SIEM solution should supply you with the underlying reason in the alert, based on the subcodes listed in Table 14 Account Logon Failures Status Codes for Event ID 4625.
4648	Informational	Logon attempted using explicit credentials (RunAs, scheduled task runs as a specific user, uses alternate credentials, and a user runs a program requiring admin rights and User Account Control enabled.)

Account Lockouts (Security: 4740, 4625/0xC0000234): The applicability of this alarm will depend on the time of day and the account time. Numerous lockouts Monday morning are "normal", while account lockouts significantly outside your organizations normal working hours are suspicious. Also, monitoring account lockouts conditions are an example where SOC can provide operational value. For example, if a service account is locked out, then the application itself is very likely down, degraded, the service capability is under attack, or a script is misconfigured.

Account Logon Use Cases:

1. **Concurrent console logons (4624, type 2) from multiple sources within a short timeframe**: This condition indicates an account is being used from multiple systems. For *most* users, any count above two is out of the ordinary. For example, an instructor in a classroom may make a change to all the classroom PCs, but an accounting staff member is unlikely to logon to three PCs at once.
2. **Logons from internal and external, within a short window, not over RDP (4624, type 10)**: This condition may indicate account misuse, credential theft, account sharing, or a behavioral issue. Note that to make this use case effective, you will need to correlate *specific* account types that indicate *user*

[32] For Windows 2000/2003/XP, the Event IDs are 529, 530, 531, 532, 533, 534, 535, 536, 537, and 539. Hopefully you will not need this information ☹.

presence such as a PC console logon and a VPN login, not an RDP login and a VPN login.

3. **Geographically improbable VPN Logins**: Modern VPN systems can provide country and city of the source IP for a connecting IP, or the data can be enriched with geo lookup data as it arrives at the log collection point. More sophisticated platforms like Splunk actually have built-in functionality to detect geographically improbable access. This means that a user logged in from one location and subsequently logged in soon after from another location that they could not travel to in the time allotted. For example, from France at 10 AM and then Canada at 10:15 AM, same day. If your SIEM doesn't have this functionality, then as a simple check, check the country code to make sure it matches the country for your user population. Depending on the user population, this check may reveal a compromised account. For systems that have a tracking list functionality, check the current login state or province and country with the prior login. If they are different and an insufficient time has elapsed from the prior login to allow for travel, there may be a problem.
4. **Network Switches/Routers/Devices**: These devices represent the network infrastructure beneath the network operating system and application stack. Thus, they must be kept secure. Even in large corporations, there are often a small group of well-known users that access the network support fabric. Regardless of the authentication method (RADIUS, TACACS+), if a user not from this group attempts to access network hardware, an alarm should be raised due to an unauthorized user access attempt.

Account Lockout use cases:

1. **Lockouts that originate from an external source:** Once lockouts reach a certain level from an externally facing login point, such as a Citrix remote desktop server or a VPN server there is external password guessing in process.
2. **Rhythmic lockouts:** Multiple periodic or rhythmic account lockouts can indicate password guessing, a service with outdated credentials, or a script attempting to logon with outdated credentials.
3. **Multiple lockouts *from different sources:*** These events will occur when the Workstation Name and/or the Source Network Address are different. If the count of unique sources is greater than 2, investigate why. The various reasons behind an account logon failure and the status codes. Review the table below to create more specific alerts. For example, a few events with a code of 0xC0000064 or 0xC000006A simply indicate user error. However, varying unique user names from the same source workstation name and/or the source network address indicate that there is account reconnaissance in progress.

Table 14 Account Logon Failures Status Codes for Event ID 4625

Status/Sub Status:	Name
0xC0000064	User name does not exist
0xC000006A	User name is correct, but the password is wrong
0xC0000234	User is currently locked out
0xC0000072	The users account is currently disabled
0xC000006F	The user tried to logon outside time of day restrictions
0xC0000070	Workstation restriction, or Authentication Policy Silo violation, which needs to be correlated with Event ID 4820 from a DC
0xC0000193	Account expired
0xC0000071	Account has an expired password
0xC0000133	System clocks too far out of sync (DC to PC)
0xC0000224	User is required to change password at next logon
0xc000015b	User has not been granted the requested logon type on the specific machine

Service accounts, interactive logins: Once an application that requires a service account is stabilized, interactive login (RDP or Interactive) should not be required. If a service account is used for Interactive or RDP logon and the designated account holder cannot explain its use right away, the account is compromised.

Microsoft Routing and Remote Access

Microsoft has had dial in and VPN functionality since at least NT 4.0. There are dozens of RRAS events which provide quite a bit of logging. Today, chances are that remote access is provided with a different VPN technology. If you are using RRAS, be aware that the tracing level logs need to be enabled using the RRAS console, and they are written to %windir%\tracing.

Normal logging needs to be enabled in the RRAS console. Choose "Log all events". Normal events are written to an actual log file: %windir%\system32\LogFiles

Monitoring Jump Boxes

One relatively inexpensive technique that can provide a high degree of security by limiting access into the server farm using RDP or SSH so it only comes from a jump box farm. The concept is that RDP, SSH, and other direct logon capabilities are blocked into and out of server segments *unless* they originate from a specific

set of jump box resources, those resources are located on their own routable segment, and only designated accounts can login to the jump boxes. Once the jump box farm is setup, deny inbound access into all other server segments for port 3389 (RDP) for Windows and 22 (SSH) for Linux machines. Some systems may use VNC on port 5800/5900, or X11 services on port 600X. Include any other remote desktop equivalent not listed here.

Jump boxes are an excellent example where an EDR applications can really shine because they provide highly granular awareness of EXEs launched, network connections, registry activity, and so forth. All jump boxes should have WEC/WEF enabled.

Jump Box uses cases:

1. **Remote connections:** Any remote access management attempt into the server segment(s) not from a jump box or the jump box network segment is, at best, suspicious and disallowed if at all possible.
2. **Limited exe's:** A limited set of executables should run on the jump boxes. Any executable outside of what's needed to permit management to and from the server segment needs to be run down. Not only that, the inventory of executables running on these systems should be very stable.
3. **Limited user access population:** Only a specific set of users should be logging into them, so login attempts outside of that group should cause an alarm. If these user accounts are in a specific AD group then that group can feed a list within the SIEM. If the user accessing using RDP or SSH isn't in this list, raise an alarm.
4. **Service accounts:** Service Accounts should not be logging into a jump box.

Table 15 RDP Events from Applications and Services Logs -> Microsoft -> Windows -> TerminalServices-LocalSessionManager

Event ID	Name
21	Remote Desktop Services: Session logon succeeded. Records user, session ID, and source address.
22	Remote Desktop Services: Shell start notification received. Records user, session ID, and source address.
23	Remote Desktop Services: Session logoff succeeded. Records user and session ID.
24	Remote Desktop Services: Session has been disconnected. Records user, session ID, and source address
25	Remote Desktop Services: Session reconnection succeeded. Records user, session ID, and source address.

1101	Remote Desktop Services: Session logon succeeded. Records user, session ID, and source address.
1103	Remote Desktop Services: Session logoff succeeded. Records user and session ID.
1104	Remote Desktop Services: Session has been disconnected. Records user, session ID, and source address.
1105	Remote Desktop Services: Session reconnection succeeded: Records user, session ID, and source address

Table 16 RDP Events from the Security Log

Event ID	Level	Name
4624, type 10	Informational	An account was successfully logged on. This is a generalized event.

Network Hardware Devices and Appliances

Network hardware such as switches, routers, access points, storage systems, IP cameras, door controllers, acceleration servers, load balancers, and all sorts of appliance-oriented systems are initially configured with default local accounts like "admin" or "root". *All of these devices* must be configured to centrally log, must be configured to use centralized time services, and *most of* them *should* be configured to use a central directory such as Active Directory via LDAP for user authentication. There may be some systems that have a characteristic that can justify a local account database, but it is unlikely that there is a solid reason why those systems should not centrally log.

Network Hardware Use Cases:

1. **Identify Network Hardware:** You can achieve this objective (or at least make progress towards achieving it) by scanning the network with nmap, looking for a response from ports 443/TCP, 80/TCP, and possibly 22/TCP. If systems are responding and not generating a log record, or at a minimum not seen as a source client IP for authentication on an AD DC, then an appliance of some sort is identified. Next step is to identify the system and determine if it can log, and should be configured to log.
2. **Collect authentication and change activity:** Depending on the device, you may want logging from them. At a minimum, you want change activity, user logins (success and failure), and system reboots.
3. **Monitor for default account attempts:** Logins to network hardware should be monitored so that default accounts like admin, root, and supervisor are not used.

4. **Monitor for outbound traffic:** Most network devices should generate very little outbound traffic outside of a few specific sites, which are most often for content or system updates. Alarm conditions will vary. For example, tuned alerts if a piece of hardware makes DNS requests or communicates to sites outside of a small list. Alternatively, review outbound traffic from a piece of hardware periodically to detect anomalies. Items of note include NTP requests, DNS requests not to the local DNS, and software updates from vendor network names.

Printing

Print servers can be configured to record when a user prints a document, the document name, and document size. You are more likely to need to enable print job monitoring through event log forwarding to support long term employee investigation. In order to support any assertion other than "User X printed a job to printer Y at time Z", you will need to enable supplemental auditing[33] to capture the print job name and conduct a forensic examination of the workstation or have an EDR application in place in order to fully support this degree of attribution.

Based on empirical observation, the Windows Print event order is: 800 -> 801 -> 311 -> 842 -> 804 -> **307** -> 802. Of these, the most relevant are 311 and 307.

Table 17 Windows > PrintService > Operational

Event ID	Level	Name (based on empirical observations)
307	Informational	Print Document owned by user (identifies user, print server name, printer, and if auditing is enabled, the file name).
800, 801	Informational	Print Job Diagnostics (spooling)
311	Informational	Printing a Document (Identifies the user and printer name)
824	Informational	Print Job Sandbox / Isolating print job
802	Informational	Print Job deletion
842	Informational	Print job isolation and print process tracking

[33] The GPO path is: Computer configuration >> Administrative Templates >> Printers>> allow job name in event logs.

Operating System Security, Change, and Stability

There are several conditions that affect system security and stability. By being able to monitor these conditions, the SOC can support helping to identify system stability issues and help to identify operational issues.

Operating system stability Use Cases:

1. **Adverse events by population**: The same adverse stability event occurring across N% of your environment. Think 2%, so if you have 1,000 servers that means an error occurs across 20 systems within a 24-hour period. As you find and remediate systemic issues, you would set this higher. Rather than focusing on "100 systems", though, this metric is better related to a single digit percentage of servers, because that metric has operational and security value. This particular condition is where the event taxonomy will come in handy, because identifying all of the source events by specific type would be exhausting.
2. **Security service failures:** The use case relates to security focused services failing, because that can indicate the environment cannot be properly monitored or active tampering is occurring.
3. **(Un)Installs outside of the change or maintenance window:** For changes (1022, 1033, 903-908), SOC should be able to perform long tail analysis of the installed application on a daily basis. For centrally deployed applications, the count of successful installation should be the size of the target population.
4. **Clearing the event log**: When configuring an alarm for this condition, make sure the alarm is for the security service and not the ADFS service. The ADFS service logs events to the Security log with event ID 1102. Clearing the event log should rarely, if ever, occur on the network. There are multiple configuration changes that can be done to compensate for reasons someone would cite to clear a log. For example: if the event log is too large, then its size can be reduced through group policy by increment, like dropping the size by 10% every six hours until the log reaches 128MB[34]. As events are written, the log will naturally trim itself. An OS can also be configured to shut down if the log is full. Given these conditions, about the only reason for legitimate clearing of the log is if it is truly corrupted and a reboot didn't fix the log.
5. **New Services**: Windows records a new service installation with Event ID 4697. These are infrequent events and should be supported with a change control item.

[34] This number is based on using Server 2008 and 2012 in a highly virtualized environment. The Windows admins found that was a size that provided several days to months of record keeping and still allowed the Event Viewer to be responsive. YMMV.

6. **New Scheduled Task**: Windows records this event using ID 4698 in the security log a very common and almost 100% reliable indicator of lateral movement when there are local logons (4624), new service (4697) and new task (4698) within two minutes of each other.
7. **Windows kernel errors (Blue Screen):** BSOD's are an issue for operational reasons as well as security reasons. Implementing a Windows rootkit is actually more difficult than one might think. If BSOD's occur on any recurring basis, it very may well be worth investigating why from a security perspective as well as an operational perspective.

Table 18 Windows OS Stability Events

Log	Event ID	Level	Name
System	104	Informational	Event Log was Cleared
Security	1102	Informational	Audit Log was Cleared
System	7000, 7023, 7024, 7026, 7031, 7032, 7034, 7013, 1069	Error	A service failed to start – these events can relate to login failures as well as general startup error and are varied.
Application	1000	Error	Application Error
Application	1001	Error	Application Hang
System	1001	Error	Blue Screen (system faults), also called a BugCheck
Application	Various	Warning / Errors	Several applications report specific errors and warnings in the local app log.
System	6	Informational	New Kernel Filter Driver
User32	1074	Warning	Shutdown Initiate Failed
System	1022, 1033	Informational	New MSI file Installed
Program Inventory	903, 904	Information	New Application Installation
Program Inventory	907, 908	Information	Removed Application
Program Inventory	905,906	Information	Updated Application
System	19	Information	Windows Update Installed
System	29	Error	Windows failed fast startup (My Exp – hardware related.)
System	41	Error	The system has rebooted without cleanly shutting down first. (Win10)

Log	Event ID	Level	Name
			These are Blue Screen of Death (BSOD) messages
Security	8222	Success Audit	Volume Shadow copy created.

Table 19 Microsoft-Windows-Kernel-Power

EventID	Name
41	The system has rebooted without cleanly shutting down first. Was there an adverse indicator right before this event – malware attempted install, NIDS alarms, excessive failed logins, failed hardware, failed service?
6008	The previous system shutdown at on was unexpected. Registered when the system actually restarts. Time delay between a shutdown and a restart should be established to make sure that the hardware recovers. Target no more than 5 to 8 minutes until you have more accurate data.
18	A fatal hardware error has occurred.
7000	The service failed to start due to the following error. In particular, security focused services (DLP agent, anti-virus, software deployment, etc.) should never fail to start.

Data Leakage (USB Insertion)

Windows has improved in its ability to track USB insertion events as new versions are released.

Table 20 USB-USBHUB3 Events

Log	Event ID	Name
USB-USBHUB3	43	New Device Information (limited observation)
USB-USBHUB3	400, 410	New Mass Storage Installation However, you will need to research storage volumes to use this properly.

Also, if auditing is enabled on this log area – "Application and Services Logs > Microsoft > Windows > DriverFrameworks-UserMode > Operational", Windows will log up to 18 events in this log when a USB drive is inserted into a system.

Table 21 Windows > DriverFrameworks-UserMode > Operational (USB, Win10)

Event ID	Level	Name

2100, 2102, 2105, 2106	Information	Pnp or Power Management operation to a particular device
2003, 2010, 2004, 2006	Information	Loading drivers to control a newly discovered device

On Windows 10/Server 2016, if "Audit PNP Activity" is enabled, the system will log these events relating to USB usage (and some others):

Table 22 Audit PNP Activity USB events

Event ID	Level	Name
6416	Informational	A new external device was recognized by the System. (Will include the name.)
6422	Informational	A device was enabled.
6423	Informational	The installation of this device is forbidden by system policy.

Brute Force Failed Authentication Attempts

This particular use case does have a nice pattern to it. Some consistent percentage of your user population will forget their account credentials routinely, every Monday morning, and will repeatedly try to login will lock their accounts, wait a little bit, and eventually call the Service Desk for assistance. Outside of that window, repeated account lockout conditions that repeatedly occur indicate one of a few things:

- A completely misconfigured system or application
- A user with a device that has an old credential that needs to be updated
- The account that is repeatedly locked out or failed to logon and is under a password guessing attack.

Misconfigured systems also have a detectable pattern: a rapid number of events from the same source address which repeats, and then either a well-defined pause or the events stop altogether as the application usually throws an error to the user. Note that the use cases below mention lockouts (Event ID 4740). Your organization may, or may not, have enabled that setting in AD.

Brute Force authentication use cases:

1. **Continual Authentication Failures and Account Lockouts**: Once failed logons reach a certain threshold or clipping level, there is a reason to monitor the account and investigate. For failed logons, consider starting at 30 failures within a 10-minute period and adjust from there.

2. **Repeated Account Lockouts**: Once an account logon failure occurs often enough, more sophisticated systems like Active Directory can be configured to lock the account. For monitoring account lockout, start with 5 consecutive lockouts before you trigger a brute force alarm, assuming the account lockout policy is set to 5 failed attempts and then the account locks for a short period of time like 1 minute.
3. **Low and Slow Authentication Failures:** Attackers need to determine how account authentications failures are handled by various applications. More specifically, web-based applications frequently have observable differences if the account name is not known, the user and password do not match, or the account becomes locked. These differences can be observed in web page results and how long it takes the site to respond, on average. Therefore, an attacker may attempt a lower volume of authentication attempts in order to discern the pattern.
4. **Password Spray:** This attack attempts to use a single likely to be true password during the authentication attempt against all accounts in the domain, one time per account. The goal is to attempt to discover an account reset to an organization's default like "Spring2018", or to see if accounts use one of the most common passwords[35]. This type of attack can be found by searching for failed logons for multiple unique user accounts from the same source machine (name or IP address).

DHCP and Data Link Layer Analysis

DHCP and data link layer (L2) analysis can reveal significant issues on the network. In order to perform these checks, your organization needs some supporting controls and practices in place to provide system awareness at the MAC level. This discussion is focused on automating the analysis processes for rogue device detection. The lists described have a specific purpose. The system in place may achieve these lists by having an attribute for the entry purpose that may work as well as separate lists.

First, you will need to build and maintain a list of known MAC addresses for "quiet" stand-alone devices. There are all kinds of devices that don't supply a host name during the lease process, don't participate in a Windows domain, and can exist on the network. If you plan on identifying devices that are not authorized, you will need to be able to identify and exclude authorized devices from the analysis cycle by MAC address. Note that this is not foolproof – it is not difficult to learn and reuse a MAC address as there is often no real security at this layer.

[35] There is a Wikipedia article with the top 25 most common passwords that can be used. https://en.wikipedia.org/wiki/List_of_the_most_common_passwords (8/18/18)

Second, you need some sort of a naming convention so that you can disregard known systems, instead identifying a system that is "not like the others" that can be implemented during the DHCP lease negotiation process.

Third, end user systems and VOIP phones should go through a pre-provision process to record their MAC addresses so that the system won't generate alerts when they come online. For example, a set of ports and a specific DHCP range can be used for pre-provisioning, a client reservation can be created ahead of time, or an unmonitored segment can be used. DHCP traffic from the provisioning segment can then be automatically added to the exclusion lists.

Fourth, and perhaps this the most difficult part, the SOC will need to know where the device actually is by switch port, floor, and building so that a technician will know about where to go to identify a rogue device. This will require naming of the DHCP scope with location information like "Bldg 16 3rd Floor East" and importing that into the SIEM as a network description. From there, the SOC can log into the switch and get the exact port information. Assuming that switch ports have the corresponding jack identifier and floor location, the end systems location would then be known. Yes, there are organizations that actually go to this level, I have seen it, they do exist.

Fifth, you will need a way to identify and differentiate networks by type so that you can create better notifications: VOIP, desktop wired, portable wireless, utility services such as conference room support or HVAC, guest, and any other network type that can be used to differentiate the types of assets on the network segment. Understanding the segment usage will further sharpen alarm development.

DHCP and Layer Two Use Cases:

1. **Non-Phone VOIP devices on VOIP networks:** If there is an identifier such as a VLAN scheme or an IP scheme that can identify IP telephony equipment you would want to know if devices other than phones are on that network. Examples: Odd numbered DHCP networks for end user network regions may be reserved for VOIP, while even numbered networks are used for end user system support. Device names may be useful. Phones may be assigned a hostname pattern, so that if a phone appears on a desktop segment the phone can be quickly identified and moved to a phone network, or if a non-phone appears on a phone network it can be identified. Or your call manager may be able to export a list of known MAC addresses of enrolled phones, and if a device that is a non-enrolled phone shows up on the VOIP networks it can be investigated (this example assumes some sort of pre-enrollment capability). Also, checking the first part of the MAC address

against the OUI identifier for the organizations VOIP phones can help improve rogue detection.

2. **Non-known workstations appearing in the desktop segments**: The ability to detect this depends on an established naming convention or full MAC registration and knowledge. For example, if you named all of your desktops something like "DSK####", portables "NB#####", and developer virtual machines "DEVVM#####", then you can monitor for two conditions of note. A system that registers with DHCP and doesn't supply a known host name in the pattern should alert. Alternatively, if there is full end workstation MAC registration, then the alarm can be raised when the "MAC address not in Workstation List" (which, of course, will need constant maintenance...)
3. **Security ISO:** There are known bootable ISO's that supply default names which can indicate malicious intent. This list will change over time though. As an example, if a host named 'kali' show up it means someone has booted a known penetration testing distribution.
4. **Rogue systems:** If a system appears that has no name, not in the auto provisioned MAC list, not in the manually build approved MAC list, doesn't supply a host name, is not on a guest network, and doesn't authenticate to the domain within a reasonable time (say 5 minutes), you have a ... Rogue One.

Next Generation Layer 7 Firewalls

Today's firewall is not simply a stateful inspection engine as they were in the 1990's. Today there are complete open source stateful inspection firewalls such as PFSense which are more advanced. As the security market has matured, next generation firewalls (NGFW's) such as the Palo Alto, WatchGuard, Sophos, FortiNet, and others are capable of acting as multilayer application gateways complete with TLS/SSL Interception with full "layer 7" analysis capabilities. Thus, they are capable of more sophisticated decision making and alerting that a stateful firewall won't provide. NGFW's provide numerous detections and alerts into a SIEM and can be placed in the ideal location, as a perimeter point control gateway.

For example, the Palo Alto NGFW (when subscriptions are purchased and enabled), can check URLs against an authorized list, site categorization, check files against a known threat database, apply an access policy by username, source address, or time, enforce protocol usage by TCP or UDP port, and numerous other application level controls as well as being the perimeter firewall. PanOS7 and 8 have eight different log record types, with some records having over 50 fields, based on the analysis performed and the result provided from its analysis engine.

One challenge that the SOC team faces is understanding what NGFWs do, how they process data and how they report conditions to the SIEM. In order to make best use of these systems by the SOC, a "mini manual" should be developed that explains the NGFWs capabilities and how to properly read the log data generated by the NGFW.

NGFW's can function like web proxies, may be able to perform file checking, and may have NIDS functionality. Review the use cases in other sections to determine how applicable they are to the NGFW. Many of the other use cases in this chapter can be applied to a NGFW.

TOR Overlay Networks

TOR is an overlay networks that require specific software to access. For some organizations, traffic using TOR may be complexly normal, and for others TOR usage may indicate a significant security risk.

TOR network use cases:

1. **Software analysis:** Looking for applications by name that are used to access TOR networks like the TOR browser and TOR messenger.
2. **TOR exit note IP addresses:** Accessing published IP addresses of TOR sites, which can be done by comparing outbound firewall traffic to a known list or by building NIDS rules to detect SYN packets to IPs in the list.
3. **TOR TLD:** The top level domain (TLD) for the TOR network is .onion. A SIEM rule to detect DNS A or AAAA record queries for domains containing .onion can be implemented. Also NIDS rule sets can detect DNS traffic attempting to resolve .onion domains.
4. **TOR SSL:** SSL analysis can be done against .onion certificates. The Emerging Threats ruleset has NIDS rules to generate these alerts.

Tor Exit Nodes:

- TOR: https://check.torproject.org/cgi-bin/TorBulkExitList.py?ip=1.1.1.1 or https://torstatus.blutmagie.de/
- If you cannot access the TOR site, there are others who provide the same information. For example: https://www.dan.me.uk/tornodes or https://udger.com/, who has a subscription service and maintains list of known attackers, fake search engines, TOR exit nodes, and other CTI type sources.

DarkNet Unused Network Monitoring

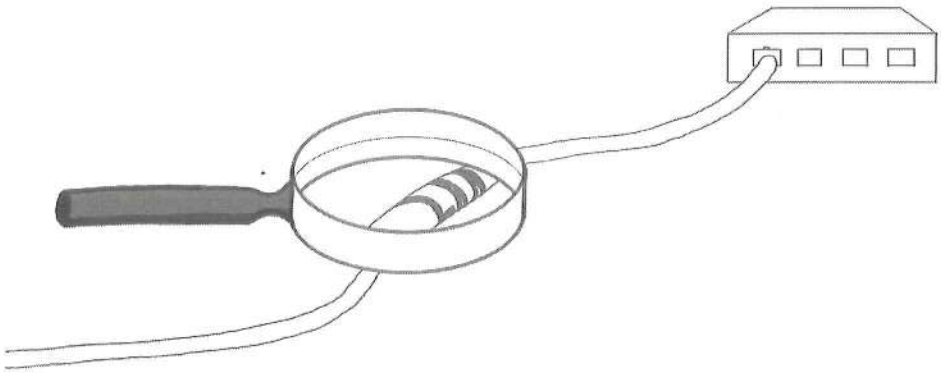
As defined in BTHb:SOCTH, a DarkNet network is an unused internal network range. For example, if your organization uses the 10.0.0.0/8 exclusively, then the entire 192.168.0.0 range and 172.16.0.0 to 172.31.255.255 range are "dark". Traffic directed towards those IP ranges or originating from those IP ranges is inherently suspect. As a recommendation on internal DarkNet monitoring, gather the list of all *known and in use* IP ranges, and then build a list of network segments *not in that list*. You would then build a report, query, or a script to analyze log data once per day to determine if there is internal traffic in the exclusion list. When you find IPs in use on networks that are not defined, then, at a minimum, you will have awareness that your network model within your platform needs to be updated. If you detect a system attempting communication to most of the IPs in a given segment, and then multiple segments, you have found an active scanning host. Common data sources to check for internal DarkNet monitoring include DNS requests, DHCP activity, authentication events to Windows domain controllers, and outbound firewall activity. There are numerous other sources – but these are the ones most likely to get a hit if a DarkNet is in use.

Network Intrusion Detection / Prevention

NIDS/NIPS systems monitor the data on the wire itself as it traverses a TAP, mirror or SPAN port, or are placed inline. These platforms search for patterns such as network scanning, protocol/port mismatches, command and control behaviors, certificate exchange issue detection, tunnel decode, and matches against a wide variety of signatures. Even the Bro network analysis system can extract files from network flows and pass them off to an analysis engine or just store them for later analysis.

Today, the challenge to a NIDS/NIPS engine's effectiveness is that more and more sites are using encryption, so when a normal legitimate website is compromised and used for nefarious purposes, the NIDS/NIPS will not be able to inspect the application flow or stream contents after the initial certificate exchange and session key is built out. Common deployments of NIDS/NIPS are at a chokepoint such as the perimeter or a DMZ, so positioning *inherently* affects their effectiveness. Also, NIDS/NIPS must be configured to know untrusted and trusted segments. NIDS/NIPS rulesets are configured in of two ways – to detect an attack from any source to any destination, or in a specific direction flow (trusted to untrusted, for example). The net effect here is that

the ruleset may catch *inbound traffic*, but may not catch someone inside using



an attack against an outsider. As a compensation, multiple NIDS/NIPS instances can be deployed with tuned rule sets based on *directionality*.

NIDS/NIPS signature detection varies when it comes to identifying true or false positives or negatives. Some rules, such as those that detect connections to the DShield.org top attackers list is very good, while rules focused on current events occurring "right now" will need tuning in order to improve over time. Further, NIDS rule effectiveness will diminish over time as the security issue they identify is dealt with. A rule that detects cleartext telnet logins against network hardware becomes less valuable once no more telnet access enabled. A rule for a spam campaign from four years ago is another example.

NIDS/NIPS Use Cases:

1. **Same alert, high volume, single target:** Repeated alerts directed towards the same "target" need to be either a) tuned because they are most likely a false positive or b) should be investigated because the rule is firing on a serious condition.
2. **Same alert, multiple targets:** When an alarm arrives for multiple targets, the same general rule applies – determine if the rule can be tuned based on an understandable condition, and if not, investigate.
3. **Multiple alarms, same system:** There are several rule conditions which can "stack" on one another or relate to part of the kill chain. When a system has multiple different alarms, especially if those alerts indicate that a machine is both the source and destination, then it is a sure sign of compromise.
4. **Vulnerability Correlation**: There is one area where ID/PS detection can really payoff: when a signature detects traffic against a port or service that has a known vulnerability validated from a vulnerability scanner, you have a high value alarm. This may also manifest as ID/PS alarm against a known

vulnerable application, regardless of the TCP or UDP port. Any alarm that can leverage a known vulnerability should be investigated.

5. **Known command and control**: Botnet or command and control triggers based on IP addresses, domain names, pulse patterns, user agents, and other conditions. Note that malicious domain names are frequently updated and there are numerous sources that can produce this information. When the NIDS/NIPS picks up C2 communication, look for secondary clues to conform C2 and other systems on the network involved in the same activity.
6. **TLS/SSL Blacklists**[36]: There are known certificates which are used by malicious software and botnets. This type of alarm can be applied to HTTPS traffic by matching the certificate necessary to setup the TLS connection. The abuse.ch site maintains a frequently updated set of lists as Suricata NIDS/NIPS rules, so to use them there is a technology dependency. For example, if you look into the open source Emerging Threats Botnet Command and Control Server Rules section, what you see is that there is a daily process to build NIDS rules based on the IP addresses where malicious TLS/SSL certificates were recently observed. These rules are built based on malicious TLS/SSL certificates identified by abuse.ch and shadowserver.org. They are written in the form "for any system in the HOME network from any port, generate a message if there is observed traffic on any port based on a known IP or if a TLS fingerprint is observed" with a reference to the rule and a well mapped name in the message field. The IP address technique works well when a *single IP address is presenting a malicious site,* but not as well when the IP is tied to dozens or hundreds of websites from a hosting provider.
7. **High alarm counts and singleton events:** Alerts at *either end* of the spectrum need specific attention. The top 3 to 5% of ID/PS alerts should be checked daily in order to tune them so they occur less often, or can be disabled. In contrast, there is hidden gold in 'singleton' alerts, *particularly if there are a few unique singleton related IDS signatures for the same source or destination*. Map the event names using a long tail analysis method.
8. **Internal vulnerability scanning:** *Internal* to *internal* port and vulnerability scanning should only come from a few authorized sources, like the Vulnerability Assessment platform or the security engineering team. Outside of these few known IPs and users, *internal* scanning is suspect.
9. ***Internal* to *external* scanning:** While it is certainly permissible to conduct vulnerability assessment or scans against your own systems, any other scans can be considered an intrusion attempt, may indicate an attacker has control of an internal system, or a user up to no good. For example, someone could be moonlighting and running their own pen test business from the organizations network.

---

[36] https://sslbl.abuse.ch/blacklist/

10. **Include Eric Conrad's Whitecap rules:** These rules are designed to detect malicious use of ICMP traffic. The ruleset is implemented to ignore (passes) known good ICMP, and alerts on the rest, so it takes advantage of how a Snort ruleset is constructed. The rule set, in effect, whitelists good traffic and then assumes that any other ICMP traffic should be investigated, so they take advantage of rule processing order and the pass rule option. ICMP can be used for tunneling, and it can also be used for communications.

ID/PS Signature Updates are a *significant* part of maintaining their effectiveness. Most commercial vendors provide some form of automated update process, or a simple process to download the current content update, review, and then commit it to the system. Open source NIDS systems like Snort and Suricata have a variety of scripts to pull down rule sets and use an Oinkcode key value with the ruleset provider.

Perimeter Security Focused Access

Every organization has a perimeter point. This is the demarc router and the primary perimeter firewall. Realize that with the advent of highly portable and powerful computing devices the 1990's view of the perimeter is effectively dissolving, so traditional prevention first thinking needs to migrate to assume compromise, improve detection, and hunt for the adversary.

Some Interesting statistics help characterize data that flows on the Internet, and thus will affect your perimeter[37] in varying degrees.

1. 51.8% of all Internet traffic is from bots, 48.2% is from humans.
2. Over 50,000 websites are hacked every day.
3. As at June 2018, there are approximately 1.89 billion websites.
4. 338 million domain names were registered as of first quarter of 2018.
5. There are 3.7 billion global mobile Internet users as at January 2018.

For small organizations or sites with just a few IP addresses, the Internet access line may actually plug straight into the external firewall interface. Larger organizations, particularly those with fault tolerant firewall configurations, will have a gateway on the router and routable IPs on the firewalls.

[37] Statistics are from https://hostingfacts.com/internet-facts-stats/ as of August 22, 2018.

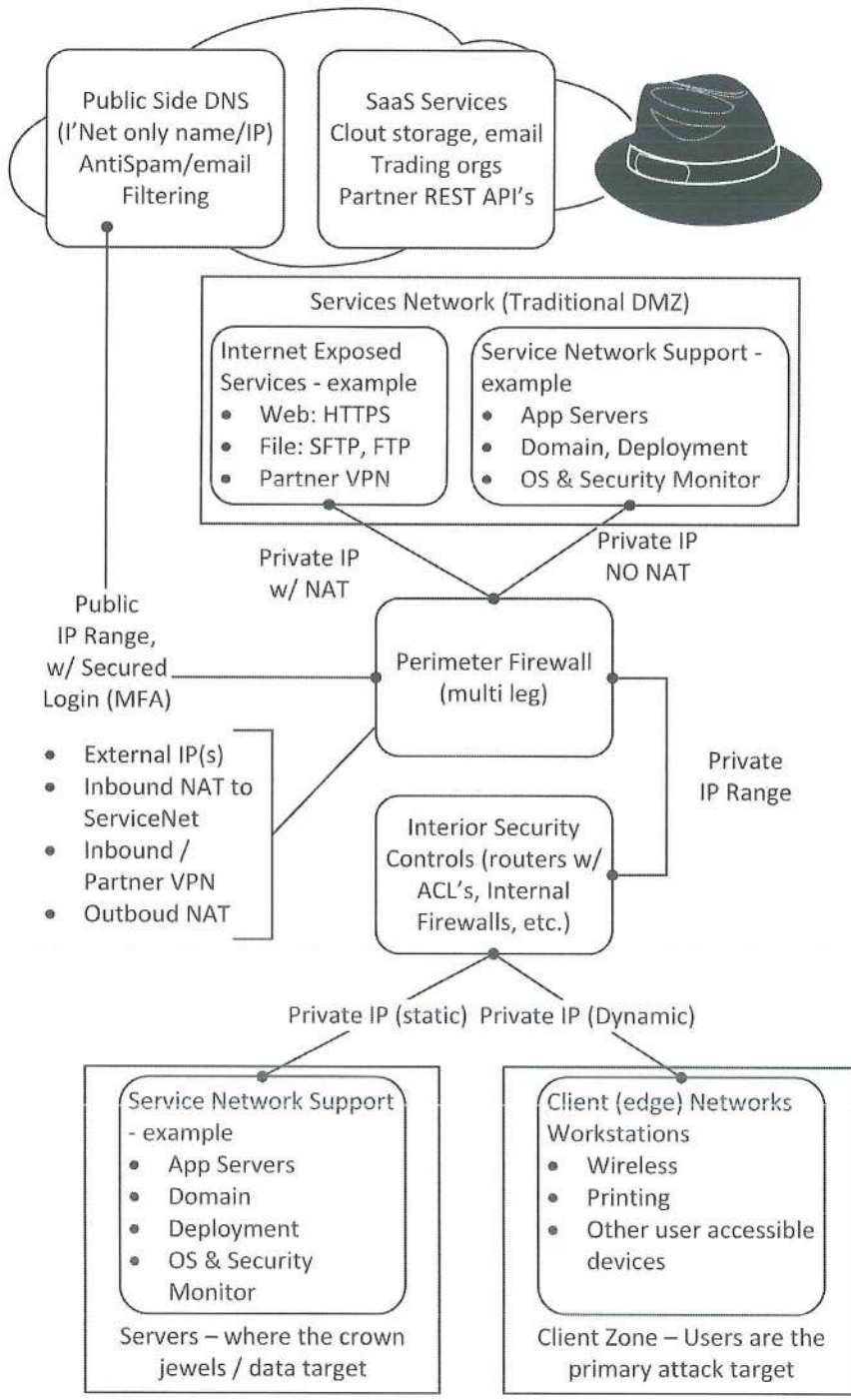


Figure 4 Perimeter Use Case Illustration

Very few company resources, if any at all, should ever be "outside" the firewall. The perimeter router is the last control point for outbound traffic and therefore

it is the first control point inbound. You should know where NAT translation occurs - on a perimeter router or in the firewall.

Organizations take very different approaches to applying security controls on the perimeter router and logging activity. In the next figure, several security zones are represented that may have different logging levels. Each one of the intersection points is an opportunity to collect session data with tools like a firewall, NetFlow, or Bro IDS. Each of the systems represented are also examples of application and operating system data sources.

One challenge you may have is resolving NAT translation because the SOC wants the true source IP address whenever possible. For example, if a user accesses the web interface on the DNS hosting site from the corporate firewall, you won't know the true client IP *from the DNS hosting site's perspective.* The best you could do is match access logs by adjusted time, source IP and port to the site perimeter firewall with its state logs of which internal IP visited the DNS site at the same time through the same outbound TCP port.

Regardless of how the perimeter devices are managed, *any and all changes or direct logins* to the perimeter router and firewall should generate a log record into the SIEM.

Architecture and configuration details provide considerations that may allow for data reduction and filtering. Examples are:

1. **Inbound accept logging:** Data will be duplicative, as most inbound traffic will be protected by the perimeter firewall and possibly a local firewall for an internet facing resource.
2. **Inbound deny logging:** Your organization should have a default inbound deny policy in the firewall. Occasionally there is a desire to send tHypberhe perimeter router logs into the SIEM. This is suboptimal because this posture will log a large amount of traffic that will provide very low value. The commodity Internet is loaded with viri, scanners, and attackers just looking for any opportunity to get in – all of which occurs 7/24/365.

Perimeter Traffic Use Cases and Detection Rules: There are thousands of application level protocols in use today. Many of these are service protocols. As you can see in Figure 4 Perimeter Use Case Illustration on page 108, several services are depicted in use on the service network. From the figure you can see a few application protocols in use on the service network. Rather than write dozens of pages that say why you should or should not allow a given network layer protocol or application layer protocol, this section will define a model for you to evaluate *if* a protocol should be allowed or not, and if not, how you can monitor for protocol usage.

IP Network Layer Protocols[38]: The next layer network protocol is defined in byte 9 of the IP header. Common next layer protocols that are likely to traverse your perimeter are listed in Table 23 IP Next Layer Protocol Numbers (IPv4) on page 110. Many of these network layer protocols are not in active use today. Common protocols your organization needs include TCP, UDP, ICMP, some form of routing, and likely use a VPN that is based on TLS, L2TP, or IPSec.

Table 23 IP Next Layer Protocol Numbers (IPv4) Likely to be in Use

ID#	Protocol	ID#	Protocol
1	Internet Control Message Protocol	67	Generic Routing Encapsulation
2	Internet Gateway Message Protocol	50	Encapsulation Security Protocol (IPSec)
6	Transmission Control Protocol	51	Authentication Header (IPSec)
9	Internet Gateway Routing Protocol	6	Border Gateway Protocol uses TCP over port 179
17	User Datagram Protocol	115	Layer 2 Tunneling Protocol Version 3

For the remaining IP Network Layer protocols, the SOC should work with the network engineering team determine if the IP protocol is actually needed. If it can be demonstrated with a communication flow and that is in in active use to support a business process, it is needed. If not, ensure that you can monitor for it or deny the IP layer protocol.

An example: You may never have heard of Stream Control Transmission Protocol (SCTP). Are you aware that this protocol has innovative features beyond TCP, such as multihoming and multistreaming in a single SCTP association? It is becoming more popular, particularly for web servers? Can your NIDS decode SCTP data? As of August 2018, Suricata can parse SCTP, but there are very few rules available in the Emerging Threats pro feed. Coverage is limited to some very specific areas such as Denial of Service and a few buffer over flow conditions. Also, the manual lists that Snort won't handle multiple encapsulations[39]. Would your firewall admin even think to mention enabling a new IP layer protocol to the security team? Here is an example protocol level advisory. ForcePoint[40] advises "A flaw was also found in the [at least Red Hat] Linux kernels implementation of SCTP protocol in which a remote attacker can

[38] IANA's protocol list: https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml

[39] http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node10.html

[40] https://support.forcepoint.com/KBArticle?id=CVE-2016-7117-and-CVE-2016-9555-Linux-Security-Vulnerabilities

trigger an out-of-bounds read (CVE-2016-9555)." Therefore, understanding and having the ability to *monitor for protocol usage* outside of what the NIDS system can process or what you expect to see is necessary.

Protocol Detection with NIDS: In order to detect protocols that are not known to be in use, you can use a Snort rule like this for IGMP (IP protocol #2).

```
alarm ip any any -> any any (ip_proto:igmp; msg: "IGMP
Observed";)
```

Or this rule for SCTP (protocol 132, Snort doesn't have a label for SCTP):

```
alarm ip any any -> any any (ip_proto:132; msg: "SCTP
Observed";)
```

Protocols by name are listed in a typical Linux system in the /etc/protocols file.

Application layer Service Protocols: Many application protocols run on a standardized UDP or TCP service port. For example, several just won't work without extensive hoop jumping unless they run on a standard port: DNS, NTP, DHCP, BootP, SNMP, SMB, NFS, and AFP are but a few. As you can see in Figure 4 Perimeter Use Case Illustration on page 108, several services are depicted in use on the service or DMZ network. For the SOC team, it is critical to understand what services are allowed into the service network segment (or DMZ) for the organization.

Perimeter use case requirements:

1. **NAT:** The SOC *must* know all of the external DNS entries and NAT translations in order to have the best possible awareness and ability to correlate internal private IP addresses to external IPs. A usage mapping is also essential.
2. **Top 1M:** Determine your daily baseline, such as top 10,000 external IPs and whether they are in the a top 1M domain / IP list. See p. 112.
3. **Protocols in use and protocol volume:** From there, you can compare each day against this baseline to locate potential deviations, abnormal flows, or systems communicating to potential suspects using an unusual protocol.
4. **Persistence:** Also, the ability to detect persistence, such as a connection lasting more than 24 hours is a *key capability*.
5. **Forgery and Private IP:** Forged IPs, private IPs attempting to egress, and other traffic anomalies.

Top One Million Site Checks

Several data sources can be checked against a top one million site list. The premise is that well know sites which are in the list have been around for quite some time, are well maintained, and if a security issue appears it will be responded to in a timely manner. Please do not infer that a site is "completely safe" or acceptable if it is in a top one million list. The primary goal of eliminating sites in this list is to more readily identify potential suspects.

There are a few lists and sources for the Top One Million sites, such as Majestic and Cisco Umbrella 1 Million[41]. Both are still offering the list for no charge. Alexa was the first to provide this list, and they stopped in late 2016. Cisco's list is based on DNS queries, while Majestic's list is based on websites names, so there are differences at the bottom of the list, approximately 990,000 and higher.

The recommendation to monitor visited sites or DNS queries outside of a top one million list should be taken to mean that the site is acceptable in the workplace, as many adult content sites are high in the list. The intention of monitoring DNS, proxied sites, or IP addresses queries outside of these inventories is to identify newly suspect domains, sites that may have a higher probability of hosting malware or are newly registered. In effect, this analysis is a significant data reduction exercise and a great example of long tail analysis.

Remember, attackers have compromised many top sites over the years, there is malware that uses Twitter, Facebook, and a Gmail based C2 library.

Several data sources provide information that can be checked against these lists.

Using DNS name top 1M ranking. Mark Baggett, GSE, wrote a domain_stats tool[42] that can return several values from the whois record of a domain and also a ranking score for the domain based on its position in the list. If your SIEM can make a web API call, then an analyst can quickly use the result. This functionality can be integrated as a right click operation. Better yet, integrate the check to an event record with a custom attribute as data flows through log collection.

Examples below are from Mark's GitHub page:

> Alternatively, you can query individual entries in the whois record by including field names in the path.

[41] https://umbrella.cisco.com/blog/2016/12/14/cisco-umbrella-1-million/
[42] https://github.com/MarkBaggett/domain_stats

```
student@SEC573:~$ curl
http://127.0.0.1:8000/domain/creation_date/sans.org
1995-08-04 04:00:00;
```

This tells us that SANS.ORG is the 25646th most popular domain on the internet. So it probably isn't a phishing site.

```
student@SEC573:~$ curl http://127.0.0.1:8000/alexa/sans.org
25646
```

Top One Million Use Cases:

1. **Review rank ordered domains outside of top 1M list:** Pull the DNS queries over the past 24 hours, remove any local queries such as a query without a period or a Windows resource record lookup, and scan through a rank ordered list of the DNS names that are not in the reference list. The point of this exercise isn't that the top site list are sites that are appropriate for your organization – rather that you need to look for odd names, new names, or first use names so this data reduction exercise focuses your attention. Further refine the result set from the step above, and add the creation date to the list. Queries against young domains are inherently suspect.
2. **Review for DGA names:** Malware is increasing its use of domain generating algorithms (DGA) so it can figure out how to communicate to its control network through names that change daily. DGA based names tend to be long – greater than 32 characters. DGA names are programmatically generated and most often relate to the date, so that these names are new every day. These names should naturally fall out of this analysis.
3. **Firewall data:** Consider running the suspect IPs recently visited list against an inventory of top sites, and if there are site/IP isn't in that list, investigate then.

Top Ten IP Address Use Cases

This section isn't combined with another section because the "top ten IPs" list can come from a wide variety of sources, with the data source influencing the interpretation of an address belonging in the "top ten". Also – the idea of responding to the "Top Ten" completely negates looking at the "Bottom Few", which is *often where the real action is*. Dr. Eric Cole has often advised that when an IP address that is in all three of these categories: top talker, highest bandwidth, and encrypted, it's malicious. The discussion below takes those points further.

Top Ten IP Address use cases:

1. **Top Ten "outbound connections" + Top Ten "data flow" + Top Ten "Workstations" or "Servers":** This condition may indicate data exfiltration, and can be used to profile the overall network activity. When an IP appears in all three lists, spend time confirming if the machine is compromised.
2. **Top ten outbound with a connection lasting more than 24 hours:** This is another example of a possible data exfiltration, particularly if it comes from the workstation side. From the server side, SOC can build an inventory of recurring patterns of "Server A to Site B", and then eliminate them from the daily check in order to identify new long running connections. Once understood, the time will likely change based on the source network.
3. **Top Ten (or more) DNS requests for newly registered domains**: If you have a DNS logging capability or have implemented PassiveDNS, then you can take "yesterdays" DNS queries and compare it to newly registered DNS names (see the DNS section for more detail and use domain_stats for bulk checks). This result set would prompt a pivot to other data, such as NIDS. An analyst should review the result set to detect anomalies or security issues. The benefit of this particular use case is that it can be fully scripted, and once done analysis can be done 'first thing' or by the overnight SOC team.
4. **Top Ten outbound denies by source and/or destination port**: Aside from the smallest of networks, servers and workstations should be deployed in their own segments (broadcast domains) and subnets. As a result, any traffic that the perimeter devices see will be using the perimeter as the default gateway. As a best practice, the perimeter should be configured with a "deny by default" posture, meaning that only permitted traffic from authorized sources should be allowed outbound. This type of a threat hunting activity can also help identify operational issues, like systems not using proper DNS, SMTP, proxy, or NTP servers.

Web Application Firewalls (WAF)

A Web Application Firewall is an application level protective system that resides in front of a web server or, in the best case, fully integrated into a web server process. WAFs function at the application level because they understand the HTTP/S protocol, monitor traffic, can load balance, monitor the execution flow, may validate document content such as validating an XML document. and can make filtering decisions based on their rule set. To be truly effective, WAFs need to see all traffic unencrypted which complicates deployment. Systems like Palo Alto NGFW, Citrix NetScaler, and F5 Local Traffic Manager perform a TLS/SSL front end interception function so they can analyze traffic. Some WAFs can be built in to the web server process like the Apache mod_security. However,

single server module deployments may not scale well when compared with a load balancer that has WAF capabilities.

For the SOC team, the primary use case for WAF is that it functions like an IDS because it can inspect HTTP traffic and can therefore identify web application attack traffic. When the WAF finds an issue, such as a "select * from TABLE", SQL injection statements like "; 1=1", or Base64 encoded fields in an XML document where there should not be one, the WAF should provide an alarm and a detailed log record. It can provide a very useful data source for network forensic analysis.

Refer to Webserver and Application Server Activity on page 117 for further information.

Web Proxy and URL Activity (V1.02)

A web proxy is one of the more valuable data sources for the security operations team. Web proxies record which source IP went to what site, the user, date, time, should provide HTTP return codes, hopefully the user agent may record if the user is redirected, and other details about web browsing. More sophisticated proxies will provide a classification value[43] or description for the URL such as "Advertising", "Dating", or "Folklore". A web proxy will also report the access control decision. Examples of these decisions attributes are allowed, allow by specific rule, block, bypass by rule, bypass by user request, or bypass on first observed. Note, here that "who" can be a user, a server, an application.

Web proxy use cases:

1. **End user workstations *not using the proxy:*** This condition should be rare – you have a proxy and users use the proxy, right? Systems *not* using the proxy are negating the security value of the proxy, and therefore do not generate any application level log records. Users not using the proxy server(s) should be blocked, and that log record enable to log, so that this condition can be corrected. Further, by employing a proxy and blocking outbound use of common webserver ports like 80/TCP, 443/TCP, 8080/TCP, and 8443/TCP, you can detect rogue devices on the network like a disposable Raspberry PI plugged in an using OpenVPN over 443/TCP.
2. **Servers using, or not using, the proxy:** If your organization allows servers to have Internet access, then you should maintain a list of authorized sites. When a server accesses a site outside that list, the site should be checked to determine if it is a risk or should be added to the known server sites list.

[43] For example, Fortinet provides a list of categories: https://fortiguard.com/webfilter/categories

3. **Suspicious user agents**[44]:. End user desktop user agent strings should identify the browser, operating system, and may identify some key features. For example, on Windows 10, Internet Explorer 11 identifies itself with this user agent[45]: "Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko". The list of user agents should be reviewed using long tail analysis. You are looking for user agents that are not used by your installed browser base, such as a web spider, a scanner, misspellings of browser user agents, the Python programming language, or applications self-identifying by the user agent string. Systems making routine use of the web and *not* supplying a UA string should also be investigated, because this isn't normal.
4. **Name to IP mismatches *over a short amount of time:*** Sites do not normally change their IP address frequently. Site to IP relationships that change more than a few times within a day are suspicious, aside from sites that are actually hosted from multiple IPs. Larger sites will be load balanced with multiple IPs so make sure to check the DNS results before going too far with this condition because sites can be load or geo balanced.
5. **First time use sites:** Users within an organization will display a habit of using the same set of sites, so the ability to detect a new site can be very useful. The very first time a site is seen may be a result of a user clicking on a spammy link, a banner add that takes a user to a malicious site, or some other condition. A quick check of new sites once per day can reveal an issue.
6. **Consistent, repeatable browser pings or beacons:** Small sets of data going to the same site (meaning DNS name) over time indicate that a persistence mechanism or something nefarious is in place on the sending system. You will likely find a variety of SaaS applications in use, or user installed "information push" applications like a stock ticker. Connections not in this inventory need to be investigated. A beacon will have specific characteristics: they will repeat on an interval, connection duration will be short, and most of the beacons will be about or exactly the same size. One powerful tool to perform beacon analysis is RITA from Black Hills InfoSec.
7. **HTTP/S traffic *without a preceding DNS lookup***. HTTP traffic to a site by IP without a preceding lookup is rare. To realize this condition, begin by reviewing HTTP/s requests directly to IPs, determine the ASN and country code. and then check the ASN and country code to determine the usage pattern. Avoid a DNS reverse lookup from the beginning, because that may tip off an attacker. Use a DNS lookup Web service instead[46]. It is possible that a site or application may redirect a user directly to an IP address, but with more and more content delivery networks in use and IPv4 overcrowding these conditions may indicate suspicious behavior.

---

[44] https://udger.com/resources/ua-list maintains a list of user agents.

[45] https://msdn.microsoft.com/en-us/library/hh869301(v=vs.85).aspx

[46] One example (no endorsement) https://www.whoisxmlapi.com/dns-api.php

8. **WebShells**: As discussed in Webserver and Application Server Activity, presence of a webshell would mean that either an internal user is actively compromising an external system, or the system has been compromised and the shell has been there for "a while".
9. **Web Access Including an Executable or script suffix (V1.02):** A increasingly common practice for an adversary is to install a minimal downloader or dropper, and then to pull down an EXE, PS1 file, VBS, or some other form of executable content. For example, the Sysinternals tool psexec can be downloaded directly from https://live.sysinternals.com/psexec.exe. Based on the list of common EXEx described on p. 120, monitoring for the sysinternals.com domain and also monitoring URL's that contain executables or script suffixes may indicate an intrusion event.

Webserver and Application Server Activity

Much of this data can be tremendously valuable for consumer analytics, so the organization may already have an existing capability when it comes to analyzing this data source. This is another case where the majority of the data is normal activity, and before data is sent to the SIEM, the data can safely be reduced.

Before you contemplate pulling in webserver data into the SIEM, investigate how web server log data is currently used. The web admins likely have profiles and software to analyze their logs. By reviewing these reports, a baseline can be established which can then drive how to detect threats to web servers. Remember that the SOC is looking for security issues, not marketing intelligence.

For an internal webserver, ensure that you collect authentication logs, the connecting user agent, and the source IP. If there is a way to record a logout, that's a bonus (users don't often click on the logout link). After login and logout activity, you would want a "critical transaction", which should show up as a submit on a specific page as a POST.

For an external webserver, you may have difficulty getting the true connecting IP. Many security architectures define a perimeter firewall with a NAT address in place, so the web server logs will not contain the true source, as the webserver is deployed in a DMZ using an RFC1918 address. More sophisticated perimeter security models employ load balancers or reverse proxies can use the X-Forwarded-For (XFF) HTTP header field. This field records the originating source address. The next step is ensuring that this field is recorded in the webserver logs, and then that the SIEM can parse the field.

Regardless of the decisions made about consuming web server logs, there are several conditions to monitor for web servers are described below. The most

valuable fields are the HTTP status code, URL path, remote IP or connecting host, request time, user agent, and request ID.

As a data reduction and learning exercise whittle down and eliminate "normal web traffic patterns" so that you can observe exceptions. Here are several considerations:

1. Produce an analysis of the number of unique pages or URL requests from a single source IP address. If 4/5 of your visitors interact with up to 125 URLs most of the time, then that represents the "normal usage pattern." You would confirm this by comparing the URL list against a stable DEV or QA version of the website – *not the production version*. There are two reasons for this. First, validate that QA mimics production. Second, if an attacker were capable of dropping off a malicious file like a webshell or the system, you would not want that file to be excluded. Now determine what remains in the last 1/5 of the time to detect spiders, vulnerability "testers", and other indicators.
2. The second option is to include just the log records that have security specific value. For example, when a user visits a particular page, such as the login page or specific forms.
3. Most log records should result in a 100 (information), 200 (success), or 300 (redirect) status code. These are all "normal" results. 400 and 500 level status codes, or a subset of them, can have real security value. For example, a 400 Bad Request, 401 Unauthorized, or 405 Method not Allowed are codes that can be a result of scanning or malicious server exploration.
4. A third option would be to update or modify the web application itself to log critical transactions – login, logout, page or form initial access, or key data changes and consume these log records.
5. If possible, then determine the "average visit time" for this period.
6. By understanding what is normal, your SOC team has a way of detecting when visiting IPs or DNS names are acting outside of this boundary. Further, automating some form of alerting for these conditions just may not be practical. Rather, scripting out this analysis and rerunning it on a monthly basis so that the SOC team just knows "what's normal" may be enough.

Key web and app server use cases:

1. **Webshells**: For server-side deployments, and in particular any system that is internet exposed and has a web server, configure file blocking for known webshells. There are several reputable lists of webshells, and there are numerous PHP shells on Github[47] (and likely many other places.)

[47] https://github.com/bartblaze/PHP-backdoors, https://github.com/JohnTroony/php-webshells, https://github.com/BDLeet/public-shell, https://github.com/tennc/webshell

2. **Attack Type Traffic:** Page requests that are *not part of the application URL Inventory.* If a user can successfully get a webshell on the server. To determine this, you would need a list of the URL's that the application has, and then look for exceptions. Something like "php-backdoor.php", "simple-backdoor.php", "RemExp.asp", or "kacak.php" should get your attention, as these are common backdoor webshell applications. Realize though – that this use case is focused on finding what should *not* be there, so if you see "/usermgmt/useraccthistlookup.asp" and it wasn't in the list of expected URL, don't assume that this is a new page focused on helping an end user see their account history. You will also detect an attacker performing a common URL scan against the system, when you connect unknown URLs with 400 error messages.
3. **HTTP return status codes:** These should be in the 100, 200, and 300 range most of the time. 400 and 500 error status codes indicate error conditions based on the HTTP request. An excessive number of either code indicates some form of reconnaissance or an attacker searching for a vulnerability.
4. **Injection Strings:** Useful strings in an IIS log that relate to an attacker attempting to penetrate a connected SQL server include:
    a. XP_CMDShell – attempt to invoke something at the command line
    b. Select * - attempt to get all data from a table
    c. Or 1=1 – an attempt to perform SQL injection
    d. SQL keywords – update (change table), cast (manipulate data type), union (merge multiple result sets)
    e. Pages that normally receive POST requests are also receiving GET requests
    f. Windows process names like "cmd.exe" and Linux shells like /bin/sh.
5. **Sessions**: Long running sessions. Consider looking into sessions that are 3 times the typical session length to start. This condition *may* indicate account compromise, because an attacker may be constantly interacting with the site.
6. **Spidering IPs**: IPs that visit most, if not all, of the possible URL's, visit URL's rapidly (say, thirty within a second or two), or generate numerous page failures indicate some form scanner, vulnerability analysis, or web content spider. Users who interact with a web server don't click on thirty links or buttons within a few seconds, unlike a web spider program or a scanner. Having your website indexed in Google isn't a technical attack, but it may be a data leakage issue.
7. **Application server conditions**: The application server provides back end access to the supporting database on behalf of a front-end web application. It is frequently a target of the attacker. There are security issues identified by the Open Web Application Security Project (OWASP) that can percolate

through to an application server. For example, any SQL injection statement run on the app server, or SQL error log conditions coming from an application server are suspicious.

Windows Firewall (V1.02)

Windows advanced firewall, when enabled, can write a variety of events to the Security log. Process ID's in WFAS events can be correlated to 4688 events by process ID. Protocols are listed by their numeric value, in decimal (TCP = 6, UDP = 17).

Event ID	Name
5146	The Windows Filtering Platform has blocked a packet.
5146	The Windows Filtering Platform has blocked a packet.
5147	A more restrictive Windows Filtering Platform filter has blocked a packet.
5148	The Windows Filtering Platform has detected a DoS attack and entered a defensive mode.
5149	The DoS attack has subsided and normal processing is being resumed.
5150	The Windows Filtering Platform has blocked a packet.
5151	A more restrictive Windows Filtering Platform filter has blocked a packet.
5152	The Windows Filtering Platform blocked a packet.
5153	A more restrictive Windows Filtering Platform filter has blocked a packet.
5154	The Windows Filtering Platform has permitted an application or service to listen on a port for incoming connections.
5155	The Windows Filtering Platform has blocked an application or service from listening on a port for incoming connections.
5156	The Windows Filtering Platform has allowed a connection.
5157	The Windows Filtering Platform has blocked a connection.
5158	The Windows Filtering Platform has permitted a bind to a local port.
5159	The Windows Filtering Platform has blocked a bind to a local port.

Windows Process (Sysmon and EventID 4688) (V1.02)

When evaluating what processes to monitor in the environment, consider a few key questions.

1. How is remote administration and what are the remote execution tools used to perform remote management by system custodians? Common methods include direct RDP, WMIC, WinRM, psexec, ssh access, Group Policy, and package build and deployment.
2. How is remote access to servers granted? Are users added directly to a local group, is there an AD group, or is there a privileged access solution?
3. What are the network flows for remote access? For example, are jump boxes used (see p. 92), specific segments for IT management, are end users granted RDP or SSH access, and what are the user accounts.

If you do not have an Endpoint Detection and Response agent, then the next best tool is to instrument Windows to collect process invocation and command line data. Windows event 4688, which can be instrumented on Server 2008 and forward to collect detailed process execution. With Server 2012 the full command line[48] can be captured by enabling the "Audit Process Creation" audit policy and the "Include command line in process creation events" GPO settings. Note that the time for the 4688 event is not contained within the event itself, it comes along with the event.

Table 24 Example 4688 Event

```
A new process has been created.

Creator Subject:
      Security ID:        SYSTEM
      Account Name:       DONSPC$
      Account Domain:     WORKGROUP
      Logon ID:           0x3E7

Target Subject:
      Security ID:        NULL SID
      Account Name:       -
      Account Domain:           -
      Logon ID:           0x0

Process Information:
      New Process ID:           0x3cf0
      New Process Name:   C:\Windows\System32\svchost.exe
      Token Elevation Type:     %%1936
      Mandatory Label:          Mandatory Label\System
Mandatory Level
      Creator Process ID: 0x338
```

[48] On a standalone system, can enable this setting in the local system registry by creating and setting the HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System\Audit\ ProcessCreationIncludeCmdLine_Enabled registry DWORD value value to 1. Since a local system does not process a domain applied GPO, the registry value needs to be manually created.

```
        Creator Process Name:
        C:\Windows\System32\services.exe
        Process Command Line:
        C:\WINDOWS\system32\svchost.exe -k wusvcs -p -s
WaaSMedicSvc

Token Elevation Type indicates the type of token that was
assigned to the new process in accordance with User Account
Control policy. (remainder deleted for brevity).
Note: 0x338 is 824 in decimal.
```

The next step is to install Sysinternals sysmon application. Sysmon can be configured to record more detailed information about processes and network connections. Sysmon records key details about processes, network connections, registry changes, hash values for a binary, parent process ID for process execution, and other key OS details. Instrumenting sysmon in your environment is essential for well-informed incident handling. The corresponding sysmon event for the 4688 event from above is:

Table 25 Example Sysmon Event

```
Process Create:
UtcTime: 2018-09-12 20:54:06.836
ProcessGuid: {1834af5b-7cee-5b99-0000-0010e0808202}
ProcessId: 15600
Image: C:\Windows\System32\svchost.exe
FileVersion: 10.0.17134.1 (WinBuild.160101.0800)
Description: Host Process for Windows Services
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
CommandLine: C:\WINDOWS\system32\svchost.exe -k wusvcs -p -s
WaaSMedicSvc
CurrentDirectory: C:\WINDOWS\system32\
User: NT AUTHORITY\SYSTEM
LogonGuid: {1834af5b-f063-5b98-0000-0020e7030000}
LogonId: 0x3E7
TerminalSessionId: 0
IntegrityLevel: System
Hashes:
SHA256=C9A28DC8004C3E043CBF8E3A194FDA2B756CE90740DF217548833728
1B485F69,IMPHASH=3A8297483C1777054C7BAFEA5E6A8853
ParentProcessGuid: {1834af5b-f063-5b98-0000-001064510100}
ParentProcessId: 824
ParentImage: C:\Windows\System32\services.exe
ParentCommandLine: C:\WINDOWS\system32\services.exe
```

Sysmon is configured using an XML file, of which there are several well-known

examples[49]. As you consider deploying sysmon, start with a default configuration (no XML file), let it run for a few days, and then review what it has found. Here, use long tail analysis in order to arrange the results from most to least so you can find the high volume binaries, and the singletons. From within a PowerShell ISE or PowerShell command shell, run with administrative rights, and run code below to show the ImageName (4th property). During an incident, you would want the command line (8th property), particularly for PowerShell. Also, singletons are examples of "rare executables".

One caution: sysmon data cannot fully replace netflow data because it does not provide the number of bytes in the connection, nor can it provide duration.

Table 26 Powershell code to list Sysmon EXE's in Long Tail Analysis order

```
$Hash = @{}
$entries = Get-WinEvent -filterhashtable
@{logname="Microsoft-Windows-Sysmon/Operational";id=1} |
%{$_.Properties[3].Value}
#| group-object Value
foreach ($l in $entries)
{
   # write-output $l
   if ($Hash[ $l ] -eq $null ) {
    $Hash[ $l ] = 1;
   } else {
    $Hash[ $l ]++;
   }
}
$Hash.getenumerator() | sort -Descending -Property value
| ForEach-Object {
   $msg = '{0} {1}' -f $_.value, $_.Key
   write-output $msg
}
```

The above code yields the following chart, as an example of long tail analysis:

LTA based charts group data together on a like basis. The intention of an LTA chart is to observe what happens at the singleton layer because single events are rarities, odd ducks, and may reveal security issues.

[49] GitHub sources – MotiBa, SwiftOnSecurity, ion-storm, MHaggis, Malware Archeology.

LTA can be used for a wide variety of data sources. If you took NIDS alerts then you can look at the high-volume alerts and better tune them, with the goal of decreasing the alarm count where possible. LTA is a powerful technique to understand your data.

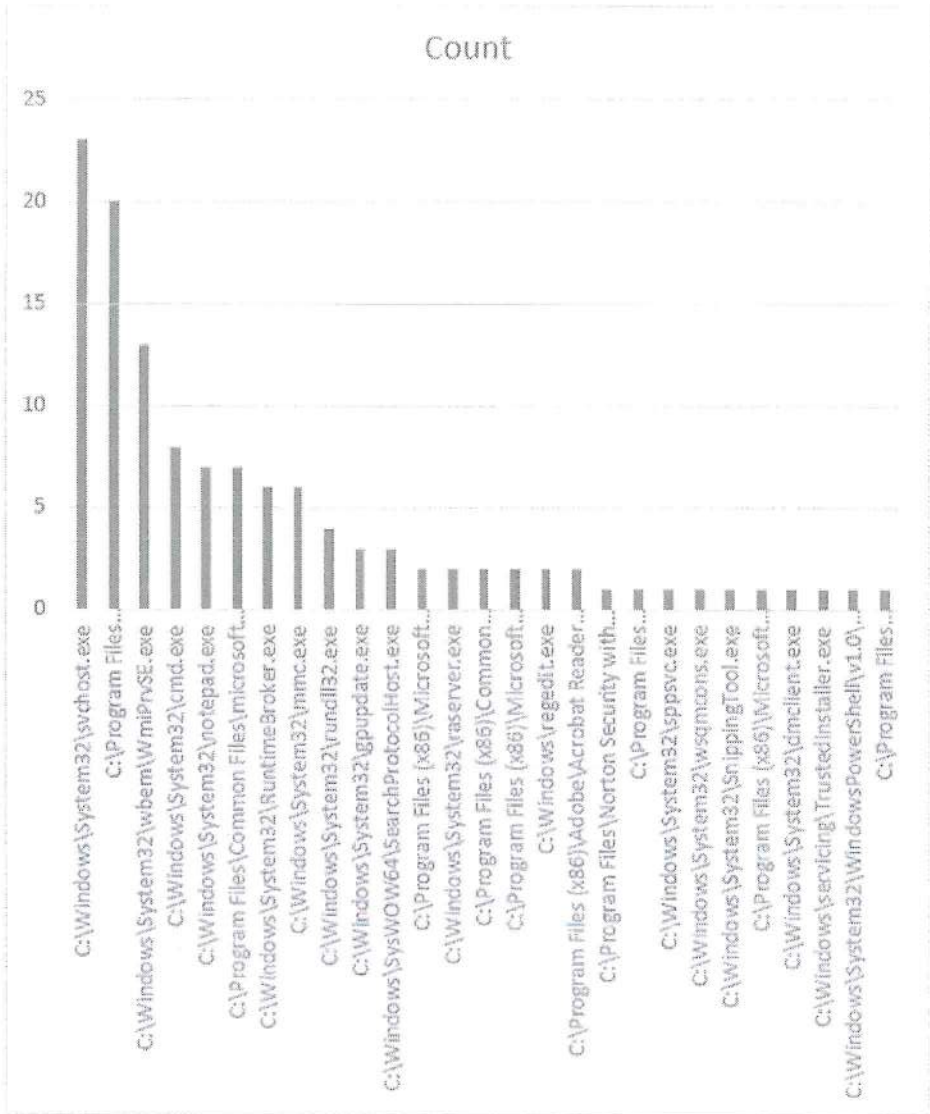


Figure 5 Windows Sysmon Process Long Tail Analysis

Table 27 Microsoft-Windows-Sysmon/Operational (v 7.01 as of March 2018)

Event ID	Name
1	Process creation

Event ID	Name
2	A process changed a file creation time
3	Network connection
4	Sysmon service state changed
5	Process terminated
6	Driver loaded
7	Image loaded
8	CreateRemoteThread
9	RawAccessRead
10	ProcessAccess
11	FileCreate
12	RegistryEvent (Object create and delete)
13	RegistryEvent (Registry value set)
14	RegistryEvent (Registry object renamed)
15	FileCreateStreamHash (File stream created)
16	Sysmon configuration change (cannot be filtered)
17	PipeEvent (Named pipe created)
18	PipeEvent (Named pipe connected)

Windows Process Execution Patterns and IoC's (V1.02)

Various Windows applications and binaries can be misused, or used to investigate a system, by an attacker. Most of these commands are executed early through an attack with a parent process name of "cmd.exe" or "powershell.exe". For the sake of organization, they are listed in alphabetic order by binary or topic, not frequency of usage.

For Version 1.02, this list was expanded. *Nonnative windows executables are indicated with a double asterisk after the exe name* - pwdump?.exe** is an example.

- **at.exe**: Used to schedule a job.
- **attrib.exe:** Can be used to hide files and directories.
- **cmd.exe**: There are a number of oddities that you can detect when cmd.exe is the parent process. Of note – cscript.exe. wscript.exe, and powershell.exe can be used by attackers. Further, when a productivity application such as Word, Adobe or Excel launch cmd.exe, the source file is most likely up to no good.
- **cscript.exe/wscript.exe**: These are older scripting tools, predating PowerShell, and are still viable today.

- **csvde.exe/ldifde.exe**: Can be used to extract Active Directory information into CSV files (bonus if you happen to find them, and the Admin Pak isn't installed).
- **dsquery.exe**: Used to extract a wide variety of information from Active Directory. Dsquery is more often used to extract user and group information.
- **dsget.exe / nltest.exe:** Used to determine the domain controller and its IP for the local logon session.
- **fsinfo.exe**: Used to get the list of connected drives
- **ipconfig.exe**: Get the NIC and DNS configuration.
- **mimikatz.exe**:** Used to extract plain text passwords, Kerberos tickets, hashes, and PIN codes from a running Windows system. The tools author also indicates it may be known as kdll, kdllpipe, and katz[50].
- **net commands**: There are numerous net commands – like "net localgroup administrators" to find out who is in the local Admin group.
- **netsh advfirewall**: Used to review and/or change the local firewall configuration.
- **netstat.exe**: Get list of listening ports.
- **ntdsutil.exe**: This is an Active Directory admin tool, and is used by adversaries for AD recon and configuration data. In particular, it is possible to extract the NTDS.DIT file.
- **ping.exe**: Test connectivity using ICMP. If the site permits ICMP to exit the network, the adversary can send an echo request to a site and detect that the request was allowed to leave the network. If so, then ICMP C2 is a viable data exfiltration method.
- **psexec.exe:** This Sysinternals tool can be used to execute remote commands on a Windows system, which it does by temporarily installing a service on the target. There should be a 5145 event in the Security log against the *\ADMIN$ share name. 7045 event in the system log when the service was remotely installed. Sysmon events 1 and 2 also provide traces.
- **pwdump?.exe****: Over the years, pwdump has appeared in many forms with increasing numbers in the file name. It is used to dump hashes and passwords.
- **reg.exe**: Query the registry, export and import sections, modify, or add keys to the registry.
- **rundll32.exe**: Rundll can be used to execute a script or invoke a DLL itself. Note that to invoke a DLL, the DLL name and the entry point for the DLL are specified on the command line.
- **qprocess.exe**: Displays process information.

[50] It may also appear with the name "mimidogs.exe".